

Person Of Interest Rapid Identification and Tracking (POIRIT) Assistance System



By: **DANISH ARIF**

Thesis submitted to the Department of Electrical Engineering, University of Cape Town, in complete fulfilment of the requirements for the degree of Doctor of Philosophy

Supervisor: DR. SIMON WINBERG

November 2023

The copyright of this thesis vests in the author. No quotation from it or information derived from it is to be published without full acknowledgement of the source. The thesis is to be used for private study or non-commercial research purposes only.

Published by the University of Cape Town (UCT) in terms of the non-exclusive license granted to UCT by the author.

Declaration

I, **Danish Arif**, hereby declare that the work on which this thesis is based is my original work (except where acknowledgements indicate otherwise) and that neither the whole work nor any part of it has been, is being, or is to be submitted for another degree in this or any other university. I authorize the University to reproduce for research either the whole or any portion of the contents in any manner whatsoever.

Date: 7th April 2023

Acknowledgments

To my esteemed supervisor, ***Professor Dr. Simon Winberg*** for his guidance and technical contributions, a heart of gratitude. He has already been a great mentor in addition to a supervisor.

To my respectable postgraduate admin, ***Ms. Nicole Moodley*** for her support throughout my postgraduate studies.

To my colleagues in the Department of Electrical Engineering, University of Cape Town, your valuable insights are highly appreciated.

To my family, for their sacrifices and unconditional support. If it was not for them, I would have never achieved this milestone in my life.

Abstract

Pakistan, situated in a highly militarized neighborhood, has been entangled in persistent conflicts. The repercussions of the Soviet-Afghan and US-Afghan wars led to an influx of millions of refugees into Pakistan, resulting in significant financial and security challenges. In response, Pakistan's armed forces and intelligence agencies successfully cleared affected areas of terrorist presence, ushering in a semblance of peace. However, this progress gave rise to new issues, as terrorist and militant factions covertly infiltrated cities, engaging in both acts of terror and street crime, thereby compromising urban safety. To counter this, the government established a specialized street crime patrol unit known as "Dolphins," equipped with modern surveillance technology, including video feedback systems. Additionally, the "Punjab Safe Cities" initiative was launched, involving the installation of cameras across cities to monitor citizen movements. While the existing system boasts 10,000 operational cameras in Lahore, it primarily relies on manual monitoring, lacking advanced features for proactive threat detection. To address this, an automated solution employing Artificial Intelligence (AI) is proposed, specifically designed to track "High-Value Targets" by processing visual data feeds from the urban environment. It begins with image acquisition from diverse network CCTV cameras across different geographical locations. This involves the implementation of network cameras with controlled frame rates, facilitated by Raspberry Pi nodes and augmented with PIR sensors for efficient frame change detection. The parallel operation of nodes and cloud uploads is integral to enhancing processing speed and reducing computational burden. Furthermore, the study deliberates on resizing images from varying CCTV camera resolutions, a critical step to mitigate computational costs without compromising performance. By employing the DORI rule, the frame resolution is adeptly adjusted, resulting in substantial memory and computational savings. The research also introduces a robust approach to cropping and storing essential facial features, emphasizing the significance of bizygomatic measurements and additional regions to account for tilts. In terms of face detection techniques, the study advocates for the utilization of YOLO v3 due to its superior performance in detecting small face faces from CCTV frames. POIRIT showcases an accuracy rate of 90.7% while detecting small-sized faces in challenging environments which makes it superior to previous studies. The research further delves into the implementation of cascaded classification layers tailored to the demographic of Pakistan, efficiently categorizing images based on gender, face type, and facial hair styles. The study meticulously prepares datasets for training deep learning models, encompassing a diverse range of samples from various urban

contexts. The research employs transfer learning models like AlexNet, GoogLeNet, ResNet, and VGG-Face, ultimately determining the most effective model for each classification layer. For gender classification, VGG Face exhibits the best performance with an accuracy of 98.41%, which is 1% more than the best performer in 2021. Likewise, for ethnicity classification, VGG Face demonstrates an overall accuracy of 97.2% proposing a unique work of Pakistan region with higher accuracy than models trained for other ethnicities. The final layer of facial hair type classification reveals that ResNet-50 outperforms other models with an overall accuracy of 97.31%. This novel work can classify various facial hair styles, but its accuracy is higher than the facial hair detection systems available. These results demonstrate the effectiveness of the chosen models in their respective classification tasks. The system not only achieves high accuracy rates but also introduces advanced image enhancement capabilities, offering solutions for low-resolution image estimation and sketch-to-color conversions. It incorporates the use of Generative Adversarial Networks (GANs) to introduce advanced image enhancement capabilities. The results demonstrate the superior performance of MS Cycle-GAN, with lower Mean Square Error (MSE) and Root Mean Square Error (RMSE) values. Moreover, the study addresses database creation and accessibility, ensuring a centralized repository for organized storage and easy retrieval of cropped images. The proposed distributed computing system, leveraging Raspberry Pi, CPUs, and GPUs, is pivotal in parallel data processing. Additionally, the research considers environmental factors like fog and smog, incorporating the Retinex Algorithm with Wavelet transform for effective fog removal. The search for persons of interest is streamlined through radial matching using cosine similarity, streamlining the process, and enhancing efficiency. Overall, the research presents a comprehensive framework for an intelligent assistance system, incorporating cutting-edge techniques and addressing critical considerations for effective facial recognition in urban environments. The system aims to assist security agencies in tracking high-value targets in busy urban areas. It offers customizable solutions with the potential for diverse applications and can be enhanced with newer deep-learning models. This research holds promise for advancing biometrics and parallel computing, adaptable for global use. Further refinements could include additional target sub-classifications, activity monitoring, and optimized hardware configurations. Implementing this system in developing countries like Pakistan could lead to safer cities, driving economic growth through increased business activities and tourism. These positive outcomes would not only benefit the nation but also strengthen international ties and promote innovation.

Table of Contents

Declaration.....	2
Acknowledgments.....	3
Abstract.....	4
Table of Contents.....	6
List of Tables	10
List of Figures.....	11
List of Abbreviations	19
Chapter 1 Introduction.....	21
1.01 Background	21
1.02 Problem Statement	24
1.03 Research Objectives and Questions	31
1.04 Proposed Solution and Methodology	33
1.05 Organisation of Thesis	41
1.06 Scientific Contributions	42
1.07 Limitations of the Research	43
1.08 Summary	44
Chapter 2 Literature Review.....	46
2.01 Overview of Literature Review	46
2.02 Overview of Image Processing	47
2.03 Classification of Image Processing	48
2.04 Applications of Image Processing in Security	50
2.05 Techniques used in Biometric Systems	53
2.06 Methods of Face Detection and Recognition.....	55

2.07	Outline of Machine Learning.....	62
2.08	Outline of Deep Learning	65
2.09	Deep Learning Techniques	66
2.10	Architectural Models of Deep Learning	67
2.11	Applications of Deep Learning in Image Processing.....	70
2.12	Face Detection using CNN	73
2.13	Transfer Learning for Classification Problems.....	78
2.14	Deep Learning Frameworks.....	85
2.15	Parallel & Distributed Computing Platforms for Image Processing....	89
2.16	Intelligent Assistance Systems.....	96
Chapter 3 Methodology.....		98
3.01	Overview	98
3.02	Literature Review.....	99
3.03	Organizational Information.....	100
3.04	Public Feedback	102
3.05	Ethical Considerations	102
3.06	Corroboration of Novelty.....	103
3.07	Hardware and Software Acquisition and Integration.....	104
3.08	Dataset Preparations.....	105
3.09	Frame Capturing and Face Detection.....	106
3.10	Face Classification System Design	106
3.11	GPU Computing for Training and Prediction	107
3.12	GUI & Database Management.....	108
3.13	Experimentation and Evaluation.....	109
3.14	Analysis Methods.....	110

3.15	Validity and Reliability	111
3.16	Research Timeline	111
3.17	Resources and Budget	113
3.18	Limitations of Methodology	113
3.19	Summary	114
Chapter 4 System Design		116
4.01	Overview	116
4.02	Parallel Computing Stage I	118
4.03	Parallel Computing Stage II	146
4.04	Parallel Computing Stage III	185
4.05	Face Categorization System	196
4.06	Target Search System	206
4.07	Summary	220
Chapter 5 Experimentation		227
5.01	Overview	227
5.02	Materials and Apparatus	227
5.03	Experimental Setup	235
5.04	Face Detection	242
5.05	Models Training for Classification System	245
5.06	Models Training for Image Transformation	263
5.07	Experimental Results	269
5.08	Summary	271
Chapter 6 Results		272
6.01	Overview	272
6.02	Face Detection in CCTV images	278

6.03	Gender Classification.....	282
6.04	Face Type Classification.....	290
6.05	Facial Hair Classification.....	311
6.06	Face Alignment.....	332
6.07	Sketch to Hyper-Real Face Construction.....	332
6.08	Blur Image to High Resolution Image Conversion.....	334
6.09	Face Matching using Cosine Similarity	336
6.10	Summary	338
Chapter 7 Conclusion and Future Work		349
7.01	Overview of the Thesis	349
7.02	Important Findings.....	351
7.03	Recommendations and Future Work.....	351
REFERENCES.....		361
ANNEXURES		394

List of Tables

Table 1 Models Specifications for Raspberry Pi	93
Table 2 Models Specifications for NVIDIA Jetson	94
Table 3 HiKey970 Specifications	95
Table 4 Neural Compute Stick 2 by Intel Specifications	95
Table 5 Both Genders Mean Bizygomatic & Head Breadths	139
Table 6 Data Speed of Wireless Theoretical vs Practical	141
Table 7 NVIDIA GPUs Comparison	150
Table 8 Network Parameters.....	164
Table 9 Pixel Density Requirements	178
Table 10 Gender Classification Training Results	270
Table 11 Face Type Classification Training Results	270
Table 12 Facial Hair Type Classification Training Results.....	270
Table 13 Image Transformation Models Experimentation Results Summary...	271
Table 14 Results summary for Face Detection	282
Table 15 Results summary for Face Type classification using AlexNet	295
Table 16 Results summary for Face Type classification using GoogLeNet.....	302
Table 17 Results summary for Face Type classification using VGG Face	309
Table 18 Results Summary Facial Hair Type Classification Using GoogLeNet	316
Table 19 Results Summary Custom Network for Facial Hair Classification	323
Table 20 Results Summary for ResNet for Facial Hair Classification	329
Table 21 Pixel-Pixel Comparison Results for MSE and RMSE.....	334
Table 22 Pixel-Pixel MSE and RMSE Results	335
Table 23 Blur to High Resolution Image Conversion Result	336
Table 24 Face Match Results	337
Table 25 Gender Models Performance Comparison Table.....	338
Table 26 Face Type Models Performance Comparison Table.....	340
Table 27 Facial Hair Models Performance Comparison Table for Facial Hairs	344

List of Figures

Figure 1 Countries with Highest Number of Terrorist Attacks	23
Figure 2 Public Places Safety for Women	28
Figure 3 Actual Screenshot from CCTV footage of Street Crime 2019.....	28
Figure 4 PSCA Monitoring.....	30
Figure 5 Social Media Viral Controversial Picture	30
Figure 6 Research Methodology Flowchart.....	34
Figure 7 Use-Case Diagram of Database Management System	40
Figure 8 Hardware Block Diagram.....	45
Figure 9 Image Processing Applications	48
Figure 10 Face Occlusion Examples	53
Figure 11 Proposed Face Detection	53
Figure 12 Basic Flow of Recognition System	55
Figure 13 Sobel Convolution Kernels	56
Figure 14 Haar-Like Features	58
Figure 15 Integral Image Concept	58
Figure 16 Viola-Jones Face Detection	59
Figure 17: Three Level Octaves	61
Figure 18 Gaussian Second Order Partial Derivative	61
Figure 19 Decision Tree Example	63
Figure 20 Two Stage CNN Detectors Evolution	73
Figure 21 One Stage CNN Detectors Evolution.....	73
Figure 22 YOLO v3 Feature Extraction Layers.....	76
Figure 23 YOLO v3 Detection Layers.....	77
Figure 24 Transfer Learning Process.....	79
Figure 25 AlexNet Architecture	80
Figure 26 VGG Architecture.....	83
Figure 27 VGG Face Architecture	83
Figure 28 ResNet-50 Architecture	84

Figure 29 Ten Phases of the Research Project Spanning 98 Months.....	112
Figure 30 High Level Architectural Diagram	116
Figure 31 Software Architecture Diagram.....	117
Figure 32 Resolution vs FOV	120
Figure 33 CCTV Cameras PSCA Lahore	121
Figure 34 Resolution vs FOV	122
Figure 35 HFOV, Sensor Size and WD for AFOV	123
Figure 36 Shutter Rate Comparison: 1/30s vs 1/125s	125
Figure 37 IP Camera Image Settings Panel	125
Figure 38 Camera Installation with Field of View	128
Figure 39 Plan view of HFOV	129
Figure 40 Field of View Calculation	132
Figure 41 HFOV for Horizontal Distance calculation	133
Figure 42 Camera Configuration	134
Figure 43 Target and View Setting.....	134
Figure 44 Camera Views	135
Figure 45 Expected image PPF	135
Figure 46 PIR Triggered Raspberry Pi Computations	144
Figure 47 Population Density Map of Lahore	147
Figure 48 CPU vs GPU Comparison	149
Figure 49 Rig Overview with Components	152
Figure 50 Histogram of A Normal Image.....	154
Figure 51 Compressed Histogram Image	154
Figure 52 Histogram Equalized to 255 Levels.....	155
Figure 53 Corrected Histogram.....	155
Figure 54 Face Detection Process using Machine Learning	156
Figure 55 CCTV Face Detection Dataset (Day)	159
Figure 56 CCTV Face Detection Dataset (Night).....	159
Figure 57 YOLO Training Dataset Sample	159
Figure 58 ROI Automation with Custom Algorithm	160
Figure 59 ROI Generation in CCTV Footage.....	161

Figure 60 ROI Generation in CCTV Footage B	161
Figure 61 ROI Rescaling	162
Figure 62 2792 x 1571 Image Breakdown Scheme	163
Figure 63 2161 x 1216 Image Breakdown.....	163
Figure 64 YOLO Face Detection Process.....	165
Figure 65 Anthropometric Proportions	167
Figure 66 Facial Dimensions Key Points.....	168
Figure 67 Rhinion Point.....	170
Figure 68 Compensation Matrix Calculation.....	173
Figure 69 Compensation Face Matrices	174
Figure 70 70 Degrees Rotated Face	174
Figure 71 30 Degrees Rotated Face	175
Figure 72 System Results with and Without Compensation Regions	175
Figure 73 Input Frame with right bottom face	175
Figure 74 Output Images with and without compensation	176
Figure 75 Input Images for Gender/Face Type and Facial Hair Stage	176
Figure 76 Pixels Allocation for 50 x 82 Image.....	179
Figure 77 Pixel Division for 224 x 365 Image	182
Figure 78 Face and Facial Hair Matrices Pixel Allocation.....	184
Figure 79 Face Categorization Workflow.....	186
Figure 80 Feature Importance by Face Elements	186
Figure 81 Face Architecture by Gender and Ancestry	188
Figure 82 Subcontinent Population Origin	188
Figure 83 Study of Pakistani Ethnic Groups	189
Figure 84 Face Types Samples	190
Figure 85 Facial Hair Types.....	191
Figure 86 Dataset Generation Process Flow	193
Figure 87 Image Rotation at 5 degrees	194
Figure 88 Image Mirroring	195
Figure 89 Dataset Images Output	195
Figure 90 Person Categorization System High Level Flow.....	196

Figure 91 Frame Folder Distribution	197
Figure 92 Face Folder Distribution.....	198
Figure 93 Categorization System Low Level Flow	198
Figure 94 Classification Folders Scheme	199
Figure 95 Classification Folders Scheme till Gender Classification	200
Figure 96 Classification Folders Scheme till Face Type Classification	201
Figure 97 Classification Folders Scheme till Facial Hair Type Classification ..	202
Figure 98 Face Type Classification	204
Figure 99 Facial Hair Classification	205
Figure 100 Prediction System.....	206
Figure 101 Process of Proposed MS-CGAN	210
Figure 102 Face Without Facial Hairs	214
Figure 103 Face with Facial Hairs	214
Figure 104 Main Page View	215
Figure 105 Face Search Page	216
Figure 106 Camera Selection.....	216
Figure 107 Date Selection.....	217
Figure 108 Time Selection.....	217
Figure 109 Radial Nearest Neighbor Search	218
Figure 110 Sketch to Color Image Converter	218
Figure 111 Low/ Blur to High Resolution Converter	219
Figure 112 Face Alignment System.....	219
Figure 113 Add Camera Node	220
Figure 114 Stage I Hardware Diagram	220
Figure 115 Process Diagram of Stage I	221
Figure 116 Time Consumption at Processing Stage I.....	222
Figure 117 Stage II & III Hardware Diagram.....	223
Figure 118 Process Diagram of Stage II	224
Figure 119 Process Diagram of Stage III.....	225
Figure 120 Use Case Diagram of POIRIT	226
Figure 121 PIR Sensor Module Side and Front View	230

Figure 122 Circuit Board of PIR Module	230
Figure 123 Netelsan Ekson PIR	230
Figure 124 Dahua Network Camera	232
Figure 125 Camera with Light	236
Figure 126 Distance Markings	237
Figure 127 Background Alterations	237
Figure 128 Camera Light Condition Settings	238
Figure 129 Projection Light	239
Figure 130 Side Light	239
Figure 131 Aluminum Foil Sheet	240
Figure 132 Face Rotation	241
Figure 133 Off Camera Face	241
Figure 134 YOLO v3 Model Training curves	244
Figure 135 AlexNet Gender Classification training	248
Figure 136 AlexNet Face Type Classification	249
Figure 137 AlexNet Training Results for Facial Hair Classification	250
Figure 138 GoogLeNet Gender Classification Training Process	251
Figure 139 GoogLeNet Face Type Classification	252
Figure 140 GoogLeNet Training Results for Facial Hair Classification	253
Figure 141 VGG-Face – Gender Classification Training Accuracy & Validation Loss	255
Figure 142 Gender classification Training & Validation loss	255
Figure 143 Face Type Classification Training and Validation Accuracy using VGG-Face	256
Figure 144 VGG-Face Face Type Classification Training and Validation Loss	257
Figure 145 Custom Network Architecture for Facial Hair Type Classification	258
Figure 146 Training & Validation Accuracy Graph using Custom-Network for Facial Hair Type Classification	260
Figure 147 Training & Validation Loss Graph using Custom-Network for Facial Hair Type Classification	260
Figure 148 Training and Validation Curve for Facial Hair Classification using ResNet-50	262

Figure 149 Training Loss and Validation Loss Curve for Facial Hair Classification using ResNet-50.....	262
Figure 150 MS CGAN Layer and Generator Architecture	264
Figure 151 Input images X1 & X2 for MS- Cycle GAN.....	265
Figure 152 Generator and Discriminator Loss for MS Cycle GAN for Sketch to Color Image	266
Figure 153 Unpaired Low to High Resolution Dataset.....	267
Figure 154 Training Results for Low to High Resolution Image	267
Figure 155 Pixel Loss vs Iteration Graph	269
Figure 156 Intersection and Union	277
Figure 157 Recall vs Precision for YOLO v3.....	280
Figure 158 ROC curve AlexNet gender classification.....	283
Figure 159 Precision vs Recall Curve.....	284
Figure 160 Confusion matrix Alex-net Gender Classification	284
Figure 161 ROC Curve- GoogLeNet for Gender.....	286
Figure 162 Precision vs Recall Gender Classification GoogLeNet.....	286
Figure 163 Confusion matrix GoogLeNet gender classification	287
Figure 164 ROC - VGG-Face Gender Classification	288
Figure 165 Precision Vs Recall - VGG-Face Gender Classification	289
Figure 166 Confusion Matrix VGG-Face Gender Classification	289
Figure 167 ROC Curve for Face Type Classification using AlexNet	291
Figure 168 Precision vs Recall Curve Face Type Classification using AlexNet.....	291
Figure 169 Face Type Classification Confusion Matrix.....	292
Figure 170 Precisions Comparisons for Each Face Type	295
Figure 171 Recall Comparisons for Each Face Type	296
Figure 172 Accuracy Comparisons for Each Face Type	296
Figure 173 F1 Score Comparisons for Each Face Type	297
Figure 174 ROC Curve Face Type Classification GoogLeNet	298
Figure 175 Precision vs Recall – Face Type Classification GoogLeNet.....	298
Figure 176 Confusion Matrix for GoogLeNet Face Type Classification	299
Figure 177 Precisions Comparisons for Each Face Type	302

Figure 178 Recall Comparisons for Each Face Type	303
Figure 179 Accuracy Comparisons for Each Face Type	303
Figure 180 F1 Score Comparisons for Each Face Type	304
Figure 181 Face Type Classification ROC curve using VGG-Face	305
Figure 182 Face Type Classification Precision vs Recall Curve using VGG-Face	305
Figure 183 Confusion Matrix for VGG Face for Face Type Classification	306
Figure 184 Precisions Comparisons for Each Face Type	309
Figure 185 Recall Comparisons for Each Face Type	310
Figure 186 Accuracy Comparisons for Each Face Type	310
Figure 187 F1 Score Comparisons for Each Face Type	311
Figure 188 Facial Hair Classes Samples.....	311
Figure 189 ROC Curve for Beard Classification using GoogLeNet	313
Figure 190 Precision vs Recall Curve for Beard Classification using GoogLeNet	313
Figure 191 Confusion Matrix for Beard Classification using GoogLeNet.....	314
Figure 192 Precisions Comparisons for Each Facial Hair Type.....	317
Figure 193 Recalls Comparisons for Each Facial Hair Type.....	317
Figure 194 Accuracies Comparisons for Each Facial Hair Type.....	318
Figure 195 F1 Score Comparisons for Each Facial Hair Type	318
Figure 196 ROC Curve for Facial Hair Type Classification using Custom-Network	319
Figure 197 Precision vs Recall for Facial Hair Type Classification using Custom-Network.....	320
Figure 198 Confusion Matrix for Beard Classification using Custom Convolutional Network.....	320
Figure 199 Precisions Comparisons for Each Facial Hair Type.....	323
Figure 200 Recall Comparisons for Each Facial Hair Type	324
Figure 201 Accuracy Comparisons for Each Facial Hair Type	324
Figure 202 F1 Score Comparisons for Each Facial Hair Type	325
Figure 203 ROC for Facial Hair Type Classification using ResNet-50	326

Figure 204 Precision vs Recall for Facial Hair Type Classification using ResNet-50.....	326
Figure 205 Confusion Matrix for Beard Classification using ResNet-50.....	327
Figure 206 Precisions Comparisons for Each Facial Hair Type.....	330
Figure 207 Recall Comparisons for Each Facial Hair Type	330
Figure 208 Accuracy Comparisons for Each Facial Hair Type	331
Figure 209 F1 Score Comparisons for Each Facial Hair Type	331
Figure 210 Face Alignment Results.....	332
Figure 211 Sketch to Color Image Conversion Results.....	333
Figure 212 Visual Inception Poll Result.....	333
Figure 213 Low Resolution to High Resolution Image Generation Using MS Cycle GAN.....	335
Figure 214 Face Matching Scores Example	337
Figure 215 Gender Classification Model Comparison	339
Figure 216 Face Type I Comparison	341
Figure 217 Face Type II Comparison	341
Figure 218 Face Type III Comparison.....	342
Figure 219 Face Type IV Comparison.....	342
Figure 220 Overall Face Type Comparison.....	343
Figure 221 Facial Hair Type I Comparison	345
Figure 222 Facial Hair Type II Comparison.....	345
Figure 223 Facial Hair Type III Comparison.....	346
Figure 224 Facial Hair Type IV Comparison	346
Figure 225 Facial Hair Overall Comparison.....	347

List of Abbreviations

AI	Artificial Intelligence
ANN	Artificial Neural Network
AP	Average Precision
API	Application Programming Interface
BBOX	Bounding Box
BMI	Body Mass Index
CIA	Central Intelligence Agency
CNN	Convolutional Neural Network
CPU	Central Processing Unit
CTD	Counter Terrorism Department
CUDA	Compute Unified Device Architecture
DL	Deep Learning
DNN	Deep Neural Network
DORI	Detection Observation Recognition Identification
FOV	Field of View
FPS	Frames per Second
FSIM	Feature Similarity Index Metric
GAN	Generative Adversarial Network
GHz	Gigahertz
GPGPU	General Purpose Graphics Processing Units
GPIO	General Purpose Input/Output
GPU	Graphics Processing Unit
GUI	Graphical User Interface
HFOV	Horizontal Field of View
HPC	High Performance Computing
HSV	Hue, Saturation, and Value
IA	Intelligent Assistance
IAS	Intelligent Assistance System
IoU	Intersection over Union
IP	Internet Protocol

mAP	Mean Average Precision
mIoU	Mean Intersection over Union
MSE	Mean Squared Error
NADRA	National Database and Registration Authority
PFSA	Punjab Forensic Science Agency
PIR	Passive Infrared Sensor
POIROT	Person Of Interest Rapid Identification and Tracking
PPF	Pixels per Foot
PSCA	Punjab Safe Cities Authority
RGB	Red, Green, and Blue
RMSE	Root Mean Squared Error
ROC	Receiver Operating Characteristic
ROI	Region of Interest
SGD	Stochastic Gradient Descent
SPU	Special Response Unit
SSD	Single Shot Detector
SSIM	Structural Similarity Index
VFOV	Vertical Field of View
WD	Working Distance
YOLO	You Only Look Once

Chapter 1

Introduction

1.01 Background

Pakistan is a country with highly militarized neighboring countries, including Afghanistan, that has been at war for decades. Millions of refugees fled to Pakistan due to Soviet-Afghan and US-Afghan war. Pakistan became one of the world's largest refugee-hosting countries, bringing problems to itself, including financial and security issues. New terror groups emerged in Pakistan as a backlash to their involvement in these Afghan issues. Pakistan's armed forces, along with intelligence agencies have achieved much success in eliminating terrorists' presence and clearing occupied areas, which brought peace to the affected zones. But it has led to new problems, where terrorist and militant forces have started covertly infiltrating cities. Not only involved in terror activities but also street crime, making the cities unsafe [1]. At the start of this century, the world faced numerous terror attacks that pushed nations into a new level of defence and awareness – the fight against terror and terrorist actions has been referred to by the media as the 'War on Terror' [2]. Much of this war is not fought in traditional battlefields using weapons such as tanks and artillery to contest terrain termed as– 'asymmetric engagement' [3]. This asymmetric engagement, in essence, is where one side conforms to the established 'rules of engagement' that maintain human rights [4]. But the other side – which may or may not identify themselves and may or may not be part of a larger collective, does not adhere to such rules or concern for the basics of human rights; anyone, civilian or military, could be a target. Urban areas, particularly densely populated areas such as capital cities, are the most vulnerable context in which this 'fight' can occur [5].

It is thus essential for greater diligence in protecting people in these areas. Protecting the population from these terrorist actions and the effect of asymmetric engagement is a complex undertraining – taken all together it is a huge expense, and causes inconveniences to law-abiding citizens and travellers, and it requires vast amounts of trained and trusted people.

Accordingly, there is an urgent need to further technologies that can assist with this war on terror and help to keep populations safe from threats from terrorists and other criminal behaviour. Pakistan has been in the highlights of the press and travel literature for a long time as a country with safety risks due to significant security incidents. This country is thus not considered a safe destination for tourists due to these security threats in the recent past [6]. People of this country are generally welcoming and very friendly to foreign visitors and show much tolerance towards all, regardless of culture and religion [7]. Yet the overall image of the country is still not viewed as a safe place to live or visit, which is largely due to the terror attacks of recent years that happened in the country.

The frequency of the attacks was controlled after mass military operations against terrorism in the western part of the country. This helped to eliminate the majority of terror elements in the country [8]. Whenever such operations are performed, other criminal elements start to rise in the major cities. Big cities were the target when security forces were carrying out the operations. Police vans, markets, cinemas, worship places, etc. were targeted in the previous years to spread the wave of terror [9] [10].

To cope with this situation, law enforcement agencies were mobilized to devise fool-proof plans to counter such threats. More forces were deployed and trained to deal with these sorts of problems [11].

Due to its geographical placement, Pakistan experienced many terrorist attacks in which thousands of civilians lost their lives [12]. As seen in Figure 1, Pakistan was ranked third in terms of countries with high levels of terrorist attacks [13]. To deal with this situation, a large amount of security force was deployed. Existing forces are continuously needing to be trained so that they are aware of, and can respond to, the latest terrorist antics – as well as to further expand the forces that are protecting the populace. Technology solutions are essential to help win this War on Terror.

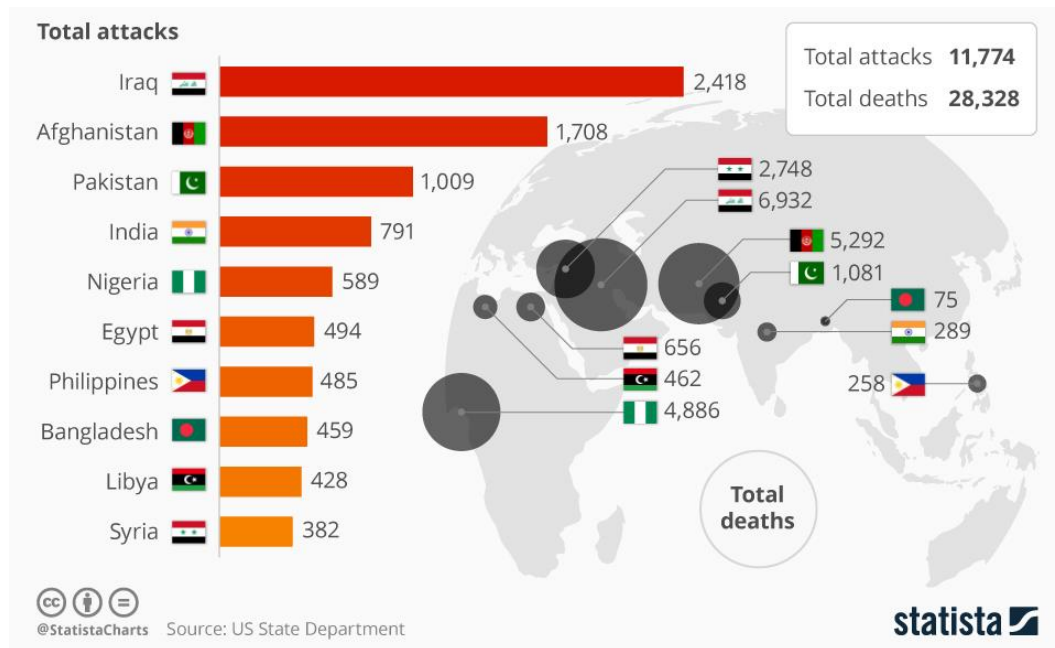


Figure 1 Countries with Highest Number of Terrorist Attacks [Source: Niall McCarthy, 2016] [13]

Technology is the best way to tackle any kind of threat at present [14]. Previously only a ‘firefighting’ approach (i.e., responding to problems that had started, or the aftermath) was being used that depended upon the humans to find the culprits after any incident. In Pakistan, this has certainly been the case for years. For instance, large expanses of the big cities are now being monitored by closed-circuit television (CCTV) cameras. The “Punjab Safe Cities Authority” (PSCA) was established in 2015 which is an initiative to make the Punjab region safer. Initially, PSCA installed 10,000 cameras for surveillance through a broad area in Lahore. This surveillance covers public offices, city entry and exit points, key infrastructure, major roads, crime hotspots, and Very Important Person (VIP) routes among other areas that could be targeted by criminals. The Punjab Safe City Authority was established to aid this fight against terrorism, which includes the use of technology to assist this objective. The first task regarding these technology aids was the installation of CCTV cameras in the biggest city of the Punjab province. This project required allied facilities, including a dedicated communication network, civil infrastructure, and a command centre. The camera and video feed infrastructure were completed within two years in the major areas of the Lahore and surveillance is being done with the help of more than 10 000 cameras. Additional cities in the province are to get similar infrastructure installed [15]. After cameras were installed, and once security forces were trained and made use

of these, the crime rate dropped. But a major issue is still that there isn't any system to track targets and find a target from the network of thousands of cameras. At present, this surveillance is manual; it requires trained humans to keep an eye on the screens to look for persons of interest or criminal actions. Considering the number of cameras that need to be watched, the demand for the number of humans for monitoring is huge. This system is thus inefficient in terms of the ability to dedicate staff to surveillance and track/find the targets.

1.02 Problem Statement

Artificial Intelligence (AI) has become one of the most rapidly emerging fields. It is commonly used in various daily life activities with both positive and negative impacts on our lives [16]. AI is frequently used in face recognition applications. Smartphones use face recognition in authorization and camera-related applications. With so many advancements, there have also been increasingly many openings and new opportunities for research in this field. Cities are getting smarter by improving their urban services in terms of quality and performance. Data analytics are improved using sensors, and are consuming massive amounts of data, such as video feeds from multiple cameras for improved surveillance.

Machine Learning is being used for capturing and processing this data. It has achieved promising results in various big data applications, like in smart city development [17]. Law enforcement personnel uniform-mounted smart face recognition system using Raspberry Pi and cloud computing has been proposed by M. Sajjad et.al. It uses a bag of features and rotated BRIEF points in the face [18]. This technique is slower to handle real-time scenarios due to the processing limitation of Raspberry Pi. According to a review of object detection/ tracking, image classification, face recognition, and labelling, deep learning models have been considerably upgraded so much to match human competencies [19]. This has been achieved by training huge datasets that are available to future developers. LSVRC-2010 ImageNet [20] training is a deep convolutional network to classify 1.3 million images having high resolution with 39.7% and 18.9% top-1 and top-5 error rates respectively. There are 0.5 million neurons and 60 million parameters with 5 convolutional layers. For faster training, it has been implemented on an efficient GPU [20]. To deal with large cities it needs a

network of camera nodes having compute capability so the work can be segmented and computed in parallel. Machine Learning develops algorithms from experience examples and predicts data from unknown data. It is successful in many applications of security, biomedical, computer vision, etc. Supervised machine learning will involve a lot of human work for labelling sessions. It is less efficient in terms of time and space complexity. In comparison, deep learning increases efficiency by using multiple layers of non-linear transformations [17].

To address the above-mentioned situations, there are a few systems that are being developed and are under the test phase. These include systems that are being developed in Dubai (United Arab Emirates) named “*Oyoon*” which was operational in 2019 [21]. In Xinjiang (China), the system automatically searches for the desired person. China has the biggest monitoring system in the world having around 170 million CCTV cameras and was planned to add 400 million more by 2020. Artificial Intelligence (AI) companies are working with the government, such as unicorn Face++, which will extract significant material such as faces, registration plates, etc. from video frames [22]. Many cities have responded to the COVID-19 pandemic's challenges by putting in place cutting-edge tracking systems to guarantee public safety. Singapore's Safe Distancing Enforcement Robot uses cameras and sensors to keep an eye on people adhering to safe distance rules in public areas [23]. Similarly, to ensure commuter safety in the city's extensive public transport system, Transport for London (TfL) uses a vast network of CCTV cameras for tracking [24]. The Domain Awareness System (DAS), a comprehensive surveillance system created in collaboration with Microsoft and used in New York City, combines information from a variety of sensors, such as environmental sensors, CCTV cameras, and license plate readers [25]. With its Smart CCTV Network, Seoul has adopted a more technologically advanced strategy, utilizing artificial intelligence to monitor vehicle and pedestrian traffic, identify anomalous behaviour, and improve public safety in general. [26]. Together, these systems demonstrate a wide variety of technologies and strategies used to meet the needs of urban tracking, each specifically designed to meet the demands and particular obstacles of the cities in question.

Numerous methodologies are used in the field of tracking people in urban environments, each with unique advantages and disadvantages. Vision-based methods track and monitor people using cameras and image processing algorithms. Although these techniques work well in controlled settings, occlusions, changing lighting, and complex backgrounds can make it difficult for them to be accurate in real-world urban settings [27]. Continuous surveillance also raises significant privacy issues and ethical questions, which call for careful thought and mitigating measures [28]. Sensor fusion techniques combine information from various sensors, including GPS, LiDAR, and cameras, to improve tracking precision. The need for complex data fusion algorithms and the high cost of deploying numerous sensors throughout an urban area presents a challenge to this approach, despite its promise [29].

Standardized protocols and calibration methods are also necessary for the integration of various sensor data sources, and these are still being researched in the field. Convolutional neural networks (CNNs) are machine learning and deep learning techniques that have shown promise in enhancing the accuracy and resilience of human tracking. CNNs are particularly good at tracking and recognizing objects. Nevertheless, their efficacy is dependent on having access to extensively annotated datasets, which may be difficult to come by for urban settings and tracking scenarios [30]. Furthermore, there are important questions that need to be answered regarding the interpretability and explainability of deep learning models. Tracking accuracy and situational awareness may be improved by incorporating social and behavioural context, such as knowing crowd dynamics and pedestrian interaction patterns [31]. However, this integration necessitates a deeper understanding of how people behave in urban settings, which highlights the value of interdisciplinary cooperation between social scientists and computer vision specialists. Keeping the above concerns in view, Myneni et.al., 2022 [32] proposed the Smart-city Video Surveillance paradigm, a large-scale distributed video surveillance service. It is a platform for collecting and analyzing video surveillance data that is used to monitor, safeguard, and decide on crucial events for smart campus and city applications. That specific research topic is to recognize and deanonymize human faces in a distributed edge cloud computing architecture [32].

There are still several unfilled research gaps in human tracking. One major challenge is real-time scalability, especially in densely populated urban areas, which calls for the creation of effective algorithms that can manage large-scale tracking in real-time scenarios. The development of strong anonymization methods and frameworks that strike a balance between the benefits of tracking and individual privacy rights is imperative to address privacy concerns. One of the main challenges in adapting to dynamic urban environments is the ever-changing nature of elements like weather, lighting, and pedestrian behaviour. The goal of research should be to create methods that can adjust to these changing circumstances. In addition, the availability of realistic and varied benchmark datasets is critical for assessing and contrasting various urban tracking methods. One area that needs to be focused attention in the field is the development of standardized datasets that accurately represent real-world urban scenarios.

As discussed earlier in this section the PSCA has installed a CCTV cameras network to provide security to the civilians. There are various challenges for this system due to multiple reasons. Lahore, Pakistan city has a population of 11.12 million of which 5.3 million are females, making up around 48%. In “*Empirical study on women safety concerns at public places: a case study of Lahore City*” conducted by Bajwa, M.U., Khan et. al., 2018 [33] discuss the perception of women about how safe they feel in Lahore. The study shows that 80% of women think public places are not safe for women as displayed in Figure 2. It can be elaborated in such a way that 38% population thinks they are not safe whereas 52% population’s feedback isn’t available in the study [33].

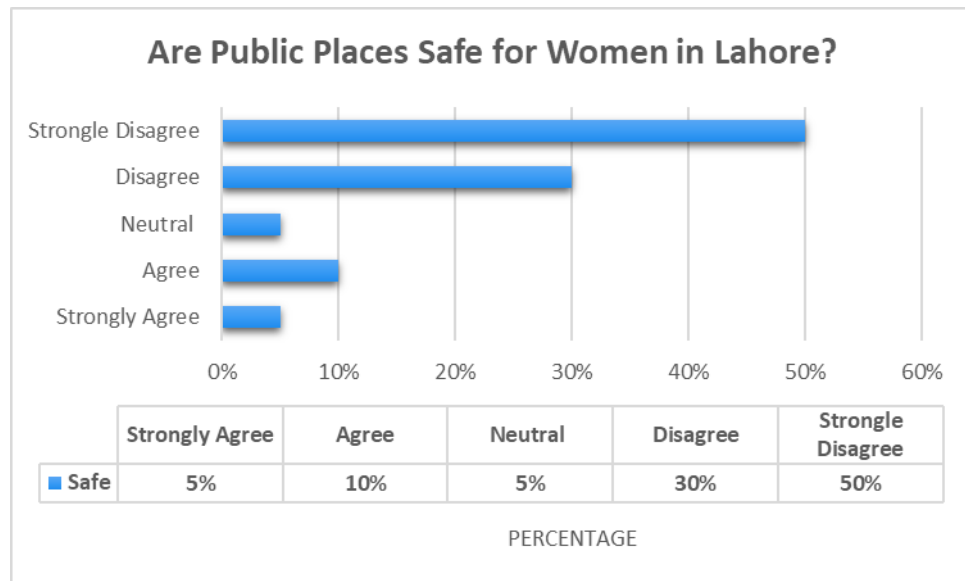


Figure 2 Public Places Safety for Women [Adapted from Bajwa, 2018] [33]

Pakistan is moving forward to make the cities smarter and safer. Huge investments are made by installing the camera network. This system can monitor all the major junctions of the city, but it is limited to observation only. In addition to the government-owned system, CCTV cameras are installed by the public to guard their premises. Not much planning is done while selecting the camera to get enough pixels per face to identify the target. As shown in Figure 3 picture retrieved from the CCTV footage taken by street camera does not allow the observer to identify the target well. Moreover, if the resolution is maintained, even then there isn't any automated method for identification and tracking of the suspect.



Figure 3 Actual Screenshot from CCTV footage of Street Crime 2019 [Source: PSCA, 2019] [34]

Currently, all the monitoring is performed at the main command and control centre where law enforcement officers keep an eye on the video streams. Figure 4 presents a

view of the command-and-control centre. Due to human involvement in monitoring the streets of the city through CCTV, another issue has been stimulated when inappropriate pictures of individuals went viral on the internet. Being an Islamic Republic, this issue triggered anger in the society resulting in the allegations that there are no policies to data privacy. All the claims are rejected by PSCA officials by the argument that data is only shared with investigation officers in case of any First Information Report (FIR) [35]. However, this statement cannot assure that these images can't be misused and denial of human involvement in the process of monitoring. Similar issues are reported in other parts of the world as well. Camera operators have been observed to focus disproportionately on people of colour in Great Britain. Black people were between one-and-a-half and two-and-a-half times more likely to be monitored than one might anticipate given their prevalence in the population, according to a sociological assessment of how the systems were run. Most of the male operators of the camera systems in Britain routinely use the cameras to voyeuristically spy on women, according to experts examining how the systems are run in Britain. The study discovered that almost one in ten women were targeted only for voyeuristic purposes.

In the US, numerous incidents have been reported. In one, partners enjoying the "privacy" of their rooftop balcony at night was caught on an infrared video camera by New York Police Department in an ariel surveillance that was ostensibly observing the gatherings during the convention held in 2004. Sometimes surveillance procedures can give the security officers opportunity to misuse the footage. Detroit Free Press exposed that women were stalked by the security force employees and their friends using the Michigan Police database [36].



Figure 4 PSCA Monitoring [Source: PSCA, 2019] [37]



Figure 5 Social Media Viral Controversial Picture [Source: SAMAA News 2019]

In response to the allegations, PSCA claims that they only take pictures of number plates. but it gives another dimension that culprits cannot be tracked. This project that was initiated from the city of Lahore, now has been extended to six more cities including Rawalpindi, Gujranwala, Sargodha, Multan, Faisalabad, and Bahawalpur. Making privacy a concern of more than 30 million people.

Nighat Dad who is cyber security expert refers that there is regulation act of 2016 to protect data. It will help citizens to get their freedom without any privacy breach. The Government of Pakistan took years in the making but now, as of June 2023, this protection bill is approved [38]. Figure 5 shows the viral social media picture in which civilians were captured during their private moments, which was published in SAMAA News (Pakistani News Channel) that caused anger in the society. Car registration plates and humans are blurred for confidentiality purposes.

As discussed in this section there are problems with human centred surveillance like scalability, accuracy, efficiency, and privacy. Whereas eliminating humans from the system is also not possible because human intelligence is required in many cases. A hybrid system can be a solution in which humans and machines can work together and achieve best results. Such systems are referred as Intelligent Assistance Systems and are made in such a way that humans and machines can work together to solve complex problems efficiently.

1.03 Research Objectives and Questions

This research provides solution to the existing problem of manual surveillance with no automated search technology with the help of intelligent assistance technology by prototyping a full-scale person of interest tracking system with distributed computing technology at low cost, having efficient cascaded classification system designed for multiple ethnicities. This can maintain privacy of the citizens by limiting human involvement in tracking the targets from CCTV cameras.

The aim of this research is to develop an intelligent assistant system that can integrate with the currently installed CCTV networks in cities with minimum hardware additions that can efficiently search/track targets by classifying the public into categories; and keep the privacy by restricting the user to view the target frame only. This system can also assist humans in i) to estimate the colour image from man-made sketches ii) improving resolution of blur/low-resolution images and iii) to predict the frontal face from the side pose image. It is designed especially for the Indian subcontinent with the ability to deal with potential scenarios of this region while

keeping in view the local concerns of privacy that can be integrated with existing surveillance systems with minor hardware and software additions.

Following this broader project aim the following research sub-objectives for this project were selected:

1. Design and implement a hardware system capable of efficient image acquisition from IP-based CCTV cameras using distributed computing devices.
2. Acquire images from a CCTV camera and implement a technique to resize frames in space and crop regions of interest.
3. Development of efficient image detection deep learning model to detect small-size human faces from CCTV images in real-time.
4. Development of a new face classification system for the Pakistan Region, using transfer learning for a customized cascaded system.
5. Develop GAN-based AI models for image enhancement i.e., Low-to-High resolution conversion and Sketch to Hyper-Real image conversion.
6. Integration of sub-systems to form a complete solution for target detection and tracking and development of a secured database system to store the processed images along with the capability to apply filters for searching.

All these proposed parts will be broadly discussed with analysis and experimental verifications in this thesis. Comparisons with existing techniques will be shown to demonstrate the efficiency improvements of the POIRIT system. Secondly, hardware prototypes are integrated to certify their benefits in real-time scenarios.

From the above objectives following research questions were generated.

1. How will images be acquired from various cameras placed at different geographical locations?
2. How images from different resolutions of CCTV cameras will be resized to reduce the computational cost without affecting the performance?
3. What region along with the face should be cropped and stored by keeping all the essential features?
4. What techniques will be used to detect human faces from CCTV images?

5. Which cascaded classification layers will be introduced to make an efficient categorization system for the Indian subcontinent?
6. How datasets will be prepared for training deep learning models?
7. Which Deep Learning algorithm will be used for the recognition system?
8. What image enhancements are required for the surveillance system and what techniques will be used to address the problem(s)?
9. How a central database will be created to store cropped images and their attributes?
10. How different users will access the database and search for a person of interest?

The above questions are answered through a comprehensive research design presented in Chapter 4. The experimental process and results are given in chapters 5 and 6. Whereas, the answer to each question is summarized in Chapter 7.

1.04 Proposed Solution and Methodology

The proposed POIRIT system is an intelligent assistance system that can work with humans and assist in finding human targets efficiently from the network of cameras. The purpose of this research is to develop a functional prototype that can deal with multiple camera nodes having the capability of initial pre-processing and interacting with the main database.

To effectively address the defined objectives and sub-objectives, this research project undertakes a comprehensive exploration of critical research questions. These inquiries serve as pivotal areas of focus for developing a viable solution. This section explains how this research project was conducted. It will highlight the steps involved in carrying out the work. A detailed flow diagram indicating the major activities are shown in Figure 6.

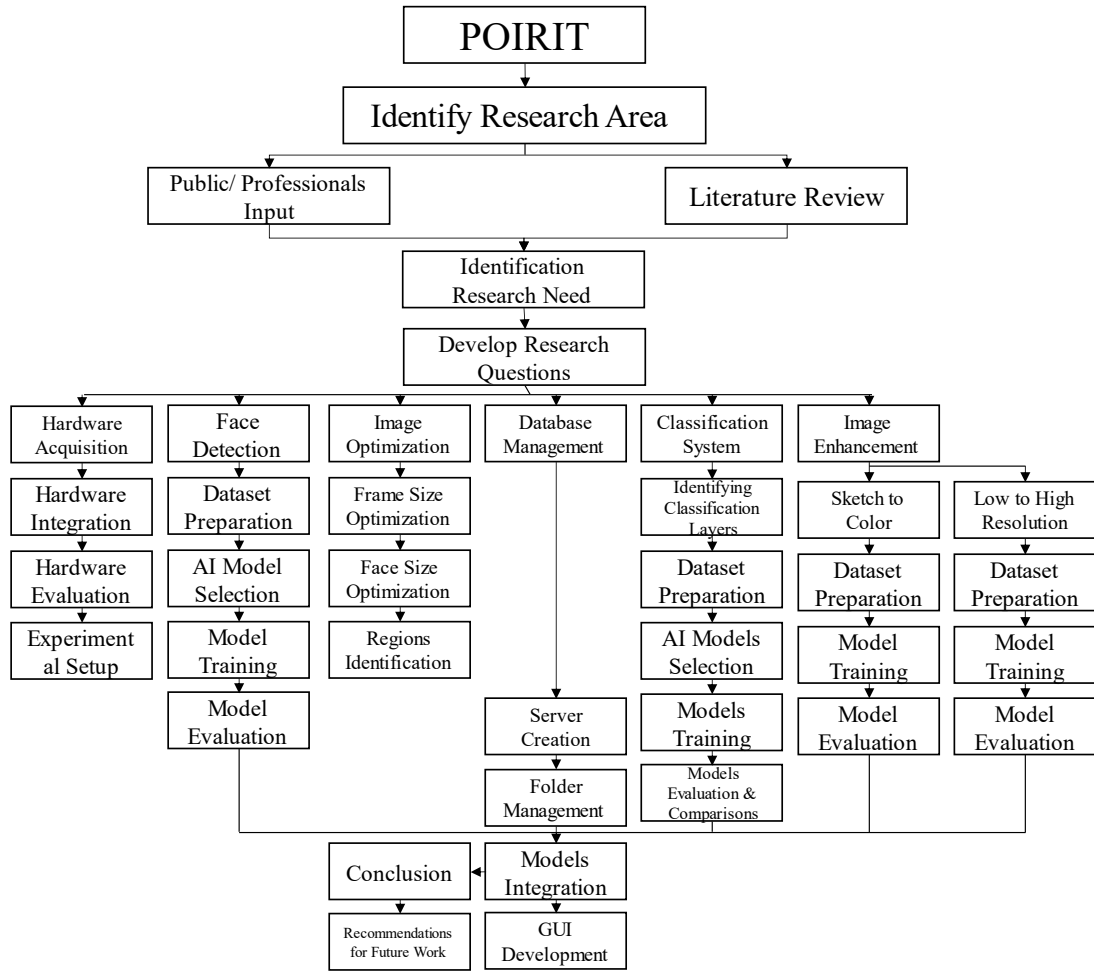


Figure 6 Research Methodology Flowchart

The process shows the detailed research steps in the form of a flow chart carried out to complete the research plan. It highlights the step-by-step method to achieve the target of this thesis and the notable stages of the research project. Initially, area of the research was identified with the help of multiple discussions with the supervisor and reading about modern trends in engineering. Being a citizen of a country like Pakistan, the major concerns of the region were highlighted, including security, health, education etc. Once it was decided that research would be conducted to solve security issues with the help of artificial intelligence, stakeholders were contacted, and the literature was reviewed. With the help of the above activities, research questions were developed. Hardware requirements were identified and managed via a selection of the cameras, Raspberry Pi, GPUs, etc., and finally the whole experimental setup. The face detection problem was resolved by identifying the scenario for face detection that is

CCTV footage, and then selecting and training the appropriate AI algorithm supported by the literature on the custom datasets. To optimize the time and space complexity of the system, calculations were performed for the resizing of the frame and face. Moreover, the region of the interest from face to be stored in the database was calculated using the literature. To make the system efficient in terms of search time, classification layers were selected specially for sub-continent regions. The dataset was gathered and trained on various models. Image enhancement techniques were studied and developed solutions for a couple of scenarios. Experimental results for all the models were evaluated and the best performers were selected. The central database was designed to handle images and Graphical User Interface (GUI) was designed to give users an interface to communicate. Finally, a conclusion was made and recommendations for future work were given to new researchers.

I. Literature Review & Thesis Write-up

The literature review begins with a review of fundamental image processing techniques and applications that are further built upon in this work. The review focuses on algorithms that were identified as particularly useful for solving problems inherent in this research topic and explores a variety of mathematical models that are used in the field of image recognition. The literature review then progresses into the subject of machine learning, and deep learning, identifying methods that are used together with image recognition techniques for efficient recognition of human faces from a variety of different angles. A literature review is updated along with the research.

II. Organizational Visits

This research is focused Punjab Safe Cities Project, in this regard meetings were held with the officials of PSCA for a better understanding of the current system installed and its capabilities. Potential research areas were outlined and the opportunities for new research questions. Their system was studied, and data was gathered. It was found that no recognition system is present, and it is not integrated with the National Database & Registration Authority (NADRA) system. Pakistan Forensic Science Agency (PFSA) was also visited to observe their systems. Officials involved in creating sketches for the criminals were also interviewed to create a scheme for the identification of targets.

III. Image Acquisition & Hardware Integration

First and foremost, the research delves into devising a methodology for acquiring images from diverse cameras located at various geographical locations using distributed computing devices connected to a network. It was essential to decide what methods would be used for gathering the images from different resolution CCTV cameras and processing them for uniform target recognition. It includes the selection and configuration of the hardware involved. This is also concentrated on finding a way to avoid the repetition of frames with the same information without involving the computing device. In response to this objective, hardware and software tools were procured. The first phase for hardware included procurement of Raspberry Pi, IP camera, USB camera (only for test purposes), Video capturing device, network switch, and router. In the second phase for hardware, a Laptop with Nvidia GPU and Passive Infrared Array (PIR) sensor were added. PIR sensors connected to the computing device is responsible for generating a trigger if any change in the infrared is observed. Such sub-systems were integrated to form a network.

IV. Optimization by ‘Frame’ and ‘Face’ Resizing

Subsequently, this research aims to investigate techniques that ensure accurate detection of human faces within CCTV images, emphasizing precision and reliability. Additionally, strategies for resizing images from CCTV cameras of varying resolutions are examined to strike a balance between computational efficiency and performance. The system requires image resizing before uploading to the database for the recognition step. This step figures out the frame and face size calculation methods in such a way that reduction does not affect the performance of the system. It also focuses on establishing rules for selecting the area to be cropped from the frame and stored in the database. It requires a thorough examination of the optimal region encompassing the face for cropping and storage, retaining all essential features. Establishing a centralized database tailored for storing cropped images along with their pertinent attributes is deemed a crucial component of the solution. It is linked with the fourth objective; only the relevant areas to be considered for the classification layers. For the recognition stage, precise calculations were required for the optimized input layer for training as well as for the recognition. Details are presented in the System

Design chapter. Different pre-processing operations for face cropping were performed at this stage. It must be noted here that the images cropped out from the frames had already passed through the pre-processing stage of the detection stage for illumination, hue, saturation, etc. So, less computations are required at this stage for such adjustments.

V. Face Detection from CCTV Images and Pre-Processing

Detecting images from CCTV cameras can be a challenge for the system. The third objective states what algorithm(s) to be used to train the model to detect small-sized faces from a CCTV frame at high speeds. CCTV images with variations of lighting and poses were obtained to train a model for small-sized face detection. Details regarding datasets and their preparation are given in Chapter 4. The experimentation process had started at an early stage. Input image frames for the detection system differ in terms of illumination, sharpness, or contrast. When experiments were performed on a variety of inputs, the results were not suitable. To cope with this situation, a feature was required that can handle the variance of the CCTV image parameters. In this regard image pre-processing/transformation stage was designed. All the image processing performed in the scope is grayscale. It must be noted that pre-processing is performed on frames for face detection as well as for the training images. Viola Jones algorithm is used to create automated regions of interest for training face detection models. Amendments were made manually for any errors before training a convolutional neural network. Resizing was performed to adjust with network input layers.

MATLAB 2018, 2019 & 2021 were acquired at various stages. Later Python along with various frameworks was used. Additional toolboxes and hardware support packages were also required for the research and were installed. Octave GNU was also used Initial codes for the image acquisition nodes were implemented on Octave after being tested on MATLAB. So, it can be used on Raspberry Pi.

VI. Development of Classification Systems using Deep Learning Models

Furthermore, the research evaluates and selects cascaded classification layers to construct an efficient categorization system specifically designed for the Indian subcontinent. Meticulous methodologies for dataset preparation are explored, recognizing it as a fundamental step in training deep learning models effectively. The project also involves the assessment and selection of an appropriate deep-learning algorithm to underpin the recognition system. Face classification is of paramount importance for this research which makes it unique from others. Major optimization is to be performed by classifying the images into categories. This points out what possible layers can be cascaded for the Indian subcontinent region with the help of available literature. It was decided to have three levels of classification stages i.e., Gender, Face Type (Ethnicity), and Facial Hair Style. There are many ways for face recognition to be done using deep learning. These algorithms show different results for various types of data. With the carefully collected/ prepared dataset, various convolutional neural network architectures were trained using transfer learning. Datasets of all three categories for target classification i.e., gender, face type (ethnicity), and facial hair style, and for face detection were collected initially with 10000 images per gender and 4000 images per ethnicity and facial hair type. This was one of the biggest challenges of the research to prepare a dataset for each layer. Open-source images were collected and then refined for training purposes. Finally, datasets are made using an enormous number of training images. All the images were processed through an indigenous code to maintain uniformity in training. All the images were manually checked before passing to the next step. Deep learning models were identified to be used for classification. Different models are to be evaluated for the same dataset. Deep learning tools in MATLAB and Python are used to develop models for the classification. Evaluation of the trained models is done based on established parameters including confusion matrix, accuracy, precision, recall, specificity, F1 score, Precision-Recall curve, and Receiver Operating Characteristics (ROC) curve.

Image processing is a complex and processor-intensive computational task. It involves very costly computational repetitive operations. When these processes are

done with CPU, they take much longer time. To overcome this problem parallel computing is performed with the help of a general-purpose Graphical Processing Unit (GPU). Training of deep learning models on GPGPUs can reduce time to 10 times than the training on CPU. A framework called TensorFlow-GPU was used for the implementation of a deep learning algorithm on GPU. It helps to calculate numerous mathematical expressions simple and computational algebra is optimized. A GPU-accelerated library of deep neural networks described as NVIDIA CUDA Deep Neural Network (cuDNN) is used along with the CUDA Toolkit to use TensorFlow-GPU features. It has professional versions of functions that typically appear in deep neural network applications.

VII. AI Models for Image Enhancement

The goal of this research objective was to develop GAN-based AI models that can handle image enhancement problems in a variety of real-world scenarios. In one such case, crimes are reported with an unidentified perpetrator and no video evidence, which leads to the victims' or witnesses' assistance in drawing sketches. It can be challenging for computer vision systems and humans alike to match these sketches with real faces. To get around this, the study estimates a colour image from a sketch using MS-Cycle GAN. The project also addresses the problem of extracting detailed images from CCTV cameras that might not have enough distinguishing characteristics. The developed system functions as an artificial support system for monitoring in addition to facilitating the estimation of a high-resolution image from a low-resolution one. With the help of this technique, images can be enhanced in a variety of ways, overcoming issues like motion blur and low resolution.

VIII. Database Management & GUI Development

Finally, considerations are made for how different users will gain access to the database and conduct searches for individuals of interest, ensuring a user-friendly and efficient interface. This objective needs the development of a system to store the incoming CCTV frames and cropped faces in the relevant folders at the local host. How a local web server is created for testing and deployment. This database stores CCTV frames as well as cropped face images processed by the previous stage. For the human-computer interface, a software dashboard was required for the system to upload

query images and other search options Python models are integrated using Tkinter. Once completed, this portal is accessible to law enforcement agencies and the developer for the sake of maintenance purposes. This portal communicates between the users and the main database stored on the server. Allowing access to the face image records and the recognition records. It is designed in such a way to be user-friendly and provide all necessary search options like Camera ID/Camera Location, Time of event, Date of event, ID, and Recognition result search. This design follows the design of the management system used by the Punjab police. Visits to relevant authorities like the Punjab Safe Cities Authority, Punjab Forensic Agency, and NADRA (National Database & Registration Authority) were conducted. Necessary use cases of the system were designated. The following diagram, Figure 7 shows the use case diagram of the system. Admin has the right to configure the system and manage the database. Whereas the user, potentially law enforcement agents can search the targets from the database. Camera nodes write the frames to the database and those are accessible to the face detection system.

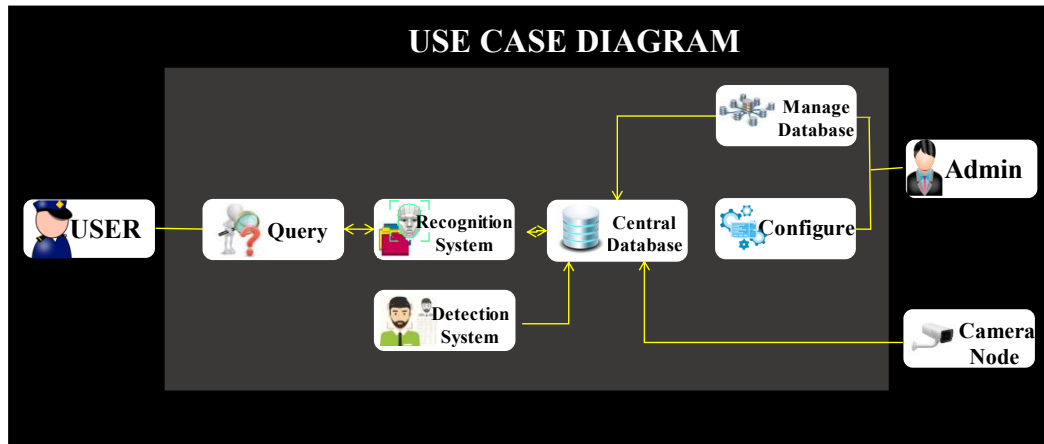


Figure 7 Use-Case Diagram of Database Management System

Various image input/transformation options as per the real-world scenarios are identified. The recognition system determines the category of the input image and then searches from the relevant folder of the user-entered location and time. Targets can be tracked among the nodes using the nearest neighbour algorithm for this feature. A detailed explanation of the research queries and the techniques that are considered to get the necessary results are discussed in the system design section.

1.05 Organisation of Thesis

This research contributed to developing a system for tracking and searching desired targets from a network of CCTV cameras. The research followed a sequence of phases, over which parts of the system were developed and tested. Tasks involved were a review of relevant work from the literature, current practices were evaluated, and relevant bodies were surveyed. Ethical parameters were judged and the scope for practical implementation of this research in a specific and controllable manner was gauged. The chapters in this thesis are arranged as follows:

Chapter 2 presents the literature overview of image processing and its classification. It also discusses applications of image processing in security. Highlights of techniques used in biometric systems. Then it focuses on methods of face detection and recognition. Then it outlines machine & deep learning. An overview of Parallel Computing and Intelligent Assistance Systems is also given in this chapter. It concludes by justifying the need for an advanced system for tracking persons of interest.

In Chapter 3, the methodology adapted for the development of the systems is discussed briefly as a detailed design is given in a later chapter.

Chapter 4, system design elaborates each block of the system in a detailed manner. It includes a comprehensive insight into the acquisition systems and the protocols for acquiring data. Then the pre-processing stage is explained to increase the face detection rate from the frame. Hardware details of the front/capturing node are also debated. Attainment of privacy by excluding human involvement is proved in this chapter too. Then the main computing system is described. It includes the hardware system as well as the software part. Training datasets are discussed and screenshots for training sequences as well as results are shared. All the stages of the recognition system are explained in detail with sample data. The second pre-processing stage that is used for recognition is described. Algorithms developed for each sub-system are discussed.

Chapter 5 focuses on the development of the experimental setup and integration of all the sub-systems. It describes the protocols followed between these systems. This

section also indicates the sample subjects for the experiments and the procedure involved in setting up the different scenarios for conducting these experiments.

Chapter 6 shows the results for various deep learning algorithms and calculations are shown to justify the significance of the POIROT. Precision, Accuracy, Recall, and F1 Score for different training models are compared. All the different cascaded stages' results are presented and mathematically proven efficient.

Lastly, Chapter 7 concludes this thesis by summarizing and presenting significant findings. Frame rate and resizing results are highlighted, in addition to this, face detection algorithm selection is overviewed. Findings for image transformation and enhancement features are discussed. Face region selection and cropping and uploading to the database are briefly presented. Classification layer selection along with the deep learning algorithms to train the models is recapped. Deep Learning algorithms used for recognition systems along with the accuracies are presented. The last section provides recommendations for future research. More efficient and recently introduced computing devices are suggested to new researchers. Additional classification layers are also recommended to sort targets into more categories. Lastly, newly launched relevant algorithms are also indicated to achieve better results.

1.06 Scientific Contributions

Particularly in the context of urban environments, the research project significantly advances the fields of intelligent video surveillance and facial recognition systems. It tackles several important research questions about face detection in images, cropping relevant face regions, resizing images for computational efficiency, creating databases, and multi-layered classification for demographic classification. The suggested framework is more efficient when a distributed computing system is implemented using devices like GPUs and Raspberry Pis. Notably, the system broadens the range of applications by introducing innovative features like image enhancement, fog removal, and sketch-to-colour image conversion. The system's versatility is further demonstrated by its ability to accommodate different ethnicities and by the possibility of adding more features and extending age classification. The suggested framework establishes the groundwork for models of collaborative computing, which could completely transform the way publicly owned resources are used for the good of

society. All things considered, this research project offers creative ways to improve public safety and security in urban settings while showcasing a comprehensive approach to intelligent video surveillance.

1.07 Limitations of the Research

The research is primarily designed for the demographic landscape of Pakistan. The system's applicability may be constrained in regions with different ethnic compositions or having uniform ethnicities like China, Japan, Nigeria, etc. The current classification capabilities of the system are limited to two genders, four ethnicities, and four levels of facial hair. This may not cover the full range of diversity present in all populations in other parts of the world, which may lead to misclassifications. The system's current parameters cannot account for transgender individuals. Adaptations would be necessary to ensure accuracy and relevance in diverse cultural settings. Another issue is that the age-related facial features may vary significantly across different ethnicities. Classification of facial features may be influenced by changes in appearance over time which may not always align with the initial classifications. This system will not be able to track a person using an old record picture. Facial hair styles can be changed intentionally to fool the system. The world has seen a pandemic where people were forced to wear masks for public safety. The system proposed is categorically designed for facial features. This could limit the system's effectiveness in environments where mask-wearing is prevalent. Variability in mask types of fits, and occlusions may lead to reduced accuracy in facial attribute classification, particularly in scenarios where mask-wearing is prevalent. The system's performance in recognizing individuals with artificial beards and hair is yet another facet that requires attention. Artificial enhancements can introduce additional complexities, potentially leading to misclassifications or false positives in facial attribute classification. The system's ability to process video streams in real-time is contingent on the computational capabilities of the hardware. Lower-end or older devices may struggle to meet these demands, potentially limiting the system's effectiveness. This system is dependent on hardware that meets the minimum capabilities outlined in the system design. This could limit its implementation in situations where the required hardware may not be readily available. The research evaluates various neural network architectures, it

primarily focuses on specific models (AlexNet, GoogLeNet, ResNet, VGG-Face). The generalization of findings to other networks would require thorough evaluation and validation. The system may be impacted by local laws regarding the collection of video footage. While the system is designed for a region without such restrictions, its applicability in areas with different legal frameworks may need to be carefully assessed. Sketch to face generation system is limited to professional sketch images only. This may not accurately represent or generate faces from non-professionally produced sketches, potentially reducing its applicability in certain contexts. It is important to acknowledge these limitations when considering the applicability and potential deployment of the proposed system, and to conduct further research and testing to address these constraints.

1.08 Summary

. The goal of this research is to create an intelligent assistance system that can integrate with the Punjab Safe Cities Project's existing CCTV camera network with the minimum hardware additions, effectively search and track targets by classifying the public, and maintain individual privacy by allowing monitoring staff to only view the target frame. Additionally, this approach can help humans to predict the colour image from hand-drawn sketches, enhance blurry or low-quality photos resolution, and forecast the frontal face from a side pose image. Problems related to CCTV camera surveillance were identified and understand the challenges of automated target detection and tracking systems in regions like Pakistan. Once the understanding was developed, and then an intelligent assistance-based solution was proposed to address the challenges in tracking persons of interest in a network of cameras by keeping in view the computational cost constraints for real-time image processing. This new solution uses the demographic diversity of Pakistan and proposes classification layers of recognition in a cascaded manner to decrease the search time. To attain the goal of the research, in the first step, images from surveillance cameras are fed to Raspberry Pi computing nodes for frame capturing and image pre-processing. Sharing the processing between multiple Raspberry Pi's increases the processing speeds by leveraging parallel computing. This sub-system is responsible for picking out the frames and writing the data on the server database that manages the unidentified faces.

From these frames, unidentified faces are stored in the respective location for the camera at each geographical location, date, and time by the next processing stage. These images are then accessed by the parallel high-performance machines that run the face recognition system on GPUs connected via a cloud server. NVIDIA GPUs provide multiple cores for processing making the system more time efficient. The system uses deep learning models for classifying gender, face type, and facial hair. It helps the search algorithm more time efficient. Targets are searched in the relevant category that are already sorted by the system. The server keeps the database for tracking Face ID, Camera ID Time, and Date. Moreover, unprocessed data is also kept on the server. Finally, face images are sorted after comparison with the central data containing faces of citizens or target images (as per search criteria). Following block diagram, Figure 8, shows the hardware setup.

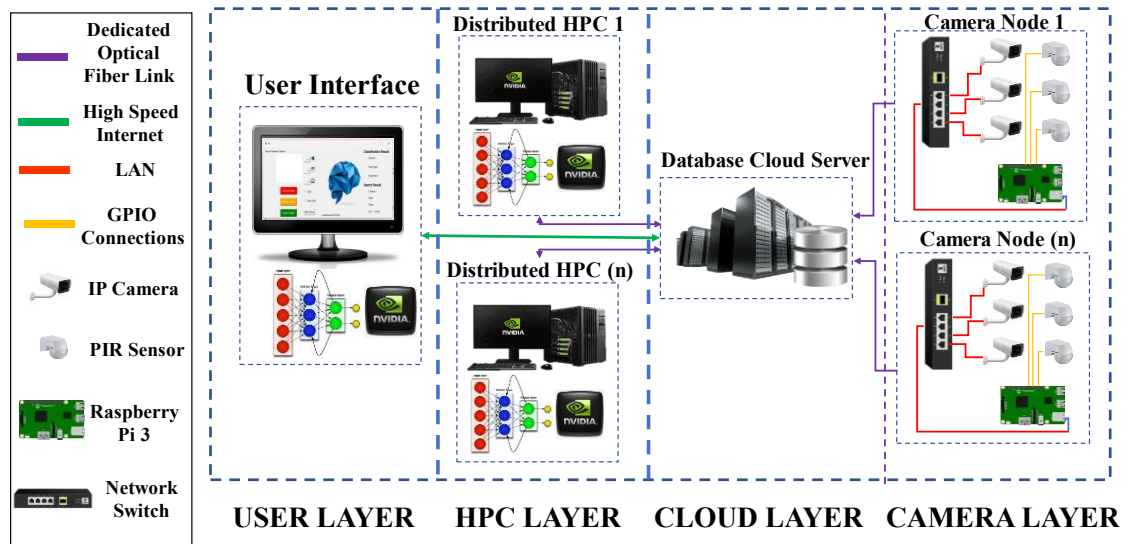


Figure 8 Hardware Block Diagram

The above arrangement shows that multiple processing nodes work together in the system, breaking down the tasks to provide efficient face detection and recognition.

Chapter 2

Literature Review

2.01 Overview of Literature Review

To build the theory and find the solution to the research problems, a comprehensive literature review was performed. Access to journals and IEEE transactions was available through the University of Cape Town and the University of the Punjab servers. This chapter presents the literature in the fields of image processing, artificial intelligence, and parallel computing. It starts with the background of the image processing and then explains the classification of the image processing. As elaborated in the earlier section this research focuses on security using visuals from the CCTV camera, therefore the application of image processing in the field of security. Biometric systems are briefly reviewed as well. Then it discusses the methods used for face detection and face recognition. The machine learning overview is then discussed. After that deep learning overview along with the techniques of deep learning are reviewed in this section. Supervised, unsupervised, reinforcement, and semi-supervised learning in the context of deep learning is explained with the help of literature. Models of deep learning including Artificial Neural Networks (ANN), Recurrent Neural Networks (RNN), (Convolution Neural Networks) CNN, Recursive Neural Networks, Generative Adversarial Networks (GAN), and Auto-Encoders are then reviewed. Frameworks that are relevant to the research including TensorFlow, PyTorch, Keras, Caffe, and CNTK are discussed. Applications of deep learning in the field of image processing are briefly overviewed. This research includes many computations that are not possible for a single CPU to perform, which is why parallel and distributed computing were required. Literature relevant to parallel computing and platforms used for distributed computing like Raspberry Pi, NVIDIA Jetson, Intel Compute Stick, and HiKey are also included in the discussion. Finally, a brief overview of the intelligent assistance system is presented to conclude the review.

2.02 Overview of Image Processing

Humans have five basic senses, namely: hearing, taste, touch, smell, and vision. Vision is one of the strongest and most authentic senses. In earlier times, the eye was the only component responsible for vision; however, nowadays it is understood that the eye is just one part of our highly complex vision system. The images we obtain from the eye are just one small part of the system – the analysis of these images from the eye, from which we make decisions and act, is done by the brain. The image ‘written’ on the screen (i.e., the retina) inside the eye. Similarly, colours are detected by the brain instead of the eye [39]. Words recognition on the page, a cup on the table, and the person entering the room seem to be an easy task. This ease of visual recognition skill contradicts the actual height of computational complexity of the process behind [40]. Face recognition is facilitated by complex neural systems present in humans. It consists of various interconnected regions. Its functional arrangement is responsible for distinguishing between invariant features of faces which is fundamental to face recognition. Moreover, variable aspects like expressions, movement of lips, eye winks, etc. are also processed by this network [41]. Subcortical structures are involved in recognizing facial expressions [42]. The recognition system can be harmed if the brain is damaged [43]. More tasks other than object recognition are performed by vision systems like object tracking, avoidance, segmentation, etc [44]. Similarly, the brain can recognize different shapes, colours, structures, patterns, and so on. The human brain is an expert in recognizing faces, even able to accurately recognize a person from many different angles but the person needs to be known, and the better known the person the more accurate the recognition and wider range of angles that the person can be recognized from.

Digital image processing usually processes the two-dimensional image using the computer [45]. The term ‘image processing’ on its own is very general – it could mean all sorts of processing done by computers on any sort of image. We can observe advancements in the image processing area every week. The increase in affordable computers and internet access caused information flow to individuals making technologies available to pockets and purses [46]. With the advancement of computer vision technologies, computer-based vision systems have progressed from the early

systems of recognizing shapes and following lines to considerably more advanced systems that are getting increasingly closer to achieving the level of image recognition and understanding that humans are capable of. Image processing can be used in many types of applications, such as medical imaging, photography, image and video editing, security, and artificial intelligence applications [47]. The recent decade has seen a fast spread of video imaging devices in various fields of life. Many video processing tasks require activity recognition in the video stream especially in the field of security, surveillance, entertainment, etc [48]. In some cases, computers have far surpassed human abilities in making image-based decisions, such as the rapid speed at which factory line inspection systems can detect tiny flaws in products within sub-millisecond timing constraints. Thus, there are elevated expectations for what computer-based image processing systems will be capable of in the future.

2.03 Classification of Image Processing

There are numerous uses of image processing. For instance, it is used in astronomy for satellite and telescope imaging for interplanetary images. Medical image processing in Magnetic Resonance Imaging (MRI), Computed Tomography (CT)-scan or X-ray images, Pattern or character recognition in number plates, sign board detection, security is also done with the help of techniques like face, fingerprint, or iris recognition, industries use image processing techniques commonly in processes like product inspection and sorting. Photo editing tools in computers and smartphones like Photoshop or PICASA are also applications of image processing [45].

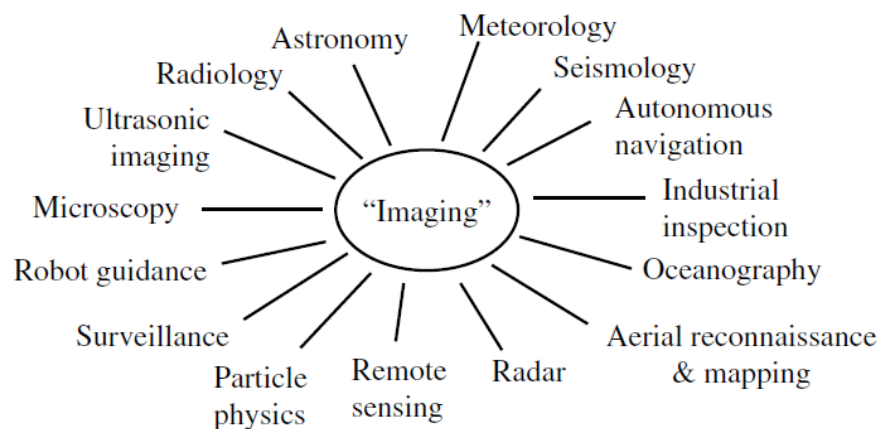


Figure 9 Image Processing Applications [Source: Bovik, A.C., 2010] [45]

Image processing is not only used on visible wavelengths. It is also used for other bands. Figure 9 shows applications of image processing in different fields of science.

Gamma rays imaging is used in nuclear medicine and astronomical analysis. In nuclear medicine imaging, a radioactive isotope is injected into the patient's body and when it decays it emits gamma rays. These gamma rays are detected by the detectors to observe disorders. In the case of astronomy natural gamma rays emitted by the cosmos are detected. These images are fed into high-performance computers for analysis [49].

X-ray imaging is done by bombarding X-rays from a controlled X-ray tube to a sensitive screen. An object placed between the emitter and screen can be observed with variation of intensities of x-rays. The screen has been digitalized to translate the images in terms of computer-readable signals. Computerized axial tomography (CAT) is another application of X-ray imaging in which 3-D imaging is done with the help of combining many image slices. Human bodies as well as other materials are inspected using x-rays. Angiography is also done with the same principle with the addition of a wire inside the arteries [50].

Ultraviolet ray imaging is used in fluorescence microscopy. It is done by colliding the ultraviolet photon with material to excite its electron to shift in the next energy band. When it loses its gained energy, it emits light. That can be read by the detectors [51].

Radio Detection and Ranging (RADAR) is the most common application of microwave imaging. In visible light, imaging is done by the reflection of the light waves from the object whereas in microwave imaging microwaves get reflected from the objects and read by the detectors to create an image. Like in cloudy weather where the human eye cannot see beyond, microwaves penetrate through the clouds and come after reflection. Different signatures are observed by different types of objects. These waves are also used for observing underground or undersea [52].

Radio band imaging is used in medicine and astronomy. MRI (Magnetic Resonance imaging) is also an application of radio band imaging. In MRI body is placed in a powerful magnetic field and radio waves are passed through the body.

Responding pulses by the tissues of the body are monitored by specialized computes that construct a 2-D image of the body [53].

Similarly, industries also use visual range image processing techniques to lessen human efforts. Many industries use image processing in level checking while bottle filling. Image processing is used by modern number plate detection systems. The camera takes a picture of the desired vehicle and sends it to the computer system to read its plate-like humans. Image processing is also used in surveillance systems. With the emergence of the threats of terrorism in the world, these systems are getting smarter day by day.

2.04 Applications of Image Processing in Security

Image processing is used in defence and security applications including missile navigation, target detection and its tracking, object or human detection, and surveillance. The main purpose of the above-discussed applications is to reduce human involvement, resulting in fewer errors and an increase in processing capabilities by volume and quality. The main goal is to develop systems to replicate human capabilities for detection and capable of making decisions using sensors [54].

Digital watermarking is a technique to hide information in the data using some algorithm for the sake of copyrights and ownership claims. For an image, some data is a hidden image. It is done by adding some secret information to the image. A similar is done to videos for piracy checks. The watermark in the image can be invisible or visible depending upon the use [55]. Information hiding is the communication of information, but it is hidden. This information can be audio, video, image, or text. Information hiding is done in three ways, watermarking, steganography, and cryptography [56].

Biometric identifiers are distinguishable property of persons used to identify them [57]. Biometric identifiers are considered physical versus behavioural features [58]. This includes physical properties of the body like face, fingerprint, palm print and veins, iris pattern, retina, scent, etc. Voice, gait, and typing rhythm are behavioural characteristics [59]. Face recognition is the most common area in biometric research. Surveillance cameras can be found in abundance in public places nowadays. Face detection has many advantages over other systems [60]. Face detection uses many

methods that will be discussed later. Generally, an image is taken as an input in the computer system. Some software converts the colour system as per requirement. It also resizes the image to make it convenient to process [61]. Eye sockets, chin, and eye contours are captured using the three-dimensional imaging sensors in new technology [62].

Target recognition is a key requirement in military applications and concerns one of the main objectives of this thesis as discussed in the objective. Algorithms for object classification and detection are highly efficient in video and image processing. These systems are also able to detect humans, vehicles, animals, etc. That is important for resolve targets from surroundings. It requires very robust algorithms that can work efficiently in all types of conditions. The same object appears differently by varying conditions that make the task more difficult [63]. There are two basic things in auto target detection, one is targeting detection and other is target recognition [64] [65]. Discusses that for number of years researchers worked in auto target detection in lidar imagery [66]. Region of interest (ROI) is the first thing that is considered to find if a target is present in this region. Then the target is detected from the ROI by the efficient algorithms [67].

Remote sensing is the area of science that is made to look beyond the sight range. Imaging is done from aircraft or satellites. It is used to measure and observe the properties of the earth's surface [68]. In the field of security, it is used in intelligence, strategy tactics, and military applications [69]. Besides security it is also used in environmental, geographical, geological, and weather information systems [68]. First thing in the process is data gathering that is done with the help of high resolution multichannel, multispectral scanner (MSS). Data is sent to earth for processing. Images are corrected by removing any irregularity caused by the sensors. Satellite altitude and attitude changes, skewing, etc. causes geometric errors that are also removed. Furthermore, sharpness, edges, contrast, etc. are also corrected with the help of image processing tools [70].

Biometric system is a pattern recognition system that can identify patterns of human body parts. Firstly, it extracts the feature set from the input data and then it compares it to the main database to give the result. The biometric system involves

some sensors to gather data [71]. Identification is required to perform variety of services. It is done to ensure that legitimate user is accessing service like Automated Teller Machine (ATM), cell phone, laptop, etc [72]. There are few factors involved to ensure the suitability of biometric system for a certain task. *Universality*: Everyone should have that specific characteristic that is being used to identify. *Uniqueness*: Characteristic should be distinguishable across the individuals. *Permanence*: The characteristics should remain invariant over longer period. *Measurability*: Characteristics should be acquired by some sensors and could be digitalized. *Performance*: This should be accurate *Acceptability*: People using the system should be willing to share their biometric patterns. *Circumvention*: Characteristics should not be easily imitated to make fake identifications [73]. Biometrics is evolving as one of the most important fields now a days due to increased threats of terrorism. It also excludes the old authenticating procedures like smart cards, PINs, keys, and passwords [74]. Physiological characteristics like face, fingerprints, finger/hand geometry, voice, retina, and iris are biometric identifiers. Behavioural characteristics include signature, typing style, and posture [75].

A recently secured face recognition system for the electronic voting system has been developed that uses Local Binary Pattern (LBP) for the characterization of face features and chi-square distribution for image classification. This low-cost system uses smart phones and web applications for learning [76]. Internet communication traffic for secured systems and targeted individuals uses auto face recognition systems that monitors the individuals appearing in the images and track their IP addresses [77]. Facial features can be obstructed due to sunglasses, scarves, etc. as shown in Figure 10, and can be hard to detect. In security-related applications scenarios with face occlusion are common making it less reliable. Such problems can be countered by Linear Recognition Classification giving better results [78].



Figure 10 Face Occlusion Examples [Source: Naseem, 2010] [79]

Few modern trackers use advanced 3D face models that match local features or facial textures [80]. In recent, social networking has become part of daily life. Law enforcement agencies has also been involved in monitoring the traffic. As shown in Figure 11, Face Tagging in social networks is used to track the missing persons using Big Data [81].

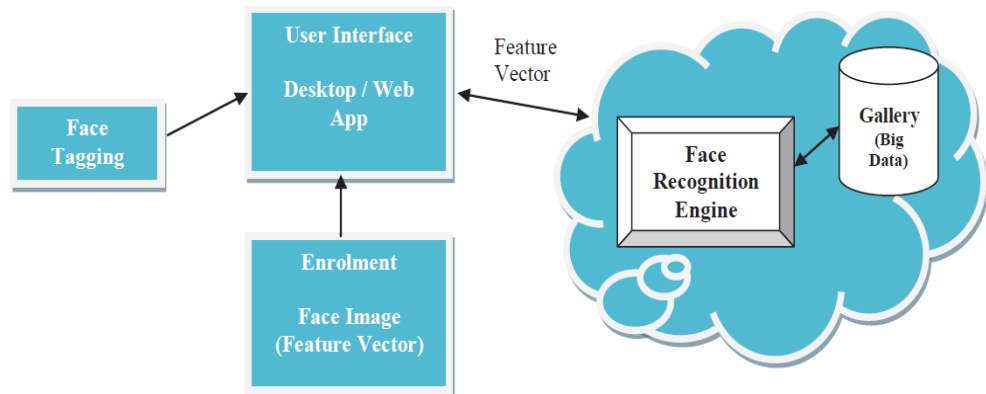


Figure 11 Proposed Face Detection by [Source: Vinay, A.et. al.,2015] [81]

2.05 Techniques used in Biometric Systems

Face recognition is an important part of our lives. We easily see and recognize people we know – which is an essential skill in our lives; for instance, we recognize people we know and trust. Most of us who have reasonable eyesight don't feel any difficulty while performing such a complicated task. Face recognition systems can also perform these tasks using algorithms and databases that store features that represent recognized faces [82]. Face recognition has many advantages over other biometric methods. Many other techniques require some contribution from the user. One must put a hand or finger for fingerprint and palm pattern recognition. Must look into the camera for iris and retina scan. Modern face recognition systems can detect individuals

from far away without their active participation. That gives it an edge in the field of security and surveillance [73].

There are a few steps involved in every face detection system. Face recognition is effectively a type of pattern recognition system. It extracts visual features and matches these with features that are stored in a database. These key features must appear on all faces. For us, a desirable system is capable of detecting faces with variation of poses, illumination, presentation, and expression [81]. Few are explained in detail in the next section. Face recognition is a difficult example of object detection and recognition. The difficulty occurs when faces change poses [83]. There are many algorithms used to detect faces. A few of them will be discussed below.

Ear-based recognition is a new area of biometrics. The ear is a part of the body that does not change its shape with age [84]. The ear is larger than fingerprints and smaller than the face, which can make it easier to capture or to capture from a distance. However, there are problems with ear-based recognition, such as when the ear is covered with hairs or caps [85]. An ear image database can be built manually by cropping ears from different images of varied sizes. However, such templates are likely not sufficient to cope with all scenarios where this type of recognition would likely be used [86]. Specialized algorithms are used to handle difficult situations [84].

Human face can exhibit different kind of expressions as well as poses. The face is a 3-D entity that can vary by a change in lighting conditions and poses. It also changes with age [87]. It is observed that one of the best face recognition systems shows high error rates when a same person's image is taken one year apart [88].

Iris detection has an edge that humans have different iris patterns and remains same over the time. It has distinctive features like zigzag collarete, corona, ridges, ligaments, freckles, rings, furrows, and crypts [89]. Colour is also distinguishable; it is amount of melanin pigment [90]. Iris biometric is different from retina scan. Iris is easily visible from normal cameras [91]. Where retina scan requires visible or infrared beam to illuminate retina [92].

Fingerprint is a pattern made on fingertips in the form of ridges and valleys. It is established in the first seven months during the development of the embryo. It has uniqueness and even identical twins have different fingerprints. Even different fingers

show different prints of the same person [93]. Fingerprints are one of the most reliable biometric indicators [94]. It is used for identifications for access as well as for identification of criminals [95]. Normal fingerprint scan is done with the help of sensor. It reads the pattern in the form of analogue signal, and it is converted to digital signal with the help of analogue to digital converter. Offline fingerprint acquisition is also done where the scanner is not available at the instance [96]. Fingerprints images are also being taken with the help of cameras providing cheaper solution avoiding dedicated sensors [97].

2.06 Methods of Face Detection and Recognition

In the process of face recognition, the first step is to detect a face from the image, or we can call it finding faces in an image. It can be said in another way that finding out faces. Face Detection is a normal and quite an easy job for humans. But for the machine to recognize the faces in a picture is not that easy. Things that make it difficult for the machine to detect the face are profile pose, tilted pose, expressions of the face, hairs, low resolution, blurred images, etc. So, the first step of Face Recognition is Face Detection [98]. Face detection can be divided into two parts, the first is to find if there is any face present in the picture and the other is to find the location of the face if it is present [99].

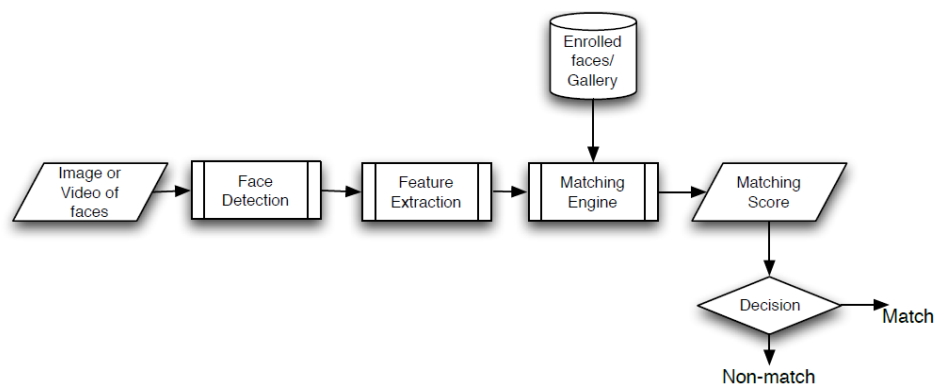


Figure 12 Basic Flow of Recognition System [Source: Rawlinson, T et. al., 2009] [82]

The sequence of the flow of face recognition from acquisition to final matching is shown in Figure 12.

Feature-based approaches are used in different classification systems but their way of using the features is different. We can classify them into different approaches. Feature-based, Geometry-, based, appearance-based, Haar-like Features, etc.

Feature-based approach banks on the facial features' extraction for face detection. In low-level analysis, pixels are evaluated. Edges are detected using Sobel or Canny filters. It is done in greyscale. Sobel focuses on high spatial frequency regions of the image that describe edges. The approximate absolute gradient magnitude is found at every pixel of the greyscale image. The operator contains a pair of 3×3 convolution kernels as shown in Figure 13. Both are 90° rotated versions of each other [100].

-1	-2	-1	-1	0	+1
0	0	0	-2	0	+2
+1	+2	+1	-1	0	+1
Gx			Gy		

Figure 13 Sobel Convolution Kernels [Adapted from Vincent et. al. 2009] [100]

They respond extremely to edges in the image as per the pixel grid. Both can be applied to the image separately to give a gradient component in Gx and Gy orientation. Local maxima are found to detect the nose from the image and local minima are for finding eyebrows, pupils, and lips. Feature analysis improves the results of low-level analysis by incorporating relative positions of eyes, nose, mouth, chin, head-top, etc. to each other. If these features are present at the correct position, then it means there is a face [101].

Geometry-based detection also uses edge detection with the help of a Canny filter and gradient analysis. It divides the predominant features of the image into blocks. All these blocks have a pixel centre, and these are connected to nearby central pixels to span the face [102]. In mathematical terms, the gradient analysis method finds edges by finding the minimum and maximum in the first derivative of the image. Image gradient along two orthogonal axes is calculated. Classical operators are used to find the first directional derivative operation [103]. In the Canny filter, this almost remains the same. Gaussian filter is applied for smoothening of the image.

The appearance-based approach works with facial extraction to detect faces. A two-dimensional image processing is done. Extracted characteristics are known as features. These features are matched with the human facial features like nose, eyes, mouth, etc.). Principal Component Analysis (PCA) and Independent Component Analysis (ICA) are used to extract feature vectors. PCA is more efficient in terms of time and space complexity as it only deals with dominant features. It is done by adding training faces. These images of faces in the database are transformed into a vector and placed into a training set. Every image is transformed into a vector and placed into the set. Face images result in a point in dimensional space. An ensemble of images, then, maps to a collection of points in this huge space. Now the average vector is subtracted from the original image. PCA is used to explain large dimensional space with a small set of vectors [104].

Eigenvectors are eigenfaces in our context. Eigenvectors having small eigenvalues are not considered because they will show important characteristic features of the faces. Training samples are added to the Eigenface space. The weight for each feature from the training images is calculated.

After training the next step is recognition. Mathematically this process is finding the minimum Euclidean distance between the test and training point. This difference between the two vectors expresses the similarity between the images [105]. Appearance-based methods are used for high accuracy of detection. Haar-like features calculate differences in averages of two rectangular regions' intensities and texture can be extracted without banking on absolute intensities. Viola-Jones uses this for face detection technique [106]. Haar-like features contain rectangular windows at certain locations shown in **Error! Reference source not found..**

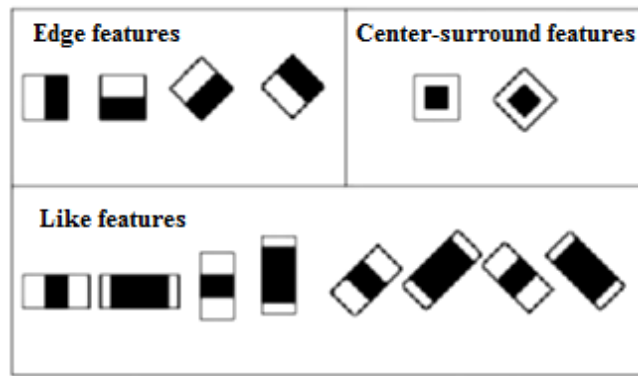


Figure 14 Haar-Like Features [Source: Qing Chen et. al., 2007] [107]

It sums the intensities of the pixels in every region and their difference is calculated. Scalar values represent average intensity differences between two rectangular regions. Intensity gradients are stored at various points, spatial frequencies, and directions by varying shapes, location boxes concerning base boxes, and sizes of the boxes [108]. The integral image at location x, y has a sum of pixels left and above of x, y , inclusive [109].

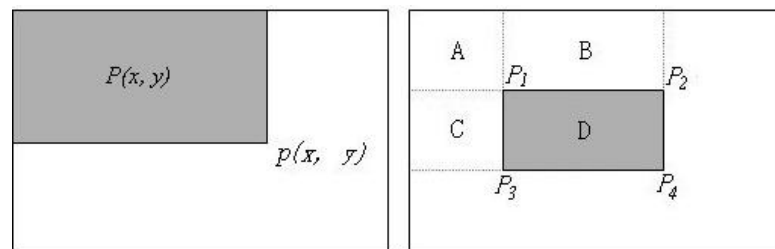


Figure 15 Integral Image Concept [Source: Qing Chen et. al., 2007] [107]

Cheeks are lighter than eyes so Haar features of two adjacent rectangles will be used for detections. One will cover eyes and others will cover cheeks. The intensities of each rectangle are summated and the sum of the values of the cheeks is subtracted from the values of the eyes. Nose, mouth, etc. are also detected by this method [102]. As in Figure 16 left rectangles compare the intensities of the eye and upper cheeks, whereas the other one compares the nose bridge and eye intensities.

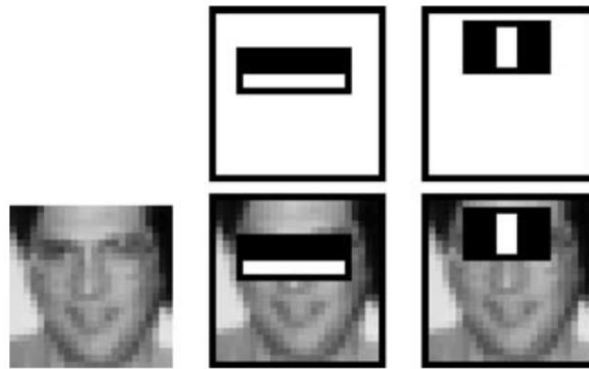


Figure 16 Viola-Jones Face Detection [Source: Paul Viola et. al, 2004] [109]

Haar-like feature identifies the region but there are a lot of features, for example 24x24 pixel window has 1,80,000 features [110]. It will require the inflated cost of time and space. In that many features only, few are strong classifiers for detecting faces. Only these features are used to develop algorithms for detection with similar accuracy. Therefore, for this purpose, AdaBoost is used. It picks only essential features when used together and gives a classifier that is operative for the detection of faces from the face. AdaBoost is adaptive and can be used in several types of situations. These classifiers are modified as per the new challenges.

Where Viola-Jones in [109] discusses boosting algorithms for learning a query online. It describes a weak classifier of a sub-window having threshold, feature, and polarity indicating the direction of the inequality [110]. Another way of face detection is by representing faces as ellipses. Only three features are required to detect a face. It includes the top of the head, chin, and eyes. The major axis is calculated by measuring the distance between the chin and the top of the head. Whereas the minor axis is the length of both eyes from one endpoint to another. Using lengths of major and minor axes ellipse is created covering the human face without ears. This algorithm is not valid for tilted faces, faces covered with hairs, or some other thing, and double chin. However, variation in the said scheme can overcome the issues [111].

Viola-Jones algorithm is the most popular face detection method. As discussed above, it uses Haar-like features. In addition to this, it uses the concept of Cascade filtering and AdaBoost for face detection. This algorithm has three stages. Integral image is calculated by *Summed Area Table* i.e., an algorithm using a data structure. It does a similar operation of computing the sum of values in a rectangular sub-window

of a matrix [112]. Mathematically the sum of pixels left, and above point (x, y) will be the integral image at location (x, y) [113]. As previously discussed, there are thousands of features for the Haar-like features algorithm. Viola-Jones uses the AdaBoost Algorithm for the selection of basic features for classifier training. It helps to create strong classifiers using weak classifiers in linear combinations [107]. A cascaded classifier structure is used to accept/reject faces at different levels. Initial levels are simpler and are used to discard the majority of non-face windows. Complex classifiers are used at the end where the false positive rate is low [114]. The higher the level of the cascade, the more complex the classifier design will be [110].

Every algorithm has time and space complexity. Developers are busy making new algorithms or making earlier versions more efficient. After the face detection process, the problem arises is that whose faces this. This process is called face recognition. There are many algorithms developed to date.

Scale Invariant Feature Transform (SIFT) is one of the techniques used for face recognition. The first step is to detect blob-type objects in an image by LoG (Laplacian of Gaussian), the space scale is made around the points of interest. These points are called key points. The input image is convolved with variable-scale Gaussian to give scale space function.

For actual scale invariance, normalization of the Laplacian, with the factor is done. Scale is automatically selected by convolving the image with the Laplacian function [115].

An efficient function for computing scale space is the Laplacian Pyramid by Burt and Adelson, 1983. It is done by applying Gaussian filters to the image with different values of σ . Subtract one from the other to get LoG. This operation is repeated by scaling the image. The image below shows a pictorial view. The typical value of σ is 1.6 and for k is $\sqrt{2}$. In the scale values of k in the first octave are $k, k^2, k^3, k^4 \dots$. In the next scale image resolution is decreased to half and then the same process is repeated but here the value of k is $k^2, k^3, k^4 \dots$ to find the Difference of Gaussian (DOG).

The value of each key point is assessed. If there is a low contrast structure, then it is removed as it would be sensitive to noise. For hidden peaks in scale-normalized LoG

operators, the ratio of principal curvature of every key point is assessed. The key point with a ratio lower than the threshold is stored [115].

Another algorithm SURF (Speeded Up Robust Features) uses similar rules and steps of SIFT but with a different scheme. It is faster than SIFT. Both use cascading filtering for scale invariant feature detection and DOG is calculated by downscaling the image. Scale space is divided into levels and octaves in both cases. Both find the extrema among 8 neighbouring points at the same level 9 points at a level below and 9 points at a level above as shown in Figure 17[116].

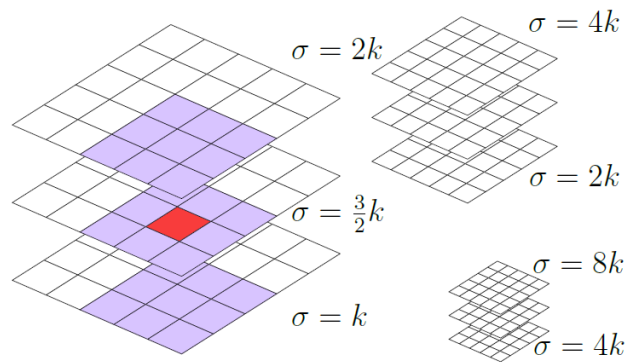


Figure 17: Three Level Octaves [Source: Pederson et. al. 2011] [116]

Hessian-based blob detector is used to locate interest points in the SURF algorithm. Hessian matrix determinant displays the magnitude of response and intensity of change around a point. This computation is overly complex. Integral images and approximated kernels speed up the process [116]. Hessian matrix is converted to discrete form and then cropped. Three colours are used black, white, and grey in the rectangles of the approximated kernels. Grey is assigned the value of '0' where white has positive and black has negative [117]. It is evident from Figure 18 that the filter size has been increased by decreasing the size of the image [113].

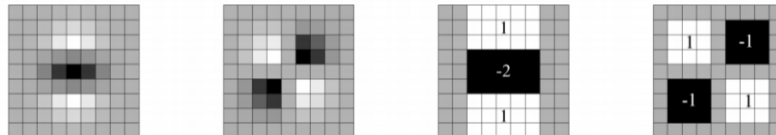


Figure 18 Gaussian Second Order Partial Derivative [Source: Binesh T.R et. al., 2012] [113]

Another commonly used algorithm for feature tracking includes KLT (Kanade-Lucas-Tomasi) which can track points in a video sequence. It can be used for camera motion estimation, video stabilization, and object tracking. It is useful for non-shape-changing objects or remains visually similar. It can be useful for shorter-time tracking [118].

2.07 Outline of Machine Learning

Machine learning is generally defined as computational methods using past information that is already available for accurate predictions. Past information refers to the data already been collected for analysis. This digital data could be in the form of training sets labelled by humans [119]. Machine learning is classified into two types, Supervised Machine Learning and Unsupervised Machine Learning.

Supervised Machine Learning is commonly used. Labelled training set of inputs and output pairs are given to training system,

$$D = (x_i, y_i)_{i=1}^n$$

Eqn. 2-1

System learns to predict output $\tilde{y}$ for given input $\tilde{x}$. Where $\tilde{x} \notin D$. Classification or Pattern Recognition has discrete values of y like $y = \{1, 2, 3, \dots, C\}$, if $C = 2$ labels or classes, it is called binary classification. Multi-class classification has more than two labels [120]. When the instances are not labelled, this type of learning is called Unsupervised Learning [121]. Set of rules are learnt from the training sets. These set of rules are classifiers. Dataset is collected by the expert and fields may be suggested. If no field is suggested, it is assumed that everything presented is perfect. This method is known as “Brute-Force”. Usually, this type requires pre-processing of the data [122]. Redundant and irrelevant features are identified and removed in the process of Feature Subset Selection [123]. Supervised Machine Learning is performed by various techniques. It includes Logic Based Algorithms, Perception Based Algorithms, Statistical based techniques and Support Vector Machines [124].

Decision trees is example of Logic Based Machine Learning. Instances are classified by sorting based on feature values classified by decision trees. Decision trees

are divided into nodes. Every Node shows a feature of an instance and displayed in Figure 19 [125].

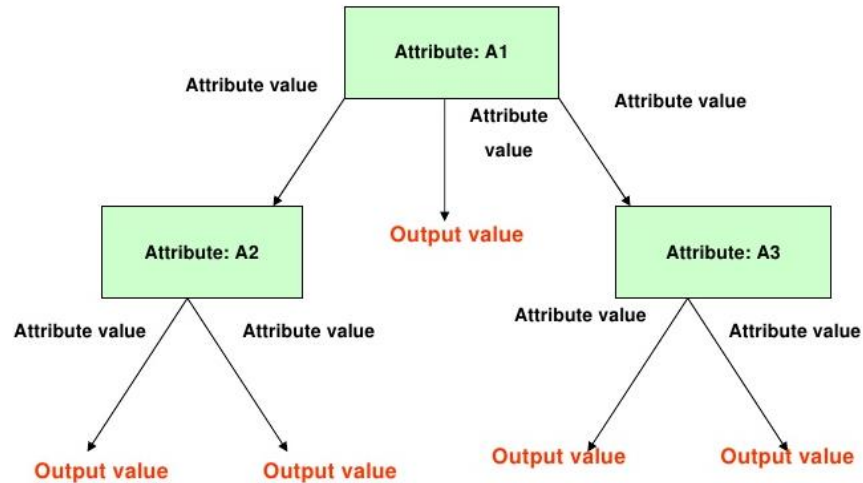


Figure 19 Decision Tree Example [Source: Kevin P. Murphy. 2007] [125]

Set of rules can be created from the decision trees. Every path has a separate rule beginning from the root and moving down to a leaf of the decision tree. These rules can be inducted from the training data based upon rule-based-algorithms [126]. Training data with small but efficient set of rules depicting all conditions is required having high predictability and reliability.

Perception based techniques are very well known. Neural Networks are perception-based algorithms. Perception can be given as: If input features values $(x_1 \dots x_n)$ are assigned weights $(w_1 \dots w_n)$. Then the perception will be the weighted sum of inputs $\sum_i x_i w_i$. Perception output is adjusted against a threshold. If its value is greater than threshold than output will be 1 otherwise 0. Usually, algorithm is run repeatedly with the training instances on the training set to find prediction vector [120]. This type of prediction can deal with linearly separable set of instances. Artificial Neural Networks can deal with linearly non-separable learning [127]. It is done by creating a multi-layer neural network having many neurons joined in pattern of connections. Units are divided into three stages: Input, outputs and hidden units. Based on set of paired data network is trained to develop input-output mapping. Then weights

are assigned to the neuron connections. This network is afterwards used for classifications of new data [124].

Statistical learning approach differs from neural networks by providing a probability of an instance to belong from each class. Bayesian Network is one method of statistical learning that shows graphical model for probability relationships between features [128]. Normal distribution can be used for calculation of probabilities [129]. Instance based learning algorithms are also type of statistical learning approach. They are generally referred to be lazy algorithms as they have lesser training computation time than bayes and neural networks, decision tree etc. but longer time is required for classification [130]. k- Nearest Neighbour (KNN) is based on instance-based algorithms. KNN is more accurate than Bayes, but time complexity is higher [131]. Group of k instances are found in the training set by KNN. These k instances are closest to the test input. Label of the predicted class depends upon the probability of occurrence of a specific class in its neighbourhood [132]. Support Vector Machines (SVM) are new machine learning techniques. Support Vector Machines works on the margin on either side of hyperplane that separates two classes of data [133]. Generalization error is reduced by increasing the margin. It is increased by creating maximum distance between separating hyperplane and instances on both sides. After finding the optimum separating hyperplane for linearly separable data, the data points that lies on its margin are called support vector points. Rest of the data points are ignored. This is called sparseness, and it does not affect the classification function [134].

MATLAB offers classification learner toolbox for learning, regression, and clustering purposes. Different models can be exported for further use in the systems developed. . Common algorithms like SVM, KNN, Naïve Bayes, Bagged decision trees, Neural Networks etc. are available for modelling. Moreover, Conventional Neural Networks can be used for Deep Learning [135]. It will be discussed in following sections.

2.08 Outline of Deep Learning

Deep learning is a sub-branch of machine learning. It is also known as hierarchical or deep structured learning. Artificial Neural Networks methods are used in deep learning [136]. Deep learning can be differentiated by the number of learnable chains; the credit assignment paths and the links between the effects due to the actions [137]. Computational models are made up of numerous layers for processing and data representation is learnt with multi levels of abstraction in deep learning. Various artificial intelligence tasks have been improved in recent years due to the introduction of deep learning. These include image recognition, speech recognition, object detection and in the field of genetics. For a large data set the link structure between the nodes becomes too complex and it becomes complex in all the layers. Deep learning overcomes the problem by generation of backpropagation algorithm to identify that how internal parameters should be changed by the machine so that each layer representation may be computed [19]. Deep learning uses various architectures like convolutional neural networks, deep belief networks, deep recurrent networks, and deep neural networks. These are used in the fields of bioinformatics, machine translation, computer vision, speech recognition, medical imaging, and material analysis. Deep learning gave wonderful results in these fields and surpassed humans in some cases [138].

Machine learning has evolved most popular area of research in recent years. It includes substantial number of applications in every major field of science and technology like social network, computer vision, image recognition, medical, media industry, energy, environmental issues and so forth. Above all machine learning applications, deep learning is the core technique which is widely applied on learning algorithm that has deep roots in convolutional neural network, generative network and recurrent neural network etc. Deep learning uses neurons to ripen deep layered complex learning models to outclass its predecessors' learning techniques. After the advancement in deep learning techniques, promising efficient results to have been demonstrated in various types of applications like data processing, audio and video processing etc. [139].

2.09 Deep Learning Techniques

Building neural networks with numerous layers is a key component of the machine learning branch known as deep learning, which aims to learn data representations. Supervised learning, unsupervised learning, semi-supervised learning, and reinforcement learning are primary categories into which deep learning approaches can be generally divided.

I. Supervised Learning

Supervised learning works on the reason which are applied externally to deduce some hypotheses. The general hypotheses lead instances of inputs towards future predictions to find some valuable solutions. The main theme behind supervised learning is to design a model to work out on feature predictor. Regression and classification are the major applications of supervised learning techniques. Object detection, image recognition, image segmentation, image classification and scene analysis are the major examples of supervised learning models [140][141].

II. Unsupervised Learning

Unsupervised learning technique is still in evolving due to its problem learning ability unless it has been showing very promising results in its various applications. In literature [142] self-learning models have been discussed having valued features. Many models and techniques have been developed using unsupervised learning technique. CNNs are widely used for video and image features analysis having very promising accuracy results [143]. GAN has also become vital technique in unsupervised deep learning model which has discussed in [144]. It has been known as high-level feature and pattern extractor technique in recent studies [145].

III. Reinforcement Learning

This learning technique is used to take decision in sequence of uncertain events in complex environment scenario. Learning ends with a reasonable solution to the problem. It works on the reward policy via agent-based learning to work on solution [146]. The model performs its function to elevate the reward in each tasks having random trials which has showed in literature [147]. Reinforcement learning is most

effective technique in the domain of search and trials. It has capacity to work on algorithm with powerful infrastructure. The computational tasks have been enhanced with potential of wide applications in recent time [148]. Simulation and scaling have been a challenging task for reinforcement learning to communicate the networks of the system since the beginning. This may lead to technique a catastrophic environment having the threat of erasing the previous knowledge.

IV. Semi-supervised Learning

In this learning technique the algorithm performs a task to work on labelled and unlabelled data while data is used after passing through the cluster environment. Data labelling is the typical use of this learning technique for unlabelled data. Because the cost of unlabelled data is extremely low as compared to labelled one [149]. Semi-supervised learning shows its assumption having continuity of same output and discrete cluster point which do have manifold of less dimension from the input label.

2.10 Architectural Models of Deep Learning

Over the past two decades, numerous deep neural network models have been created. In the context of total layers, layer kinds, layer shapes, and inter connections, each model has a unique network architecture [150].

I. Artificial Neural Network (ANN)

Artificial neural network is described as a group of neurons in any layer of neural network which can also called feed-forward neural network. The layer network of forward feed neural network is comprised of three basic layers which input, hidden and output [151]. Therefore, each layer performs specific task which is given to it and to try learning specific weight against each single task. It has an ability to work and solve the issues which are linked to text data, image data and tabular form data. It has capacity to learn any type of function i.e., nonlinear function. It shows results on any kind of learned weights to map due to activation function property of it, because it helps to work on complex working relationships in between output and input [152]. The main power of artificial neural network is its ability of activation function. The major issue is the parameters increase dramatically with the size of an image during training. Therefore, in an image analysis it loses the spatial features during

arrangement of pixels. The other common networks problems can be vanishing and gradient which is directly associated to backpropagation and the backpropagation is used to update the issue of vanishing and gradient. On the other hand, for very deep neural networks ANN is extremely limited for dealing of sequential information [153].

II. Recurrent Neural Network (RNN)

Recurrent neural network has looping connection in the networks of hidden layers. The recurrent constriction makes sure that the information is apprehended with in the input. This type of network for data is used for video analysis, time series applications and text data as well. Therefore, RNN has ability to work with parameters and share their information across each step-in parameter sharing process. These sharing parameters play vital role in training and to reduce the cost of computational tasks [154]. RNNs are also in common network problems like ANN, gradient and vanishing which may have in every type of neural network. It is an appropriate for the use of temporal data. The compatibility level is low as compared to CNN in feature mapping and extraction. It has an ability to work and process random sequential inputs by using internal memory. It is most significant for and uses the information of time series for speech and text analysis. In case of prediction, the network learns and work by itself to achieve required prediction by back propagation.

III. Convolutional Neural Network (CNN)

Convolutional neural network (CNN) is being used and applied to extract the most relevant information of given data using simple to multi-level hidden layers networks. The basic feature of CNN is to convolve an image input with filters to extract features from it having a bias as input that needs weight [155]. A dot product is applied in between input and filter to receive a single value as a result. It has an ability to learn automatically on filters in a specific training dataset for prediction of feature mapping related to problem. The output should be highly impressive against the input for recognition of features [156]. The major application of CNN is image and video analysis ideas for classification and recognition. CNN works on various kinds of features from which spatial feature of an image using filters which can be worked as detectors. It refers to the working arrangement and sequences of pixels values from one dimensional layer to two dimensional layers. It helps to identify and recognize the

location and shape of any object with its domain of analysis [157]. Research has shown that convolutional layers can be applied to output layer instead input. It shows hierarchical decomposition by applying stacking of layers in convolutional networks. Therefore, CNNs are required to work and save spatial image structure in given input scenario [158].

IV. Recursive Neural Network

Recursive neural networks models are non-linear in behaviour. Models can learn the pattern of deep structured information. Learning rate is robust for being appropriate processing and complex computational cost [159]. Contextual information is used for the recognition of complex face using video sequencing approach and recursive neural network can be applied for the localization of face in complex mode [160]. Graph based approach can be used for the localization of face using recursive neural network technique [161]. Symbolic and structural features of face can be assumed using recursive neural network approach and technique. The establishment of face position inside the image of any video sequences can be established by recursive neural network [162].

V. Generative Adversarial Networks (GANs)

Generative adversarial networks (GANs) work on supervised learning problems in a very clever way for the training of generative model on two neural networks models for the generation of new accurate data. They are unsupervised learning models to work and learn on features automatically [159]. They have been widely applied in the generation of voice, image and video. GANs have great capacity to work for both good and bad. They can work on the learning of mimic for the distribution of any type of data. They can manipulate the information to fake content using deepfakes techniques. The basic working of generative algorithms is informative with discriminative algorithms to predict the data for its classification and mapping. The working of GANs can be discriminated with the generation of new data having discriminator to evaluate the information for its genuineness. Generator works on the generation of new, authentic images on the prediction that they would be genuine source of information [163]. Therefore, the aim of generator is to work and identify a real data from the fake information. They work on random images and to send random data to the

discriminator on parallel ground to check the authenticity of images. They can have double checker feedback design in the form loop within the domain of images. Both GANs and discriminator are applied for the optimization of various functions for any given tasks. It is known as actor-critic model for generator and discriminator behaviour. GANs are being advanced by putting promising results across key issues by translating the real-world tasks. GANs are executing to work in the solution-based models which have been domain specific i.e., image to image pattern and feature recognition and translation. The most recent advancement can be seen as data augmentation having remarkable results and less probability of generalization errors. [159].

VI. Auto-encoders

Auto-encoder works on the principle of unsupervised learning model. It works on learning algorithms for compression of information and the methods of data reconstruction to receive back from encoding and compression. It reduces the noise and dimensionalities of data during learning. The main component of encoder which is applied for the compression of data [196]. Then it is converted to encoded representation of input information having low level of dimensions. The encoded information is passed by decoder for reconstruction of data into its unique form. It can be observed the reconstruction loss by the efficiency of output. Auto-encoder has two parts, encoder, and decoder. The main purpose of decoder is to order the auto-encoder to select the useful feature of encoded data. Therefore, it can be seen in auto-encoder, reconstruction has healthier and promising results. Auto-encoder has ability to improve the characteristics features of datasets [163].

2.11 Applications of Deep Learning in Image Processing

Classification rearrangement, identification, language processing, video processing, gesture detection, robotics, etc. are just a few of the numerous computer vision difficulties and issues that need to be resolved. The world of deep learning is currently advancing, and all computer vision issues are deemed to be solved. CNN-based models, or Convolutional Neural Networks, have completely changed the field

of computer vision in recent model-based progress. Now, creating a variety of Deep Learning setups utilising modifying with pre-trained weights is really simple. Using ImageNet [20] for categorization, as an example. However, the more complex problems of object detection and segmentation require more sophisticated approaches to the solution. While segmentation looks for specific pixels that fit each object, object detection involves learning the objects and constructing a rectangle-shaped bounding box. One of the most common variances in picture categorization is that a single image may contain numerous items and people that are all in completely different sizes, lighting conditions, and spatial relationships, as well as being partially obscured [164]. Image recognition process is being used in deep learning techniques for the detection of objects, colours, texts, writing style, place, people, gender etc. Cameras are the main source to capture scenes and store its internal memory to make them in physical form to view and analyse. Image recognition uses multiple algorithms techniques to work on labelling, features, and patterns of images with various kinds of tags implementations. In an analogous way, human recognizes the multiple objects in an image with their computational capacities of brain inside skull. Recent advancements in deep learning techniques which uses software tools to assess the quality and content of images. Successful results have been achieved using convolutional neural networks (CNN) technique which is deep learning technique [165]. Algorithms have been trained on the various aspects of image analysis from scene detection to edge detection inside the image components. Millions of pre-labelled images data have been trained to make image recognition analysis more advanced. Therefore, targeted trained datasets are the main sources to work on social media advertising and images galleries-based activities [166]. Visually impaired analysis research work has been working on image components analysis, colours schemes, object detection and recognition to provide a practical solution for visual impaired people and make their life easy to see the world beauty. Image classification has been effective using convolutional neural networks (CNNs) which is most significant deep learning technique for visual processing. It works on image dimensions, image classes, image parameters, image channels etc. CNN provides pixel level results with high accuracy [167]. It uses labelling on input data to work on prediction accordingly for annotation and tagging of images automatically. In image analysis it is required to focus on targeted region.

Therefore, region-based convolution neural network technique can be used for the object detection and analysis. This learning technique is known as RCNN. Mask RCNN which is also called faster RCNN is being used to on image processing at hum level. The output results of Fast RCNN are tremendous even to recognize the category of products [168].

Object detection seeks to locate a group of target items, picture classification attempts to describe the image. The detection task is divided into two parts, the first of which is the classification task and the second of which is the category information and likelihood of the target. The other positioning duty is to determine the target's precise location by using bounding boxes with labels. The two primary categories of current mainstream strategies are the one-stage strategy like SSD / YOLO and the two-stage approach like R-CNN. The two-step method produces a sparse set of the bounding box from the image in the first stage. The final detection results are then improved by corrections depending on the bounding box region. Single-stage method, in contrast, directly calculates the image and produces detection results. Although the single-stage detection speed is quicker, it has a lesser level of detection precision. The two-stage strategy, however, does the exact opposite [169]. A lot of applications use deep learning because of its potent capacity to extract picture features, precise recognition, and quick real-time. It has particularly accomplished remarkable accomplishments in image identification for remote sensing, facial recognition, and medical imaging. Prior to the development of deep learning technology, image recognition was primarily accomplished using conventional technical techniques, such as principal component analysis, the Laplace feature map method, local value-preserving mapping, sparse representation, neural network dimensionality reduction, etc. However, because of the shortcomings of conventional technical methods, the performance of image recognition is not optimal. Deep learning technology advancements have made it possible to gradually use deep learning to image recognition. Subsequent scientific study has produced promising findings in terms of recognition efficiency and recognition effect. Deep learning technology advancements have made it possible for researchers to gradually apply deep learning to image recognition. Positive outcomes have been obtained in subsequent scientific work, and

image recognition effectiveness and efficiency have significantly increased compared to traditional recognition methods [170].

2.12 Face Detection using CNN

While selecting the algorithm to detect faces, the main thing to be taken care of is the type of images we are dealing in. For face detection, the first choice used to be Viola Jones but while using the viola jones for face detection it is fast but not effectively translated dealing with the small images that a CCTV image contains. As discussed in the literature *Deep Learning*; CNN based detectors. Since 2013 till date, we can see advancements in CNN based systems. Two major types of detectors are one stage and two stage. In two stage detectors one stage is for region of interest and the other stage is for classification. Whereas in the one stage detector, localization and classification is done at the same time. Problem is solved by performing the classification and bounding box regression directly on feature maps making it regression problem instead of detection problem. This solution reduces detectors inference time.

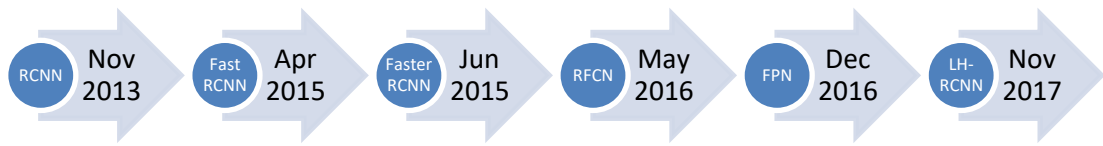


Figure 20 Two Stage CNN Detectors Evolution



Figure 21 One Stage CNN Detectors Evolution

Belhassen et al. (2019) [171] conducted a study comparing Face and Person Detection algorithms, presenting an extensive overview encompassing frame rates and accuracies of various algorithms. Authors evaluated both speed and accuracies on the dataset while selecting a technique. In comparison viola jones has a limitation of detection of side poses and small faces, whereas Faster RCNN can resolve the issue by detecting small faces. But the Faster RCNN is slower as compared to other methods. Whereas YOLO shows better results when compared to Faster RCNN. While comparing different variants of YOLO, YOLO v2 is faster than YOLO v3. But in terms

of accuracy, YOLO v3 outperforms YOLO v2 [171]. For best results RCNN, Faster-RCNN and YOLO are the best choices. YOLO v2 is faster as compared to YOLO v3 but it is less capable of detecting small objects. More layers are added in the YOLO v3 that makes it slower but efficient in detecting small objects [172].

YOLO v3 which has a clear advantage of small object detection on YOLO but requires anchor box estimation. The image must be rescaled to the network layer for the YOLO v3 parameters. It was sorted out by keeping the input layer size smaller than that of the stored image to keep training and execution faster. The architecture of YOLOv3 is a convolutional neural network (CNN) based network which consists of 75 convolutional layers. The network is trained to predict bounding boxes and class possibilities for each object in the input image. These bounding boxes are defined by the arrangement of the upper left corner and the lower right corner, and the probability of a class, which indicates that the object may belong to the same class. YOLOv3 uses a multiscale method of the network to perform image processing at multiple scales and make predictions at each level. The network uses anchor boxes to improve the accuracy of predictions. An anchor box is a combination box of different shapes and sizes used to locate different objects. The network is divided into three parts: feature extraction subnet, convolutional subnet, and prediction subnet.

The architecture of YOLOv3 is a convolutional neural network (CNN) based network which consists of 75 convolutional layers. The network is trained to predict bounding boxes and class possibilities for each object in the input image. These bounding boxes are defined by the arrangement of the upper left corner and the lower right corner, and the probability of a class, which indicates that the object may belong to the same class. YOLOv3 uses a multiscale method of the network to perform image processing at multiple scales and make predictions at each level. The network uses anchor boxes to improve the accuracy of predictions. An anchor box is a combination box of different shapes and sizes used to locate different objects. The network is divided into three parts: feature extraction subnet, convolutional subnet, and prediction subnet [173].

This subnet starts with a multiple of convolutional layers stacked in series with a kernel size of 3x3 and a step of 1, followed by batch processing and Leaky ReLU

activation function. The number of filters at these levels starts at 32 that increases with the depth of the CNN network. This subnet includes a maximum number of layers with a kernel size of 2x2 in step 2. Shortcut links are used to connect the output of some layers, which helps improve accuracy by reducing the trailing gradient problem.

This subnet consists of several convolutional layers with a kernel size of 3x3 and a step of 1, followed by batch processing and Leaky ReLU activation. The number of filters at these levels starts at 64 and increases with CNN network depth. Some layers also use a 1x1 convolutional layer to reduce the number of channels, which is useful for reducing computational complexity [174].

This subnet is composed of a series of convolutional layers with a kernel size of 1x1 and a step size of 1, followed by batch processing and Leaky ReLU activation. The number of filters at these levels starts at 128 and increases with network depth. Anchor boxes are used to improve accuracy. The last layer predicts the bounding box and class space for each object in the image.

In addition to the three underlying networks, YOLOv3 also includes a global and fully connected (Dense) layer used for classification. The network also uses skipping connection to make the output of different levels, which helps to improve the accuracy by reducing the gradient decent challenge. YOLOv3 also utilizes Non-Max Suppression (NMS) to eliminate duplicate associated boundary boxes and improve the final prediction. NMS evaluates the class probability and IoU (Intersection over Union) between the closed boxes and keeps only the boxes with the highest-class probability and the highest IoU. YOLOv3 is capable of real-time object detection with high accuracy, which makes it suitable for fast detection [173].

Convolutional layers are a sort of layer in the neural network, and they are referred to as "conv" in YOLOv3. Since convolutional layers can efficiently capture the spatial information and relationships between pixels in an image, they are frequently utilized in computer vision applications like picture classification and object detection. In a convolutional layer, the input is processed via several learnable filters to create a set of activations, which are then forwarded to the network's subsequent layer. The filters can learn to recognize different features in the input data, such as edges or textures.

A residual layer in the YOLOv3 neural network is a sort of layer that makes it simpler to backpropagate gradients, which enhances training. The residual layer enables the network to more efficiently learn the desired input to output mapping by allowing the "residual" of the activations from one layer to be added to the activations of the following layer.

The "route" layer in YOLOv3 is a particular kind of concatenation layer. It merges several feature maps from preceding network layers into a single feature map. The layer configuration specifies the feature maps to be combined, and the combined feature map has the same spatial resolution as the original feature maps. In YOLOv3, multi-scale predictions are implemented using the "route" layer, which enhances the reliability and precision of object recognition.

A method of upscaling an input tensor by boosting its spatial resolution is referred to in YOLOv3 as "up sampling." The size of the feature maps from a lower layer can be increased using this technique to match the spatial resolution of the feature maps from a higher layer. The decoder portion of the network in YOLOv3 employs up sampling to enlarge the feature maps so that they may be utilized to produce the final bounding box predictions for object detection. A transposed convolution is commonly used to accomplish the up-sampling operation in YOLOv3 and increases the size of the feature maps by a predetermined scale factor.

Layer #	Layer	Filter Size	Repeat	Input	Output
0	Conv	32 3 x 3 / 1	1	608 x 608 x 3	608 x 608 x 32
1	Conv	64 3 x 3 / 2	1	608 x 608 x 32	304 x 304 x 64
2	Conv	32 1 x 1 / 1	1	304 x 304 x 64	304 x 304 x 32
3	Conv	64 3 x 3 / 1		304 x 304 x 32	304 x 304 x 64
4	Residual	1		304 x 304 x 64	304 x 304 x 64
5	Conv	128 3 x 3 / 2	1	304 x 304 x 64	152 x 152 x 128
6	Conv	64 1 x 1 / 1	2	152 x 152 x 128	152 x 152 x 64
7	Conv	128 3 x 3 / 1		152 x 152 x 64	152 x 152 x 128
8	Residual	5		152 x 152 x 128	152 x 152 x 128
12	Conv	256 3 x 3 / 2	1	152 x 152 x 128	76 x 76 x 256
13	Conv	128 1 x 1 / 1	8	76 x 76 x 256	76 x 76 x 128
14	Conv	256 3 x 3 / 1		76 x 76 x 128	76 x 76 x 256
15	Residual	12		76 x 76 x 256	76 x 76 x 256
37	Conv	512 3 x 3 / 2	1	76 x 76 x 256	38 x 38 x 512
38	Conv	256 1 x 1 / 1	8	38 x 38 x 512	38 x 38 x 256
39	Conv	512 3 x 3 / 1		38 x 38 x 256	38 x 38 x 512
40	Residual	37		38 x 38 x 512	38 x 38 x 512
62	Conv	1024 3 x 3 / 2	1	38 x 38 x 512	19 x 19 x 1024
63	Conv	512 1 x 1 / 1	4	19 x 19 x 1024	19 x 19 x 512
64	Conv	1024 3 x 3 / 1		19 x 19 x 512	19 x 19 x 1024
65	Residual	62		19 x 19 x 1024	19 x 19 x 1024

Figure 22 YOLO v3 Feature Extraction Layers

As shown in Figure 22, 53-layer variant of Darknet is used by YOLO v3. This network is trained on ImageNet. Another 53 detection layers are piled to it, making a

total of 106 fully connected layers 1x1 kernel is applied on the feature map to generate output. The network is mostly composed of convolutional kernels that are either 1 x 1 or 3 x 3. There are three different size kernels that are applied on feature map at different layers of the network to get the output. For this research CCTV image is initially divided into sections of “608 x 608” sized images. The final detection kernel can be estimated by the relationship “ $1 \times 1 \times (B \times (5 + C))$ ”. Where B is the total boundary boxes a cell can predict. The number “5” represents the four parameters of the boundary box (x coordinate, y coordinate, width, and height) and the confidence level of the prediction for that class. For the dataset used, the value of B is three and C is one as there is only one class that is face. Calculating the values, the size of the kernel was found to be 1x1x255. YOLO V3 uses DarkNet-53 as a feature extraction tool. Pooling layers information loss is prevented by the use of fully connected layers. There are “five” residual modules that help to extract features in an efficient manner. Prediction is done at three different scales of the down sampled input image. The input image is scaled down to 32, 16 and 8 times for prediction [174].

75	Conv	512 1 x 1 / 1	1	19 x 19 x1024	19 x 19 x 512
76	Conv	1024 3 x 3 / 1		19 x 19 x 512	19 x 19 x1024
77	Conv	512 1 x 1 / 1		19 x 19 x1024	19 x 19 x 512
78	Conv	1024 3 x 3 / 1		19 x 19 x 512	19 x 19 x1024
79	Conv	512 1 x 1 / 1		19 x 19 x1024	19 x 19 x 512
80	Conv	1024 3 x 3 / 1		19 x 19 x 512	19 x 19 x1024
81	Conv	255 1 x 1 / 1		19 x 19 x1024	19 x 19 x 255
82	YOLO Detection Large Size Face				
83	Route 79				
84	Conv	256 1 x 1 / 1	1	19 x 19 x 512	19 x 19 x 256
85	Upsample	2x	1	19 x 19 x 256	38 x 38 x 256
86	Route 85, 61				
87	Conv	256 1 x 1 / 1	1	38 x 38 x 768	38 x 38 x 256
88	Conv	512 3 x 3 / 1		38 x 38 x 256	38 x 38 x 512
89	Conv	256 1 x 1 / 1		38 x 38 x 512	38 x 38 x 256
90	Conv	512 3 x 3 / 1		38 x 38 x 256	38 x 38 x 512
91	Conv	256 1 x 1 / 1		38 x 38 x 512	38 x 38 x 256
92	Conv	512 3 x 3 / 1		38 x 38 x 256	38 x 38 x 512
93	Conv	255 1 x 1 / 1		38 x 38 x 512	38 x 38 x 255
94	YOLO Detection Medium Size Face				
95	Route 91				
96	Conv	128 1 x 1 / 1	1	38 x 38 x 256	38 x 38 x 128
97	Upsample	2x	1	38 x 38 x 128	76 x 76 x 128
98	Route 97, 36				
99	Conv	128 1 x 1 / 1	1	76 x 76 x 384	76 x 76 x 128
100	Conv	256 3 x 3 / 1		76 x 76 x 128	76 x 76 x 256
101	Conv	128 1 x 1 / 1		76 x 76 x 256	76 x 76 x 128
102	Conv	256 3 x 3 / 1		76 x 76 x 128	76 x 76 x 256
103	Conv	128 1 x 1 / 1		76 x 76 x 256	76 x 76 x 128
104	Conv	256 3 x 3 / 1		76 x 76 x 128	76 x 76 x 256
105	Conv	255 1 x 1 / 1		76 x 76 x 256	76 x 76 x 255
106	YOLO Detection Small Size Face				

Figure 23 YOLO v3 Detection Layers

As shown in Figure 23, the first detection is made by the 82nd layer. A down sampling is performed in the first eighty-one layers. At the eighty second layer first detection is performed. For the input image having the dimension of “608 x 608” the feature map has now the size of “19 x 19”. For this detection, “1 x 1” kernel is used. The feature map now has the dimensions of “19 x 19 x 255”. In the network, different sized objects are to be detected. For this reason, after layer 79, the feature maps are up sampled by two times and now have the size of “38 x 38”. After that this new feature map is concatenated with the “61st” layer feature map. Now the concatenated feature maps are merged with the help of “1 x 1” convolutional layers. Now the “94th” layer is responsible for the second detection that results in “38 x 38 x 255” feature map. The same steps are performed again in which feature maps from “91st” layer is concatenated with the feature maps of “36th” layer. Resulting in the last detection at the “106th” layer. The final feature map has a size of “76 x 76 x 255”. Large faces are detected by “19 x 19” layer, “38 x 38” detects medium faces and the “76 x 76” layer detects smaller faces [174]. Due to space constraints only YOLO v3 is discussed in the review.

2.13 Transfer Learning for Classification Problems

System was to be designed in such a way that it must be accurate in terms of classification of the detected faces in true categories from which they belong and tag them accordingly. In addition to the accuracy, the other parameter that was to be taken care of was, the time required for training as well as for recognition. In the literature section detailed discussion was made on various types of artificial intelligence techniques for recognition.

Advantage of Deep Learning algorithms over Machine Learning algorithms were established above. Therefore, deep learning algorithms were decided to be implemented for designing the models. In the detection stage three algorithms were evaluated after the pre-processing of each image of the dataset including RCNN, Faster RCNN and YOLO v3. One method was to train the YOLO v3 network to classify the faces into genders and face types. Similar approach is followed by Srithar, S et. al. [175]. Convolution base of these networks plays a role of feature extraction for input image. On the top of these convolution base feedforward network is attached which act as classification on category. The swapping process is shown in Figure 24.

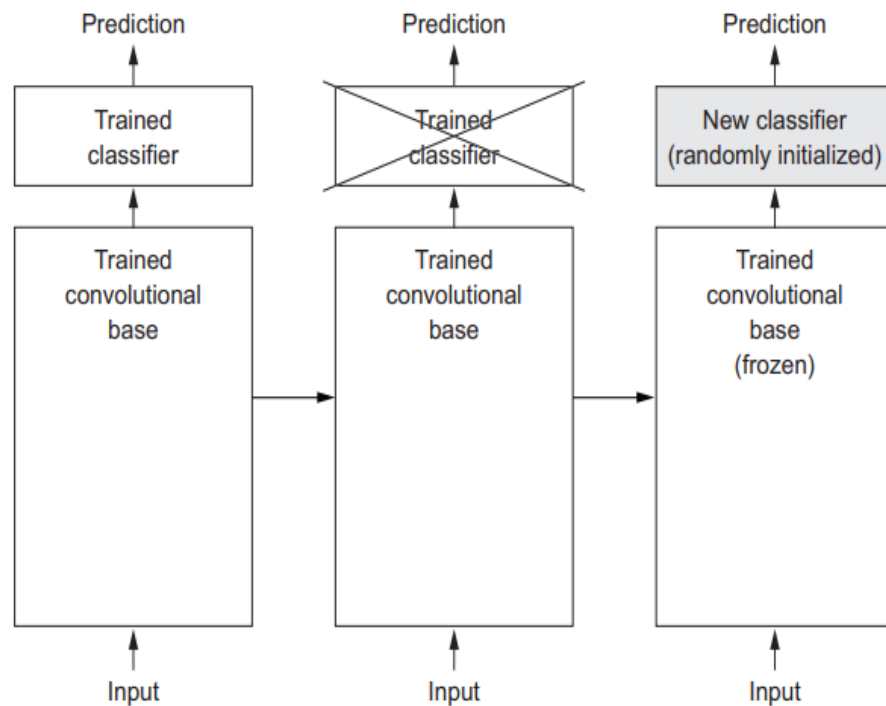


Figure 24 Transfer Learning Process

To perform transfer learning, it begins with the selection of a pre-trained model that is already trained on a large and diverse dataset. GoogLeNet, AlexNet, ResNet-50, VGG and Inception are commonly used pre-trained models. It must be noted that the pre-trained model must include the similar data. Then top layers are often removed those are responsible for specific predictions. Lower layers are not changed in the step, so the general feature prediction remain intact. New layers on top are added as per the training data. The number of new layers and their complexity can vary depending on the complexity of the new task. It is possible for the pre-trained layers' parameters to get frozen, which means they won't be modified while the layers are being trained on the new assignment. Next, the dataset is used to train the model. Only the weights of the recently inserted layers are changed since the pre-trained layers are frozen. This aids the model in adjusting to the particulars of the updated data. After this fine tuning is done and this fine tuning involves the unfreezing of the layers and then training on the new dataset. It helps to adjust models with new features. Before training hyper parameters like batch size, epochs, regularization methods and learning rate are tuned for optimized training [176].

I. AlexNet

AlexNet is one of the states of the art pretrained model used for transfer learning for many computers vision tasks. AlexNet was presented by Alex Krizhevsky in 2012. This architecture has 650,000 neurons with 62,3 million parameters. To get best optimize result from any network, tuning of hyperparameter plays a key role. Best settings of network can deliver best results. AlexNet have sequential model architecture and quite simpler than GoogLeNet and VGG-Face structure.

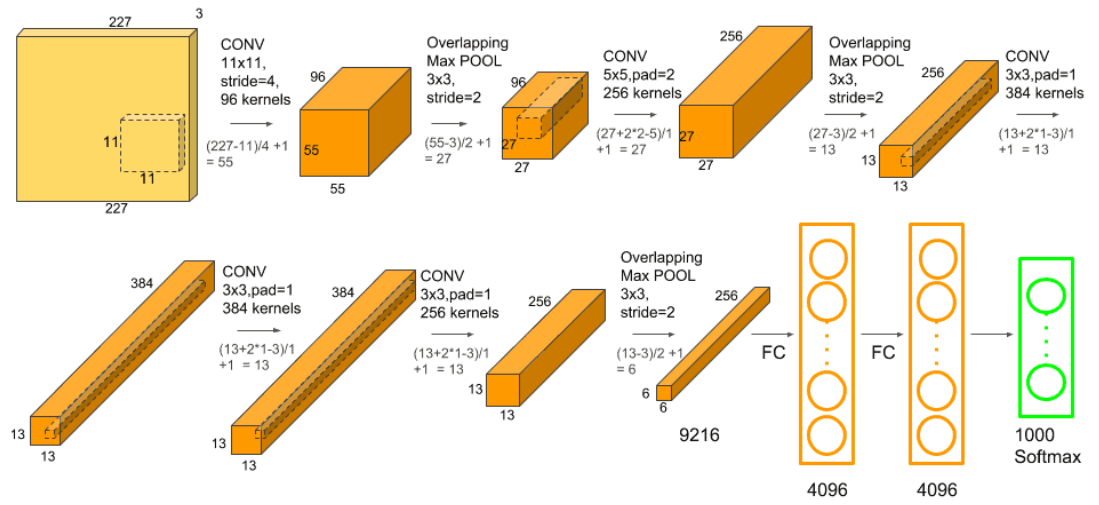


Figure 25 AlexNet Architecture [Source: Abhijeet Pujara, 2020] [177]

As shown in Figure 25, five convolutional layers and three fully connected layers make up AlexNet. The important features in an image are extracted using multiple Convolutional Kernels (filters) [177] [178]. Images with a size of “227 x 227 x 3” are used as the model's input. Padding was used by the authors to limit the dimension of the feature maps from significantly decreasing because AlexNet has a deep architecture.

II. GoogLeNet

This deep convolutional neural network architecture was proposed by Christian Szegedy et. al., in “2014 ImageNet Large-Scale Visual Recognition Challenge”, it set a new benchmark for classification and detection (ILSVRC14). This architecture's offered a better utilization of the network's computational capabilities. This was made

possible by a properly conceived design that permits expanding the network's depth and width while maintaining a fixed computing budget. The Hebbian principle and the multi-scale processing intuition served as the foundation for the architectural choices that were made to maximise quality. GoogLeNet, a 22 layers deep network, is one specific implementation that was used in our submission for ILSVRC14. Its quality is evaluated in the context of classification and detection.

III. VGG and VGG-Face

Convolutional Neural Networks (CNNs) are widely used in image classification and have become a prominent tool in computer vision. The development of CNNs dates to the late 1980s and early 1990s, with a significant boost in performance after the advent of deep learning techniques in recent years. VGG is a deep learning model that was introduced in 2014 by researchers at the University of Oxford. The model was designed to achieve high accuracy in image classification tasks, and it became widely adopted as a benchmark for other models to compare their performance against.

The architecture of VGG is characterized by its small, deep convolutional filters, which enable the model to learn hierarchical representations of images. This design enables VGG to capture both low-level and high-level features of images, making it well-suited for image classification tasks. VGG was trained on a large dataset of over 1 million images and was evaluated on benchmark datasets, such as ImageNet [20]. The results showed that VGG achieved state-of-the-art performance on these datasets, outperforming many other models at the time of its release. Despite its impressive performance, VGG has some limitations. For example, it may not perform well on images with complex structures, such as those with multiple objects or fine details. Additionally, the model may be vulnerable to adversarial examples, which are images specifically designed to fool the model. VGG has been applied to various image classification tasks, including object recognition, scene classification, and medical image analysis. The model's high accuracy and robustness make it suitable for use in real-world scenarios, such as autonomous driving and medical diagnosis. VGG is a powerful model for image classification and has been widely adopted in various applications. Despite its limitations, it continues to be a critical reference for researchers in the field of computer vision and deep learning. Further research is

needed to address the limitations of VGG and to improve its performance in challenging scenarios [179].

Face recognition has been a critical area of research in computer vision and pattern recognition for several decades. The increasing demand for biometric identification, surveillance, and human-computer interaction has led to the development of various face recognition algorithms and models. VGG Face is a variant of the VGG network that is specifically trained for face recognition. It was developed by researchers at the University of Oxford and introduced in 2014. VGG Face is designed to capture the hierarchical representations of faces, which are crucial for face recognition.

The architecture of VGG Face is based on the standard VGG network, but it has been modified to better handle the challenges of face recognition, such as variations in pose, illumination, and facial expressions. VGG Face was trained on a large dataset of over 2.6 million images of 2,622 individuals. The model's performance was evaluated on benchmark datasets, such as the Labelled Faces in the Wild and the YouTube Face datasets. The results showed that VGG Face outperformed many of the state-of-the-art face recognition models at the time of its release.

Despite its impressive performance, VGG Face has some limitations. For example, it may not perform well on faces with heavy occlusions or low resolution. Additionally, the model may not be robust to changes in pose, illumination, or facial expressions. VGG Face has been used in various applications, including biometric identification, surveillance, and human-computer interaction. The model's high accuracy and robustness make it suitable for use in real-world scenarios, such as border control and law enforcement. VGG Face is a powerful model for face recognition and has been widely adopted in various applications. Despite its limitations, it continues to be a critical reference for researchers in the field of face recognition.

While AlexNet and GoogLeNet are pertained model which are trained on various object images, VGG-Face network is specifically designed for facial images problems. This DNN model is special variant of VGG-16 that is trained on face images only. That gives it an outstanding capability for feature extraction from faces. In this model, global "Average Pool" is responsible for feature extraction instead of classification

layer. Finally, this Deep Learning Architecture has totalled, 13 layers and the feature vector have a length of 512 [180].

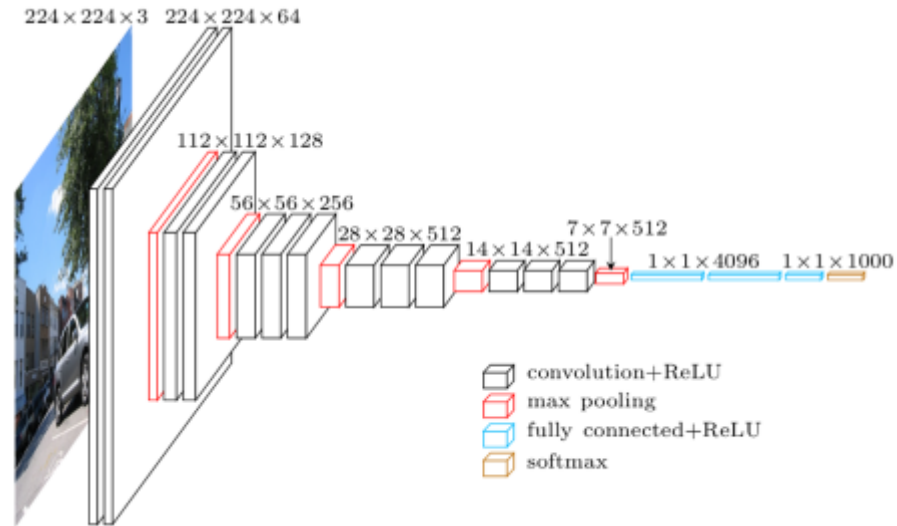


Figure 26 VGG Architecture [Source: Simonyan, K. et. al. 2014] [179]

As displayed in Figure 26, the architecture starts with a series of convolutional layers, which are used to extract features from the input image. VGG accepts an RGB image with “224 x 224” pixels. To maintain a uniform input image size for ImageNet [20] competition, central “224 x 224” patch from each image is cropped.

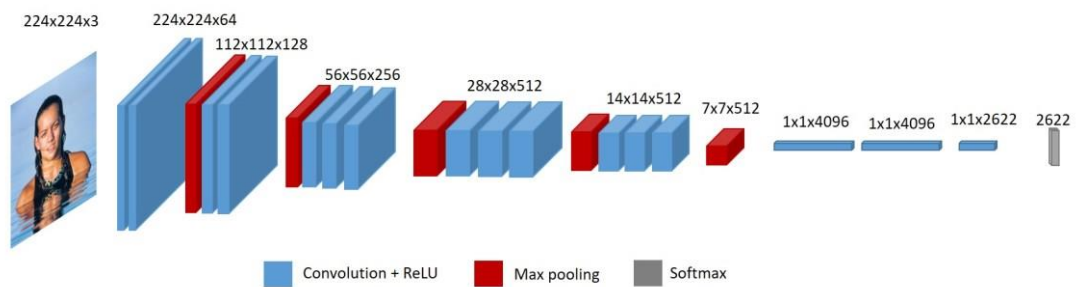


Figure 27 VGG Face Architecture [Source: Sefik Ilkin Serengil 2018] [181]

As shown in the Figure 27, the architecture starts with a series of convolutional layers, which are used to extract features from the input image of 224 x 224 x 3. Each convolutional layer consists of a set of filters that are convolved with the input image to produce feature maps. The architecture of VGG Face is designed to capture the

hierarchical representations of faces, which are crucial for face recognition. The combination of deep convolutional layers, pooling layers, and fully connected layers enables the model to learn high-level features that are robust to variations in pose, illumination, and facial expressions [181].

IV. ResNet-50

In addition to AlexNet, GoogLeNet and VGG-Face, ResNet-50 was used in the facial hair classification. Based on convolution layers, ResNet, which is based on convolutional layers, was presented in 2016 at ILSVRC and had a top 5 error rate of just 3.6%. which was inferior to AlexNet, GoogleNet, and inception. The goal of ResNet is to use residual blocks to solve the vanishing gradient problem. As further layers in another deeper network cause the gradient to evaporate, this is an issue. Practically speaking, increasing the number of layers does not guarantee greater accuracy. In many instances, adding more layers has a larger error rate than shallow networks. Connecting the output of the previous layer to the subsequent ones is the key concept. This mapping just provides the relationship between the previous and subsequent layers and lacks any trainable parameters. To tackle the vanishing gradient problem and train deeper networks, it is highly helpful to think about connecting the output of the previous layer to the layers ahead. The term "Skip Connection" refers to the idea of joining layers in this fashion [182].

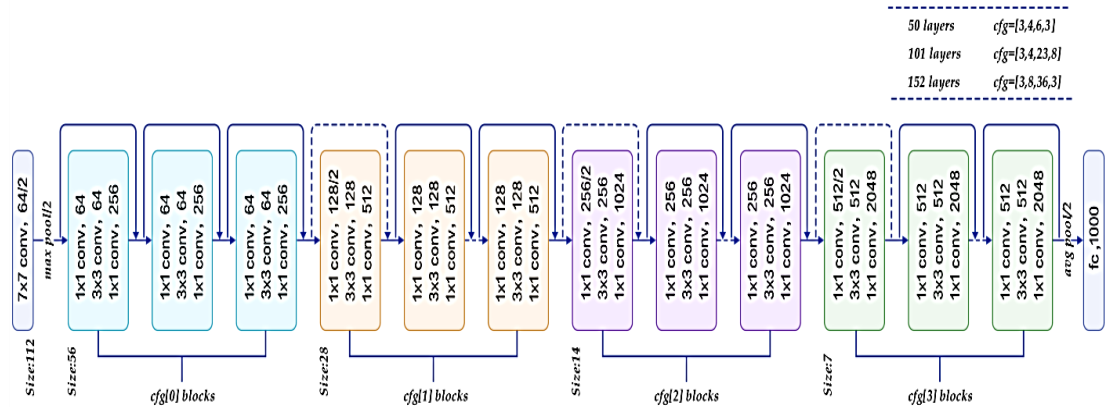


Figure 28 ResNet-50 Architecture [Source: He et. al. 2016] [182]

Details of the layers of ResNet-50 are shown in Figure 28. It was selected after evaluation of the results in comparison with the other techniques for facial hair classification. Results are shown in the results section.

2.14 Deep Learning Frameworks

A key component of the Fourth Industrial Revolution of today is deep learning, a subfield of machine learning. Deep learning technology, is derived from artificial neural networks and has become popular in the computing world due to its capacity for learning from data, is used extensively in various applications, like health sector, visual recognition, text analytics, cybersecurity etc. In actual, requirement is dynamic and continuously changing, that is why creating an acceptable deep learning model is tough job to do. In addition to changing behaviour, deep learning has become like black box due to limited basic concepts, which does not allow the growth at certain level [183].

Among machine learning approaches, neural networks constitute a subset. Instead of being accurate representations of the brain, these networks aim to mimic complex issues with reliable algorithms and data structures. Units (neurons) in neural networks are arranged in layers. Input layers, hidden (middle), and output layers are the three main types of layers. Neural networks can be categorised as shallow networks (one hidden layer) and deep networks (multiple hidden layers). This multi-layer hierarchy of neural networks provides the power to predict correctly. A network can learn how to make predictions by properly training it to show inputs as features to various scales, gather these features into higher order feature representations, and relate them to output variables [184].

When using neural networks to solve problems, the network design and training procedure are two related issues that must be resolved to obtain high prediction accuracy [185].

Since matrix-based operations are at the foundation of many machine and deep learning solutions, many-core accelerators like the GPU's massively parallel design enable them to accelerate these computations. Manufacturers frequently supply accelerated libraries, which give highly optimised primitives, algorithms, and functions to leverage the massively parallel capabilities of GPUs, as well as the option to upgrade hardware configuration with additional cores GPUs for better performance. The accelerated libraries that are most usually referenced in this section.

NVIDIA has introduced CUDA (Compute Unified Device Architecture) that is the parallel computing platform and programming model that can be used for general computing [186]. Drop-in acceleration in many domains, including linear algebra, image and video processing, deep learning, and graph analytics, is made possible by GPU-accelerated CUDA libraries. A development environment is offered by the NVIDIA CUDA Toolkit for the creation of high-performance GPU-accelerated applications [187].

I. The Deep Neural Network Primitives

Deep Neural Network primitives are GPU-accelerated using the NVIDIA CUDA Deep Neural Network library (cuDNN). For common procedures like forward and backward convolution, pooling, normalising, and activation layers, the cuDNN offers finely tuned implementations. It enables deep learning users to spend more time on neural network training and software development and less time on low-level GPU performance adjustment. Many deep learning frameworks, including Caffe2, CNTK, TensorFlow, Theano, PyTorch, and MATLAB, utilise the cuDNN [188].

II. TensorFlow

TensorFlow framework is designed and created by Google which has been available for training and building the deep learning models. TensorFlow library has been used for distinct types of computational tasks having data flow graphs. In data flow graphs, nodes and edges are applied to represent mathematical operations and multidimensional arrays data. These data arrays can also call tensors which are used to communicate in edges. The research in [189]. explains the structure of data movement in graph and processing node application. In the graph, each node explains mathematical operation between data arrays. There are several types of applications which have been used TensorFlow framework having high accuracy and computational speed. In the paper [190]. TensorFlow is used for gesture recognition and the effect of recognition using convolutional neural networks.

III. PyTorch

PyTorch is an open-source Facebook deep learning framework of predictive modelling for data science developers. It is used to work and perform computational tasks in an efficient way on graph-based models for differentiation [191]. Domineering programming and dynamic graphing are the distinguished characteristics of PyTorch in deep learning frameworks. It works in an analogous way as NumPy implements on neural network. PyTorch models are being applied for predictions and multilayer perceptron (MLP) and convolutional neural networks (CNN) [192]. In the layout of deep learning models' nodes are connected for outcome results, in which all the input nodes are linked and connected to corresponding layers and networks to propose the entire computational task [193]. MLP model is often reasonable for tabular data, although there have been other predictive models which identify problems using MLP. Most of them are multiclass regression and classification as well as binary classification. There are diverse types of datasets which have been used for classification and regression [194].

IV. Keras

Keras is an advanced neural network library framework which operates on Theano, TensorFlow etc. Keras allows extremely fast and easy prototyping on CPU and GPU. It works using the Python language code which is very extensible to use [195]. It has been vital for optimized actions and user-friendly feedback for errors. It is composite for building blocks during configurations having less restrictions. APIs are often consistent and easy to minimize the actions of user. It can be extended by writing required ideas of building blocks. Keras is unique in terms of machine learning because it has been significant applications using single API on multi-level machine learning frameworks [196]. It builds a special neural network which can be controllable and adaptable to watch and work. It integrates the frameworks using several machine learning frameworks at the same time of execution. Features in Keras can be distinguished as it can work using datasets of Pandas library. The code language is quite simple and easy for drafting as compared to other frameworks. Keras works for high level details and prevents to low level and explicates the results in categorical way. GPU based coding can be easily accessible for each framework and library [197].

V. Caffe

Caffe is C++ programming-based framework and designed having architectural features of MATLAB and Python. It works on new application-based models and innovations by using both CPU and GPU having single flag [198]. The latest advancement of its speed makes it to process and executes more than 60M images in a single working day. It has been implemented on both industrial and educational ideas and projects using various kinds of neural networks. Caffe works on individual source code instead of direct technique [199]. It can be drawback to deploy it. It doesn't support higher level of APIs for experimentations with its source code except to work with middle to low level API's. Due to C++ based programming, it runs and executes single layer configuration of GPU. Therefore, for a multi-level layer it is recommended to work initially on breaking strategy to its individual nodes. Caffe framework can be more appropriate for edge deployment computational tasks. The overall performance is greater than other frameworks in terms of internal criterion [200].

VI. CNTK

Cognitive toolkit is developed by Microsoft which has been open source. It is a complete package to build and design a deep neural network by using initial building blocks. It is developed by writing C++ and Python along Java and C#. It works on predictions having low level and high-level API [201]. It has the feature to work on different components for monitoring and debugging the neural networks. It does work on prediction at component of networks within the online applications to provide micro services. CNTK can work on C# and Java at the same time if not having the features in Python version of CNTK. Mathematical operations can have low level API for the building of neural networks and automatic process of differentiation [202]. The scalability and performance are better comparatively. It is also very efficient and smart in case of resource practice. It can be integrated to Azure Cloud to increase the systematic competence. But it has limited community to implement and use in computational applications.

2.15 Parallel & Distributed Computing Platforms for Image Processing

Closed-circuit television (CCTV) systems have been widely employed for surveillance and are growing in popularity every day. The most typical objective of CCTV systems is the tracking of things to prevent crime and disruption [203]. Image processing problems solving with deep learning requires huge amount of computational power in terms of training the network as well as running the model. Real time image processing is also a very costly task to perform. In the problem statement, a technique is required to handle multiple imaging devices i.e., CCTV cameras. Multicore processors including GPUs and multiple computing devices can solve the issue.

I. CPU vs GPU

Computers were designed to boost-up computation speed and help to perform repetitive mathematical tasks. Vast applications of the computing systems have now transformed computers into high performance machines capable of processing huge data [204]. According to conventional architecture known as Von Neumann architecture, tasks are performed in series called Single Instruction Single Data (SISD). Instructions are executed in steps; decode instruction, fetch data, perform instruction and writing result. These tasks are performed by CPU. CPU consists of registers, coder unit, ALU and memory interface [205].

In general purpose computers performance was enhanced by increasing the clock speed of computer. Chip size has also been decreasing with increased number of transistors. But it has a trade off with dissipated heat. Now multi-core chips with better energy efficiency are being made by the manufacturers. Each core of multi core system may not be as fast as single core systems but in combination they can perform much faster computations [206]. Parallel computing can be done in many ways to save the time cost of the process. It can be performed by introducing multi core processors, multi processors, clusters, and grids of computers. Intel and AMD both offers multi core processors [207].

For predicting the theoretical speedup acquired by use of multiple processors, Amdahl's Law is used. It gives the speedup gain of parallel computing. If N processors are used in parallel for f fraction of task. Then speedup is given by S [208].

$$S = \frac{1}{1 + (1 - f)/N}$$

Eqn. 2-2

Recently computational accelerators have captured the market with their high speed of processing. Computational accelerators attached to general purpose processors can give 10 times faster performance. Graphical Processing Units (GPU), General floating-point accelerators and Field Programmable Gate Arrays (FPGA) are used for enhancing the speeds [209].

GPU has several multi processors and each with many streaming cores having common single instruction decoder and memory interface. They are suitable for matrix and vector operations [210].

GPUs have simple processors but in massive amounts. They have limited amount of memory with high processing speeds. Their internal buses are fast to carry operands and results. High computation rates are required for the computer graphics those are performed by GPUs. High Performance Computing (HPC) also involves similar computational demands that made GPUs part of the HPCs. Relatively GPUs are cheap because of their mass of sale in general market for graphical purposes [211]. GPUs are now being used for high-speed calculations instead of CPUs because of GPUs memory bandwidth and arithmetic throughput that can utilize large number of parallel cores to perform parallel threads up to tens of thousands [212].

NVIDIA and AMD are two big players in the field of GPU manufacturing. Several basic processors operated by streaming processor works in parallel. In different GPUs architectures, number of processing elements varies. In each processing element there are arithmetic units and there are number of registers and shared on chip small memory of few KBs in each streaming multiprocessor. Where GPUs have large main storage present in off-chip GPU global memory [213].

Restricting to NVIDIA, NVIDIA provides a special platform for its programming its GPUs called CUDA (Compute Unified Device Architecture). Programming languages are like C platform [214]. GPUs are used through platforms like OpenCL and CUDA. CUDA's performance is 30% better than that of OpenCL in majority of the applications. In few comparisons claimed to be fair, shows that both have similar performance. OpenCL is portable as compared to CUDA providing good substitute [215].

Many general programming applications like Medical Imaging, Image and Video Processing, Computer vision etc. uses GPU devices [216]. GPUs are also used in face recognition applications. Results shows by Cheng, K.T. that a mobile GPUs can speed-up the task by 4.25 times with far less consumption of energy [217]. Machine Learning algorithms have also been implemented on GPUs. Many approaches of SVM training had been implemented on clusters for parallel computing. Experiments done by [218] shows that time for training is reduced to 5 to 24 times for classification as compared to SVM classification performed on CPU [218].

MATLAB 2017 and above offers solution to avoid reprograming in CUDA by generating optimized CUDA code from MATLAB code. NVIDIA CUDA libraries are used in the generated CUDA code. These libraries include cuBLAS, cuDNN and cuSolver. These libraries are integrated to the project. Thus, CPU Coder™ provides easy access to the GPUs for accelerating intense computational parts of the codes [219].

II. Single-Board Computers

With the advancement of technology, multiple microcontrollers are available today. These chips can be used in the development of signal processing as well as simple embedded systems. But one should know the right device for the right application [220]. Previously, microcontroller used to have separate electronics to perform operations, but now single board solutions based on powerful microcontrollers are available. These boards are widely used in automation industry as well as in the household appliances. Microcontrollers available can be distinguished on the basis of their features and developer should be aware of these features and requirements of the application [221].

The topic of selecting a hardware platform for industrial applications arose since intelligent image processing activities are relevant in contemporary settings. The challenge is contrasting several platforms that vary in size and computational power, as well as pertaining to various pricing categories [222]. Modern computing platforms include Raspberry Pi 3 B+, Raspberry Pi 4 B+, NVIDIA Jetson Xavier, NVIDIA Jetson Nano, HiKey970 module, Intel® Neural Compute Stick 2 etc.

(A) Raspberry Pi

Raspberry-Pi is one of the most capable and popular electronic prototyping boards available today. In terms of speed and effectiveness, it functions like a minicomputer. It can be applied to a variety of disciplines to create various real-time applications that are helpful to us in our daily lives. Image processing is the process of analysing and extracting useful information from an image for one's own needs [223]. An ARM CPU and various operating systems, including RISC OS, ARCH LINUX, OPENELEC, RASPBMC, and RASPBIAN, power the Raspberry Pi. Linux is gradually being used as the operating system for all portable electronics. Even mobile devices like Android phones run an operating system based on Linux. Linux is an extremely adaptable and versatile operating system because it is open source. Not all of these, though, are portable and appropriate for Raspberry Pi. As a result, the Linux-based Raspbian OS is chosen, and anyone may easily install it without any prior technical experience. When a monitor, keyboard, and mouse are connected to it, it can function as a desktop computer. Additionally, it can be connected to a local area network using an Ethernet connection or a Wi-Fi adapter so that the user can log in using an SSH client like Putty. All of its operations and scripting are performed using Python as the core programming language. Among the languages supported are BASIC, C, C++, JAVA, Perl, and Ruby. When compared to a traditional computer, the Raspberry Pi has a notable advantage because it has readily accessible GPIO connections that allow it to connect directly to buttons, motors, sensors, and other devices. Utilizing Raspberry Pi, numerous programmes and prototypes have been created [224]. Raspberry Pi was first launched in 2012 and upgraded on regular basis. Table 1 shows specification of each Raspberry Pi model along with the launch year.

Table 1 Models Specifications for Raspberry Pi [Source: <https://www.raspberrypi.com/>]

Name	Variant	Memory (MB)	RJ-45	Wi-Fi	GPIO Pins	Year
Raspberry-Pi	B	256	Y	N	26	2012
Raspberry-Pi	B	512	Y	N	26	2012
Raspberry-Pi	A	256	N	N	26	2013
Raspberry-Pi	B+	512	Y	N	40	2014
Raspberry-Pi	A+	512	N	N	40	2014
Raspberry-Pi Zero	Zero	512	N	N	40	2015
Raspberry-Pi 2	B	1024	Y	N	40	2015
Raspberry-Pi 3	B	1024	Y	Y	40	2016
Raspberry-Pi Zero	W/WH	512	N	Y	40	2017
Raspberry-Pi 3	A+	512	N	Y	40	2018
Raspberry-Pi 3	B+	1024	Y	Y	40	2018
Raspberry-Pi 4	B	1024	Y	Y	40	2019
Raspberry-Pi 4	B	2048	Y	Y	40	2019
Raspberry-Pi 4	B	4096	Y	Y	40	2019
Raspberry-Pi 4	B	8192	Y	Y	40	2020
Raspberry-Pi 4	400	4096	Y	Y	40	2020
Raspberry-Pi Zero	2 W	512	N	Y	40	2021
Raspberry-Pi Pico	N/A	264 kB	N	Y	40	2021
Raspberry-Pi Pico	W	264 kB	N	Y	40	2022

(B) NVIDIA Jetson

One of the most popular accelerators for the inference stage of machine learning, NVIDIA's Jetson is quite promising. Jetson has a CUDA-capable GPU that can be readily customised to accelerate challenging machine-learning operations. The CPU can boot the OS on this GPU heterogeneous architecture platform. Additionally, it is an excellent solution for situations where weight and power are limited because to its tiny form factor, low weight, and low power consumption. However, optimising Jetson hardware and NN algorithms is necessary to fully utilise Jetson and achieve real-time performance. Numerous recent works have attempted to overcome these issues [225]. Table 2 shows specification of NVIDIA Jetson variants.

Table 2 Models Specifications for NVIDIA Jetson [Source: <https://www.nvidia.com/>]

	Jetson Orin Nano series		Jetson Orin NX series		Jetson AGX Orin series		Jetson AGX Orin Developer Kit
	Jetson Orin Nano 4GB	Jetson Orin Nano 8GB	Jetson Orin NX 8GB	Jetson Orin NX 16GB	Jetson AGX Orin 32GB	Jetson AGX Orin 64GB	
AI Performance (TOPS)	20	40	70	100	200	275	
GPU	512-core NVIDIA Ampere architecture GPU with 16 Tensor Cores	1,024-core NVIDIA Ampere architecture GPU with 32 Tensor Cores	1,024-core NVIDIA Ampere architecture GPU with 32 Tensor Cores		1,792-core NVIDIA Ampere architecture GPU with 56 Tensor Cores	2,048-core NVIDIA Ampere architecture GPU with 64 Tensor Cores	
GPU Frequency	625 MHz		765 MHz	918 MHz	930 MHz	1.3 GHz	
CPU	6-core Arm® Cortex®-A78AE v8.2 64-bit CPU		6-core Arm® Cortex®-A78AE v8.2 64-bit CPU	8-core Arm® Cortex®-A78AE v8.2 64-bit CPU		12-core Arm® Cortex®-A78AE v8.2 64-bit CPU	
	1.5MB L2 + 4MB L3		1.5MB L2 + 4MB L3	2MB L2 + 4MB L3		3MB L2 + 6MB L3	
CPU Max Frequency	1.5 GHz		2 GHz		2.2 GHz		
Memory	4GB 64-bit LPDDR5	8GB 128-bit LPDDR5	8GB 128-bit LPDDR5	16GB 128-bit LPDDR5	32GB 256-bit LPDDR5	64GB 256-bit LPDDR5	32GB 256-bit LPDDR5
	34 GB/s	68 GB/s	102.4GB/s	102.4GB/s	204.8GB/s	204.8GB/s	204.8GB/s
Networking*	1x GbE				1x GbE		RJ45 connector with up to 10 GbE
					1x 10GbE		
Power	5W - 10W	7W - 15W	10W - 20W	10W - 25W	15W - 40W	15W - 60W	

(C) HiKey970

The developers building robotic systems and Internet-connected products are the target market for the 96Boards HiKey 970 Single Board Computer. The Huawei architecture consists of CPU, GPU and dedicated processor for neural networks, is the foundation upon which the specialised NPU AI Hikey970 platform is constructed (NPU). The platform supports widely used neural networks that utilise NPUs to accelerate their processing [226]. Specifications of HiKey are given in Table 3.

Table 3 HiKey970 Specifications [Source: <https://www.96boards.org/product/hikey970/>]

Component	Description
SoC:	HiSilicon Kirin 970 - HiAI Architecture, Dedicated NPU
CPU:	ARM Cortex-A73 MPCore4 @up to2.36GHz, ARM Cortex-A53 MPCore4 @up to1.8GHz
GPU:	ARM Mali-G72 MP12 GPU
RAM:	6GB LPDDR4X 1866MHz
Storage:	64GB UFS 2.1, Micro SD
Connectivity:	Bluetooth/WIFI/GPS
Video:	1080p@60Hz HDMI, 4 line MIPI/LCD port
Expansion Interface:	One 40-pin Low Speed (LS) expansion connector
	UART, SPI, I2S, I2C x2, GPIO x12, DC power
	One 60-pin High Speed (HS) expansion connector
	4L-MIPI DSI, USB, I2C x2, 2L+4L-MIPI CSI
User Interface:	Power/Reset
	8 LED indicators
	4 -user controllable
	3 -for radios (BT and WLAN and GPS activity)
	1 – for CAN
Power Source:	DC Power: +8V to +18V

(D) Intel® Neural Compute Stick 2

Without the need for bulky, expensive hardware, neural network capabilities can be accessed via the Intel Neural Compute Stick 2. It has an Intel Movidius Myriad X Vision Processing Unit (VPU) that makes it possible to add artificial intelligence (AI) and computer vision to IoT and edge devices. For high-performance vision and AI inference applications, the most recent version of Intel VPUs has 16 potent processing cores (referred to as SHAVE cores) and a dedicated deep neural network hardware accelerator—all at low power [227]. Table 4 shows the details of Intel® Neural Compute Stick 2.

Table 4 Neural Compute Stick 2 by Intel Specifications [Source: <https://www.intel.com/content/www/us/en/developer/articles/tool/neural-compute-stick.html>]

Component	Description
CPU	Intel Movidius Myriad X Vision Processing Unit
Frameworks	TensorFlow, Caffe, Apache MXNet, (ONNX), PyTorch & PaddlePaddle

2.16 Intelligent Assistance Systems

Automated systems typically comprise complex interaction of machines and humans. This is especially the case for systems having complex scenarios in terms of safety and security like aerospace, power plants etc. Human errors can lead to big disasters. Design problems and employee operation mistakes led to the Chernobyl Nuclear Power Station's 4th Unit reactor explosion [228]. Just 50 miles west of New Delhi, India, in November 1996, Saudi Arabia Airlines Boeing 747 and Kazak Ilyushin IL-76 collided in mid-air. The 349 people aboard both aircraft—passengers and crew—died in total, making it the third deadliest air catastrophe in aviation history. One of the reasons was language barrier [229]. Similar issues arise in the war field where target selection and engaging is a challenge. According to research, time-pressed war fighters in the air and on the ground are prone to a variety of psychological biases that prevent them from fully weighing all pertinent information, which frequently results in devastating outcomes [230].

Not only humans controlled but machine centred systems without the presence of humans can be fatal in some cases. Autonomous vehicles, which allow a driving automation system to substitute a human driver to handle the vehicle with improved identification, decision, and driving skills, have been developed to increase road traffic safety [231]. There are inherent obstacles regarding safety and security for autonomous cars. The in-car network is affected if one autonomous vehicle component malfunctions or is attacked, which could lead the on-board computer to send the incorrect command and so directly endanger traffic safety [232].

Keeping in view the hazards of totally automated systems in the critical systems, integrated human-machine systems came into existence those are also referred to intelligent assistant systems [233]. Intelligent assistant systems are complex systems that need various expertise like statistics, programming, machine learning and human-computer interaction [234]. There are various Intelligent assistant systems currently being used to make lives of their intended users easier. A mobile device app, an intelligent walking stick, wearable smart glasses, and an online information platform make up the intelligent support system for blind and visually impaired persons is available. People who are blind or visually challenged can detect impediments by

using the proposed intelligent walking stick and smart eyewear. When a person who is blind or visually impaired falls, the relevant data (GPS, fall down, etc.) is collected and uploaded to the online information portal. The suggested mobile device application allows users to examine related data as well [235]. In order to provide intelligent human computer interaction system and medical help, a natural and ergonomic hand gesture system is created. The technology can operate in real-time for practical purposes and is resistant to drastic changes in illumination [236]. Siri: Made by Apple was the first personal virtual assistant, Alexa: Made by Amazon is digital assistant having Echo home device, in market since June 2015, Cortana: Made by Microsoft is "digital agent." manages calendars, can acquire weather details and dictation on a document, Assistant: Made by Google is like Siri [237]. AI based face recognition system for "Virtual Assistant" is recently proposed, this system allows only authorized users to gain access and helps to monitor who trying to penetrate into the system [238]. Intelligent driving assistant systems can make conventional driving considerable safer and convenient. Many safety supporting devices are installed outside and inside the vehicle. It can perform many operations like detecting front-object under driving condition, departure detection, detecting side-object while driving and back-object. It can mention alarm according to detection results from different elements. It can store the real-time image according to the detection and then transfer the associated information to other places for real-time notification with matching the information with GPS [239]. There has always been a space for betterment in the field of research. Kaizen is a Japanese philosophy about gradual but continual improvement and moving towards excellence not ensuring achievement of perfection). It is used in various corporations such as by Toyota. Kaizen essentially links to the six-sigma approach for quality to reduce errors. Similar technique is practiced in IT. Being a human-centred approach it is useful in a human-machine integrated system improvement [240]

Similar intelligent assistance system is aimed to be developed in this research as already discussed in Chapter 1, Section 1.04

Chapter 3

Methodology

3.01 Overview

This research project started in response to finding a solution to assist with the problem of the large amount of human operator time that is needed to do effective manual surveillance of security camera footage, particularly to assist with the tracking and monitoring done by the Punjab Safe City Authority. The need of focus is real-time detection of ‘persons of interest’ in the camera footage and being able to search for known persons of interest in archived surveillance videos. Accordingly, this research aims to develop an intelligent assistant system that can integrate with the currently installed CCTV networks in cities with minimum hardware additions that can help human operators efficiently track or search for targets (i.e., known persons of interest). This is to be done through a process of classifying the public (such as potential matches), while maintaining privacy – as well as boosting monitoring efficiency – by restricting operators to view only target frames. This system is also to be extended to assist the human operator in i) estimating the colour image from man-made sketches, ii) improving the resolution of blurred or low-resolution images, and iii) predicting the frontal face from a side-pose image. This system, as per the application domain focus, is designed especially for the Indian subcontinent with the ability to deal with potential scenarios of this region while keeping in view the local concerns of privacy that can be integrated with existing surveillance systems with minor hardware and software additions. This research was planned by identifying the current problems faced in the field of public security in the Indian subcontinent. Relevant literature was reviewed to build up the foundation and gain knowledge in the field from the previous work. The margin for new and beneficial contributions from this work was also gauged.

Stakeholders, including citizens and law-enforcement agencies of Lahore city, were consulted in planning this research. Details of the currently installed surveillance system were reviewed by visiting the PSCA office. The survey was conducted to get feedback from the 240 civilians at 8 different locations in Lahore city. (See Annexure

A) From the results, it was evident that the majority of the public has privacy concerns due to human-based surveillance cameras. All the participants positively showed interest in the intelligence-assistance automated surveillance system to get security without disturbing their privacy. The feedback from the public directed the research to develop the research statement. Ethical constraints were also considered and maintained throughout this project. Once the main research and development targets had been established, the system was conceptualized and then designed and developed, building upon existing software and hardware solutions. Datasets were an essential part of this research, which were gathered carefully and concerning ethical considerations. Details of the datasets are given in Sections 4.03 and 4.04.

Machine learning prediction models for detecting persons of interest were trained and evaluated. Classification layers were identified, and AI models were developed using multiple methods and were evaluated based on parameters such as accuracy, precision, recall, and F1 score among others. A specialized setup for experimentation was designed to replicate real-time scenarios; this was done partly because of these experiments being done during the COVID-19 pandemic by the need for enhanced health and safety restrictions of social distancing.

3.02 Literature Review

After having a clear understanding of the problem, it was time to find a solution that should be capable of dealing with the local scenarios. It started with studying the relevant recent work done in this field. There is much prior work in the literature regarding the field of face recognition and tracking systems. Accordingly, the conventional method for the literature survey was adopted in this research. This started with understanding the basics of image processing and image transformations. Applications of image processing in the field of biometrics were also studied. Biometrics involves the measurement and statistical analysis of people's physical and behavioural characteristics. These characteristics are unique to individuals and can be used for identification or verification purposes. As the research was only focused on facial recognition, details of the methods used in face detection and face recognition were reviewed. Various algorithms and techniques were analysed. Mathematical expressions were studied and discussed in the literature.

Artificial Intelligence is the future of science, and its applications in the field of face recognition were reviewed. Machine learning algorithms like fuzzy logic, decision trees, KNN, SVM, etc. were discussed. The other and more advanced subcategory, Deep Learning was thoroughly studied. The application of deep learning algorithms in the field of face recognition was the focus of this part of the literature. As this is the major area of our research is based on deep learning algorithms. Famous algorithms RCNN, Fast RCNN, Faster RCNN and You Only Look Once (YOLO) v2 and v3 were reviewed and YOLO v3 was selected. Deep learning models along with the frameworks were also reviewed.

As this research requires multiple computations so need for parallel computing was identified at the early stages. Parallel computing techniques were discussed and the introduction of GPUs for the parallelization of the processes was highlighted. Structural design comparisons between CPU and GPU were reviewed. Various single-board computing devices are also reviewed to attain cost-effective distributed computing. This research was conducted to help humans, not to eliminate them by putting their jobs at stake. In this regard, the concept of Artificial Assistance was described. That helps humans to perform their jobs efficiently and makes them able to achieve impossible goals with the assistance of the machine. This sums up the part of the literature review, which helped to understand each component of the research. A few parts were intentionally left for discussion at later stages in the System Design. So, to have quick relevance while justification of each component of the system.

3.03 Organizational Information

Initially, the operations of the relevant organizations including the Punjab Safe City Authority and Police Department were studied with the help of publicly available information. To find out more about the organization it was decided to visit the PSCA headquarters located in Lahore. Two higher officials of PSCA are active members of the IEEE Lahore section and they were contacted to arrange this visit. An open-ended discussion was held to find out the details regarding this organization. PSCA system only contains cameras with high resolution and are connected via an optical dedicated fibre link. All the surveillance is done in the command centre manually. They have the face recognition software that can be operated manually. After the incident, the frame

is cropped by the operator who passes this image to the database. So, there is no system for tracking the target. It was found that many allegations were made due to privacy concerns because of the human involvement in the monitoring. Few are included in the introduction section. This system is also used for traffic management and issuing e-challan. But unfortunately, all the systems behind it are also manual.

After having enough understanding of the system. The national database of the citizens was to be studied. NADRA has a record of every citizen with pictures. They have the family database very well maintained so that all the family members can be tracked. Moreover, data regarding education, income status, banking details, tax details, etc. are available in the main database and connected to other national agencies. These details are often accessed by law enforcement and forensic agencies for solving their respective cases.

After learning the system details of PSCA and NADRA, the next question was how forensic scientists find the targets. For this reason, Asia's finest forensic agency i.e., Punjab Forensic Science Agency was contacted. 4 graduates of the Department of Electrical Engineering, University of the Punjab are working there and were contacted in this regard. Questions were asked to get an understanding of the facial features and types of faces for estimating a face. These features were to be the foundation of our system design. Officials who are responsible for making sketches were interviewed. It was learned that before making a sketch or defining the suspect few things are being asked. The first thing is the gender of the suspect. It was said that Pakistan is a country where more than one ethnic group live. These are characterized based on their overall appearance. Usually, they categorize them into four types. These are discussed in detail in the System Design. Then facial hairs are considered as Pakistani people have a variety of facial hair styles. In addition to these distance between the eyes, lips, and nose is also observed. Depth of the eye socket is also considered but it is also addressed in the case of ethnicity. Skin colour is also considered but usually sketches are made in grayscale.

3.04 Public Feedback

After understanding the currently installed systems by gathering available information from the relevant organizations like the Punjab Safe City Authority and Police Department. Key locations of the city were identified, and the human traffic at these spots was observed. Camera positions, angles, and ranges were observed. A brief questionnaire was given to the random 240 random people at 8 separate locations to get a public view. A copy of the questionnaire is attached in Appendix A. The result was analysed based on gender and age groups. 90% of females and 70% of males responded that they felt safe after the installation of PSCA cameras. On the other hand, 83% of females and 23% of males said that these cameras on the main road are a threat to their privacy. Whereas 95% of females and 87% of males said that installing cameras in the streets will be a concern of privacy. Only 3% of females and 12% of males have acquired help from PSCA for solving their case whereas 21% of females and 45% of males have heard that PSCA had solved any case with the help of the footage. 2% of females and 5% of males experienced a terror attack within the range of 1km. Due to recent news regarding the video footage scandal, 99% of males and females knew that these cameras were being monitored by humans manually. 71% of females and 92% of males think there should be more cameras installed. All the participants wanted an automated system to track the targets without putting privacy at stake.

3.05 Ethical Considerations

Images of closed-group participants with their consent and open-source images are used for the preparation of datasets. These images only helped in the development of the models of each class. They cannot be further accessed by anyone. Once we got the model with us, we had an average face. For the sake of the recognition part, research students will be used as samples, and for the testing phase, we may add all students of the faculty of engineering and technology by their written consent. The consent form attached in the appendix was signed by the students and staff of EE, PU so experiments may be performed within the premises of EE. As discussed in the introduction, in Lahore, the crime rate has significantly reduced after the installation of CCTVs. In this part of the world, every single person is bound to give his/her biometrics to the

government. This system will further be used by government agencies, and they have the right to surveillance. This system is a closed system, and it won't be possible to sneak in by an infiltrator. The security part is not in the scope of this research, so it is left to implementing authorities. As discussed in methodology, face occlusions can be treated as threats and alarms may be generated to look for some entity. Being an Intelligent Assistant System, it may not risk the employment of security personnel, but it will help them to increase efficiency. This system is helping to achieve undoable tasks by humans not replacing them. If you look at the police patrols in Lahore, Pakistan it has many branches like Counter Terrorism Department, Criminal Investigation Agency, Special Protection Unit, Punjab Elite Force, Punjab Boundary Force, Punjab River Police, Punjab Traffic Police, Punjab Highway Patrol and Dolphin Force. Patrolling units are equipped with cameras that can send the video stream back to the command centre. These cameras can also be used as additional nodes to assist officers. Safety will be increased by this system. It may be noted here that, in publications, pictures will be published after getting written consent from the samples.

3.06 Corroboration of Novelty

The novelty of this research was the ability of the system to deal with the people of Pakistan and its ability to work with the current system installed in Lahore city. Moreover, this system must be cost-effective and unique. This research has few novel contributions to the knowledge. It includes the new classification system in a cascaded manner that includes determination of gender, face type, and facial hair styles then it searches into the library of individuals having those specific characteristics. This combination helps to speed up the process of recognition. Distributed computing systems with multiple GPUs have been introduced for parallel computing. Each node usually has 4-6 cameras and all of them deal with the same Raspberry Pi unit. It means for the whole city multiple Raspberry Pi modules will be working in parallel. This scheme is an addition to the area of face recognition systems. Furthermore, GPU(s) are proposed in this research at the second processing stage. It is also proposed that more central systems can be added in the regions to speed up the process. It may be noted that a complete system can represent the effectiveness of the design. New pre-processing stages are proposed in this research for better detection and recognition

rates. Details of the processing blocks are already discussed in the system design section. Furthermore, by adding a layer for detection, this system can assist the operators by popping up alarms in case of some suspicious detections (persons with occlusion). This ability will make this system an IA (intelligent assistance) system. Once the database is available, queries can be trimmed to find out more relevant results.

3.07 Hardware and Software Acquisition and Integration

In the preliminary phase, the basic system was conceptualized, and comprehensive requisites concerning both hardware and software components were defined. It would not be wrong to call MATLAB the most powerful software tool available for research. Knowing the capabilities of MATLAB in the field of image processing and deep learning, it was decided to use it as the main platform for prototyping the system. This research was started in somewhere mid-2016. At that time many of the features were not introduced in MATLAB 2016a. MathWorks was joined to keep its knowledge updated and get newsletters regarding the latest updates. Few webinars were attended, and, in these years, MATLAB introduced many powerful toolboxes for Deep learning. It helped a lot in saving time for learning a whole new platform. Python was used for the development of the final model as few models were not available in MATLAB directly and required indirect techniques. Support libraries and frameworks were also required for Python. It includes TensorFlow-GPU, Keras, CuDNN, CUDA Toolkit, etc. Raspberry Pi does not involve much computation except frame capturing with the hardware trigger control which is why Octave GNU was used.

For the hardware part, first, a specific working space was required. In this regard, a whole new lab was established known as the “AI & Remote Sensing Lab”. This lab is equipped with several types of cameras, computing devices, GPUs, and allied setups. Network CCTV cameras were required for the experimentation. 4 MP and 8MP network CCTV cameras were used. Multi-port network switches were also used for network connectivity among the nodes. Details of each hardware are given in the system design chapters. More than four computing platforms with different generations of CPUs and GPUs were changed in the process as the computations were

getting more complex as the system got closer to the final shape. The final computing machine was Acer Nitro with 6GB RTX 2060.

3.08 Dataset Preparations

CCTV footage was required for the preparation of the dataset to train the face detection model. Small-sized face images had to be detected from the CCTV camera frame. Open-source CCTV footage dataset was available and included in the training images. In addition to the open-source dataset, custom training images were also introduced. It was accomplished by engaging undergraduate students and a consent form was made and got signed by the group of people who volunteered to participate in the research. It was communicated to the participants that their images would not be shared or published on any of the platforms without their prior permission. It was also agreed that images would be deleted after training of the models. The images used in this manuscript only include the supervisor, PhD candidate, and members of the Remote Sensing Lab with their consent. As discussed earlier, this research for face categorization focuses Pakistan region only. That is why the system was to be made keeping the parameters of the region in view. No specific research was made in this field for this region. A new dataset was required for training the models. Few open-source datasets were used along with the manually collected open-source images for the classification system. Details for the dataset classes are elaborated in the design chapter.

Once the image collection was done the images were processed so that they could be used for training data. The algorithm was designed to create the library to maintain uniformity and save time. Image augmentation is performed to enrich the library. Details of the processing applied to the images with the justifications are given in the design portion of the document.

3.09 Frame Capturing and Face Detection

The initial system was designed for single-node computations. Basic operations on various images were performed. It was done to access the result of the small functions that form part of the bigger system. Acquisition devices were integrated with the computing systems. The frame rate was calculated in such a way that each person could be captured in at least one frame. All the calculations are shown in the design section. Calculations were performed to choose the right resolution. Further mathematical operations were performed to optimize the settings so that the output efficiency is not decreased and save the machine cycles of the processing nodes. Frame storing after pre-processing of the image was implemented on Raspberry Pi. A passive Infrared (PIR) module that can detect object movement, was integrated with Raspberry Pi as a triggering source to avoid duplication using hardware. This can also be done with software, but it will consume important cycles.

Cropping the face out of the frame after detection has been done was a major task. A few more operations are to be performed after face detection. That is why the marginal area was to be calculated with the help of experiments and data available. These criteria were set for the cropping out face so that it can be useful for all three upcoming layers for recognition.

Capability of each stage was evaluated. Pre-processing algorithms were developed to increase the detection efficiency of the system. Viola Jones algorithm was chosen for the initial evaluation of the pre-processing algorithms. Further replaced by YOLO v3 trained model.

3.10 Face Classification System Design

While defining the outlook of a person few parameters are considered including gender, ethnicity, facial hair style, body type, etc. Moreover, for tracking purposes, the location where the person was last seen along with the date and time is required. This statement is also matched with the information provided by the forensic scientist. Keeping in view the above, the demographics of the country under study that is Pakistan were observed through literature. Gender parity, ethnicity division, and facial hair styles commonly adopted in the country are studied. Then classification methodologies along with the software tools were analysed. Regions required for

extracting facial features and facial hair type prediction were calculated with the help of data. This helped to crop the relevant part of the face for each layer. Classification systems were created with the help of deep learning algorithms in MATLAB initially and then implemented in Python. The cropped face images undergo pre-processing to inspect for illumination, colour, saturation, and other factors. So, for such modifications, fewer calculations are needed right now. Different convolutional neural network architectures were trained using transfer learning on the meticulously gathered and prepared dataset. Models for target classification used include GoogLeNet, AlexNet, ResNet, and VGG-Face. All the systems were integrated for prototyping the main system. Based on predetermined criteria, such as the confusion matrix, accuracy, precision, recall, specificity, F1 score, Precision-Recall curve, and ROC, the trained models are evaluated. The result of each architecture for all the types of classification was then compared and the best-performing algorithm was selected.

3.11 GPU Computing for Training and Prediction

This research includes massive amounts of data to be processed. Multiple repetitive operations were performed on large datasets such as cropping, alignment, and image histogram corrections. Additionally, there were also more complicated tasks that needed to be done for the research. This included training the deep learning networks with thousands of images. For deep learning networks, the training time required is much higher due to convolutional calculations to be performed in multiple layers. For such enormous computations, CPUs are not enough to perform the tasks efficiently. To overcome this limitation of the CPUs, GPU computing was adopted. GPUs were evaluated on how they can benefit in terms of solving the classification problem in real time. The programming environment to enable GPU computing was studied. Additional hardware and software support requirements were accessed. NVIDIA GTX 960 was used for initial testing, and for full-scale dataset training, RTX 2060 was used. GTX 960 has 640 CUDA cores and CUDA compute capability of 5.0 whereas RTX 2060 has compute capability of 7.5 and 1920 CUDA cores. General-purpose computing on GPUs is done for parallel computing and application programming interfaces. While coding in Python, the TensorFlow library was used for training the models with CPUs whereas TensorFlow-GPU was used to use the CUDA computing

platform. That requires further support packages to be installed. Development environment for high-performance GPU applications is offered via the NVIDIA CUDA Toolkit. One can create, refine, and deploy applications on GPU-accelerated embedded systems, desktop workstations, business data centres, cloud-based platforms, and high-performance computers with the help of the CUDA Toolkit. C/C++ compiler, runtime library, debugging and optimization tools, and GPU libraries are included in the toolkit. The fundamental building pieces needed to run neural networks are provided by the CuDNN library. Using CUDA, the cuDNN deep neural network library was created. It offers GPU-accelerated capability for deep neural nets' typical operations. It can be used directly, however TensorFlow, and other libraries have already developed abstractions that are supported by cuDNN. Using all these hardware and software supports, deep learning networks were trained. These models also run on the GPUs to make computations fast and solve the real-time classification problem. Multiple systems with GPUs can access the same database and work in parallel to classify the face images into the relevant categories. Computational speed can be enhanced by adding the computing devices working in parallel.

3.12 GUI & Database Management

This research focuses on the network of CCTV cameras where multiple frames from each camera have to be uploaded to the database that is accessible through the cloud. A prototype has to be built in this work, which can propose an efficient solution for classifying targets. A huge amount of data has to be uploaded to the server in the form of CCTV camera frames. These frames are uploaded by Raspberry Pi devices connected to cameras. Multiple high-performance computers with GPUs then further process the frames by picking out faces from the images. These frames are then passed through three classification layers and placed in the relevant folder. A step-by-step approach is followed. Specialized algorithm was designed for the creation of a database hierarchy for the placement of the images. A virtual server was made using XAMPP for prototyping the system. The system is designed in such a way that multiple computing devices can work together to solve a big problem. Then the last problem was to search for a target from the database. That requires a graphical user interface to communicate with the system. GUI for the system is designed using

Python Tkinter. Tk GUI toolkit has Python binding called Tkinter. It serves as the mandated default GUI for Python and is the official Python interface to the Tk GUI toolkit. Standard Python installations for Linux, Windows, and macOS come with Tkinter. This GUI integrates all the trained models and offers additional features. Input image transformation including Sketch to Colour Image conversion, Low resolution/ Blur image enhancement, and face alignment options are made available in the GUI. The next task for the system was to search the target from the database, the same GUI got this option. Search methods were studied, and the radial search method was adopted. Starting from the given camera node at a specific date and time it searches circularly for the nearest neighbour and finds among the relevant gender, face type, and facial hair style locations.

3.13 Experimentation and Evaluation

The experimentation requires two major parts, the hardware, and the scenario building. For the experimental setup, a place was required to replicate the real-time situations of the road. The CCTV camera footage includes a specific type of background and varying lighting conditions. It was decided to establish a similar setup inside the university. A vacant outdoor place was selected next to the Remote Sensing Lab. Arrangements were made to vary the lighting conditions, add lighting flares, and alter the targets. Research students were used as test cases. It was assured the test images do not contain the person whose image has already been used in the training images. The environment creation part only includes and verifies the face detection part. Further stages are not affected by the environmental variance. For further experiments only face images with transformed resolution were used. All the cameras used in the experiments do not have the same specifications as the cameras used by the PSCA. Distances of the objects and angle of depression were varied as per the specifications of each camera. It is covered in detail in the next chapters. Manual measuring tools were used for estimating the distances and the angles.

Using the carefully prepared dataset training was performed on various deep learning networks. Training such a huge dataset requires a long time. Two separate local computers with NVIDIA GTX 960 and NVIDIA RTX 2060 were used to save time. Pre-processing operations are mostly performed at GTX 960. For training models

using Multistage-Cycle GAN (MS-CGAN), Google Colab was also used. It provides free computation power of GPU for 12 hours per day. Keeping in view the limitation only specific models were trained using Google Colab. After training the models using YOLO v3, AlexNet, GoogLeNet, VGG-Face, ResNet, and MS-CGAN.

3.14 Analysis Methods

The next task was to analyse the performance of each model. Datasets are divided into two portions, one for training the data and the other for testing the data. Performance metrics were selected to gauge the results. Confusion matrices were used to display the accuracy and precision of the model showing the true positives, true negatives, false positives, and false negatives. Accuracies, precision, and F1 Scores were calculated with the established formulae for binary class and multi-class problems. These parameters are not the only way to assess the performance of the deep learning models. Precision-Recall curve was made with the varying threshold for considering true positives. The receiver operating characteristic curve (ROC curve) is displayed to estimate how well the classification model performs across all classification thresholds. Two categorization layers of the system had multiple classes, for that one class versus the rest of the classes were plotted. Mean Average Precision (mAP) is also used as an evaluation metric, in object identification and picture classification tasks. Intersection over Union (IoU) is also measured in object detection and image segmentation. Mean Squared Error (MSE) and Root Mean Squared Error (RMSE) are used to evaluate GANs-based models. The cosine similarity method is used to check the similarity between two images as a final matching index of two different images of the same person.

Training curves are included to depict the performance of the machine learning model performance over as a function of iterations or epochs. These curves show information on the model's behaviour during training and can be used to identify and avoid over- or underfitting. Experiments were performed in such a way as to evaluate the performance of each stage in terms of its efficiency. The time consumed for each task was critically monitored. Calculations were performed to reduce the input size without disturbing the overall performance. These calculations for the frame and face resizing are explained in the design chapter. Moreover, the area of interest for the

classification layers was also calculated. Details of the experimental setup are given in the experimental setup chapter.

3.15 Validity and Reliability

While designing the system it was important to evaluate the validity and reliability of the system. As discussed earlier this system deals with the tracking of targets from the network of cameras. Being a complex system, it involves multifaceted considerations. For validity, this system is currently valid for two genders, four ethnicities, and four levels of facial hair. Validity is made more robust by image augmentation used while training the models. The validity of the system can also be affected by local laws regarding the collection of video footage. Currently, this system is designed for the Pakistan region and has no such restrictions. Hardware is also an important parameter for gauging the validity of the proposed system. This proposed system is valid for the hardware meeting the minimum capabilities described in Chapter 4, System Design. The sketch-to-face generation system is valid for professional sketch images only.

Reliability encompasses the system's consistency and stability in its classifications. This requires verifying the system's consistency across multiple runs and environments, ensuring classifications remain stable over time. This system was found reliable for the classification of all three stages. The system is tested on various datasets and test cases available at the University of the Punjab. Details are shown in the experimentation and results section. Datasets were gathered and evaluated manually to critically judge the reliability of the training data.

3.16 Research Timeline

The research timeline outlines a comprehensive plan for a study, spanning from the initial identification of the research area to the final write-up and submission. Figure 29 showcases the approach to the research project, allowing for a systematic progression through each critical stage of the study. The process begins with the Identification of Research Area, which took approximately six months. This phase was followed by an extensive Literature Review, which took 19 months, indicating a thorough examination of existing scholarship. Another feature of this research is the

engagement with the public and professionals for input, which commenced of 6 months of the subsequent year, showcasing a commitment to inclusive research practices. The development of the Research Question is set to take place between 22nd and 27th month, signifying a thoughtful and considered approach to formulating the study's central inquiries. Moving forward, 3rd year marks the presentation of the Research Proposal. Simultaneously, Hardware Acquisition and Setup was scheduled, underscoring the importance of infrastructure in the research process. An interruption due to the COVID-19 pandemic is evident, with the entry 'COVID' from March 2020 to December 2021, indicating the period of disruption and adaptation. Meanwhile various tools were learnt required for the research, emphasizing a dedicated phase for acquiring essential skills and tools for the research endeavour. Subsequently, Model Training & Experimentation, was scheduled the following year, highlights the meticulous and time-intensive process of refining the research model. The timeline concludes with compilation of Results & performing analysis from 73rd and it took 4 months. Final thesis writeup & submission phase, spanning around one year, ended in July 2023.

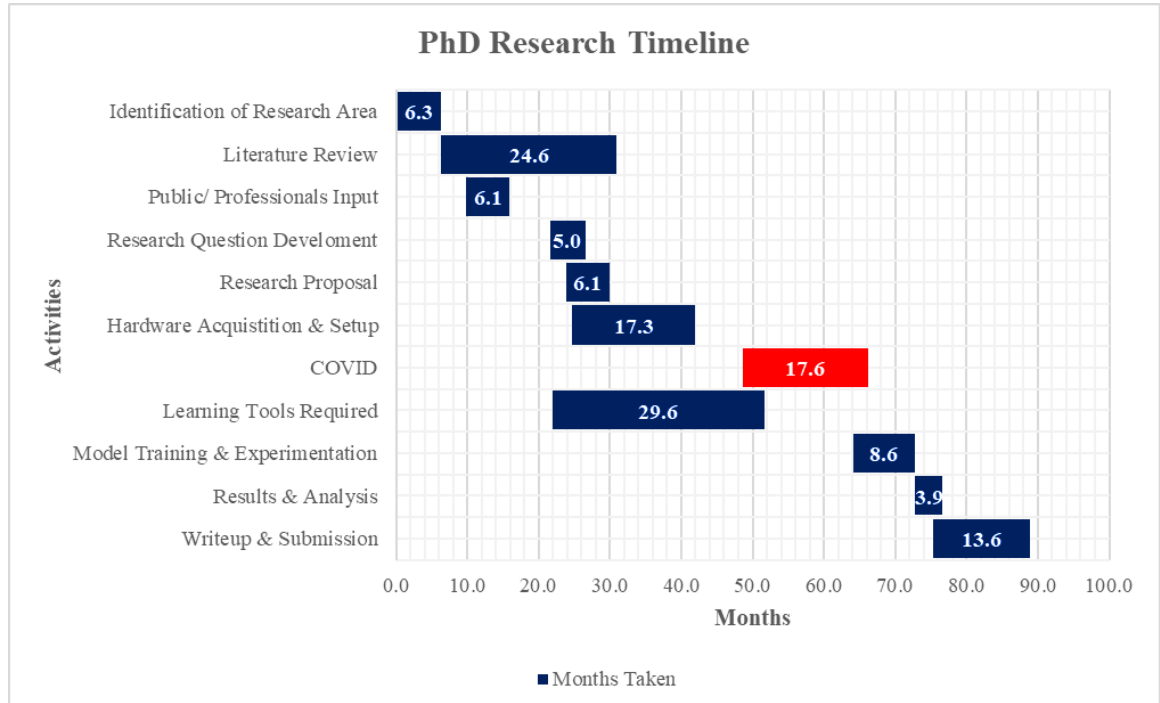


Figure 29 Ten Phases of the Research Project Spanning 98 Months

3.17 Resources and Budget

The research required a comprehensive set of resources and budgetary allocations. A significant portion of the budget was allocated towards hardware acquisition and setup, including the procurement of low-powered embedded platforms including Raspberry Pi 3 and Raspberry Pi 4, various types of cameras like IP Camera (wired and wireless), Cannon Digital Single Lens Reflex (DSLR) camera, and high-performance computers with GPUs for parallel processing. These computers include professional laptops like Alienware, Acer Nitro 5, desktop machines with RTX 2060 Super, and GTX 1660 GPUs. This infrastructure formed the backbone of the proposed system, enabling the processing and analysis of visual data feeds at various stages. This substantial funding for computing systems was dedicated to the development and implementation of AI models, which were pivotal in achieving real-time face detection and classification. These encompassed expenses related to software development tools, deep learning models, and the necessary licenses of MATLAB. Additionally, an allocation for an optical fibre network was imperative to ensure seamless data transmission and processing for the research prototyping.

Furthermore, human resources played a vital role in this research venture. Skilled personnel technicians proficient in networking were employed. Considering the ambitious scope and potential impact of the research, provisions for contingencies and unforeseen expenses were also factored into the budget. This included allowances for additional hardware or software requirements that arose during the project. Finally, travel expenses for conferences, workshops, and potential collaboration initiatives were considered, as they served as valuable platforms for knowledge exchange and dissemination of research findings. All the expenses discussed above were managed by the researcher and supervisor. The physical space for installing the setup was provided by the University of the Punjab, Lahore, Pakistan.

3.18 Limitations of Methodology

The methodology employed in this research develops advanced deep learning models for precise classification of gender, ethnicity, and facial hair styles. Integration of GAN-based models enhances the capability of the system in terms of image

enhancement. However, there are limitations of the methodology due to constraints such as time and cost, potentially affecting the comprehensiveness of the research. Firstly, biases in the training data used to create these models may be introduced by depending too heavily on pre-trained deep learning models, such as VGG face, GoogLeNet, and AlexNet for gender and ethnicity classification and GoogLeNet and ResNet 50 for facial hair type classification. Furthermore, because the methodology separates ethnicity and gender, it may not be able to handle intersectionality well. People who identify as belonging to intersecting ethnic or cultural groups or who identify with multiple gender identities may be misclassified as a result of this. Moreover, the dynamic nature of facial features and expressions may not be adequately captured by a classification system that relies solely on static images. The methodology might not be able to sufficiently account for changes in appearance over time, such as aging, variations in facial hair, or changes in makeup, which could result in incorrect classifications or decreased accuracy. Furthermore, the quality and degree of detail in the source images may impose restrictions on the use of MS CGAN for low to high-resolution image generation and sketch-to-colour image conversion. The system's overall efficiency may be impacted if the generated images don't always display a high degree of consistency or reality. Lastly, variables like lighting, image quality, and changes in posture or expression may have an impact on the methodology's performance. These factors might add more sources of variability and have an impact on how accurate the classification results are.

3.19 Summary

This PhD focuses on the development of a distributed system for the detection and tracking of persons of interest, based on a network of face recognition systems. This project aims to contribute towards the overarching objectives of addressing the threat of terrorism through technology. Accordingly, this project aims to contribute to the development of high-performance security systems designed around the context and needs of Pakistan. Significant development has been seen in this area in recent years. Efficient surveillance systems are being established for this purpose. Third-world countries like Pakistan do not have automated surveillance systems. The Safe City contains cameras capable of providing distant views only. This research will provide

a system capable of detecting and tracking high-value targets from networks of cameras. Parallel computing techniques are utilized to deal with a large amount of data. Contribution to new knowledge includes a system that can acquire images from different nodes of the network and a cascaded classification system. Deep learning algorithms are to develop models for detection and recognition. It proposes an efficient technique to detect human faces from the CCTV image frames and categorize the faces into gender, ethnicity, and facial hair types. Solutions to problems like pose variance, low resolution, and sketch images are also addressed in this research. A central database is created to store cropped images and their features. Indigenous classifiers are used to create an efficient face recognition system. The results of various models are compared and evaluated with the help of real-time scenario inputs. The system can locate desired targets from a network of cameras. It can be implemented on currently installed surveillance systems and help track targets automatically. It will increase efficiency by searching targets accurately in less time and keeping the privacy of the civilians.

Chapter 4

System Design

4.01 Overview

This chapter covers the overall system design, addressing each processing stage of the system. The complete system is made up of multiple parallel computing nodes as shown as High-level architectural diagram of the proposed system is given in Figure 30.

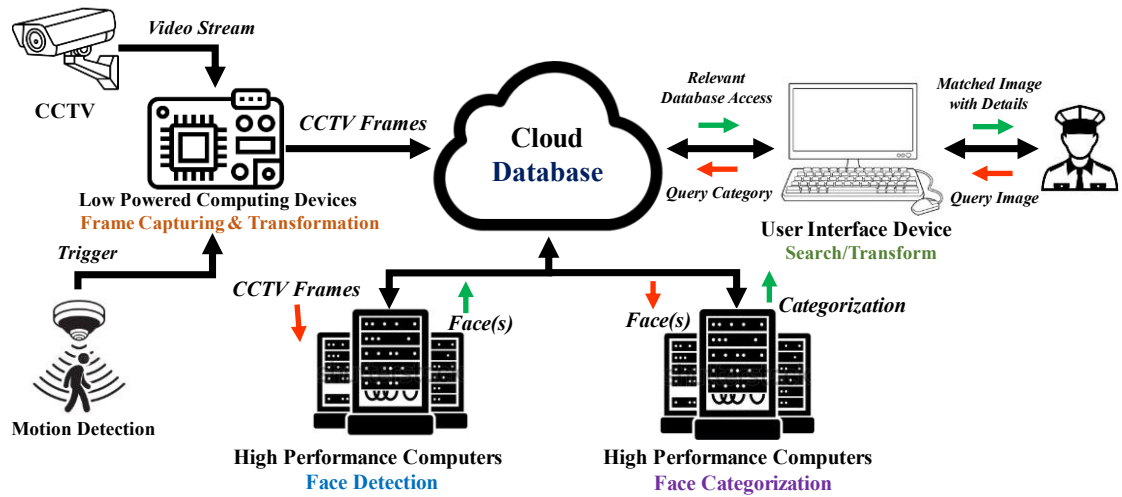


Figure 30 High Level Architectural Diagram

First stage refers to the image acquisition and basic pre-processing. Second computing stage is for face detection and third computing stage is dedicated for the face classification. Whereas the target search system is responsible for searching the target from the relevant category. Software Architecture Diagram showing the computations at each level of parallel computing stage is given in Figure 31.

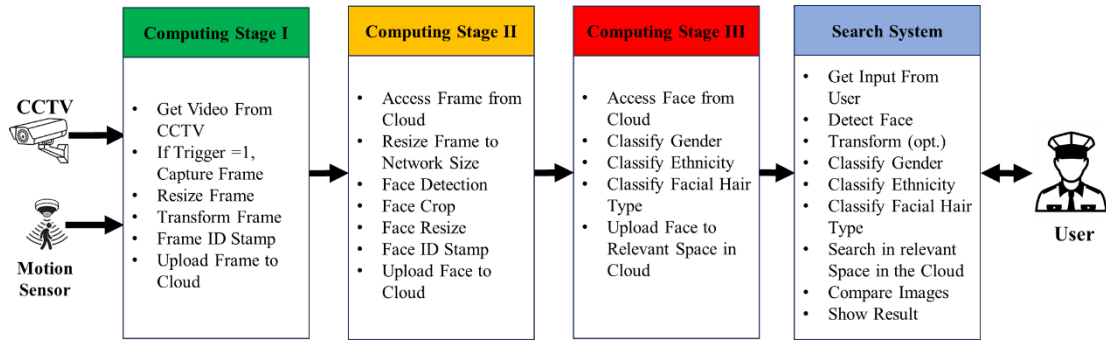


Figure 31 Software Architecture Diagram

While explaining the first computing stage, the selection criteria of the imaging devices and the important parameters of the CCTV cameras are discussed. Significance of video quality parameters of CCTV cameras in face recognition are discussed in this chapter that includes frame rate, shutter rate, brightness, saturation, hue etc. Being a distributed computing system, this research requires multiple cost-effective low powered computing devices. Selection criteria for initial computing is also overviewed in this chapter. Size of the frame to be processed is an important parameter that must be considered, that is discussed in 4.02-IV. Increasing the image size may lead to slower computation whereas decreasing the size can decrease the detection and recognition efficiency. Frame size calculations are made in the chapter to establish a relationship for resizing the input image to attain optimized results. It includes the resolution of the camera and pixels per foot requirement for efficient recognition system. Significance of grayscale conversion is discussed later in the chapter. There can be thousands of the CCTV cameras in the network creating big data connected through cloud. Memory requirement for storing this data and bandwidth requirement for transmission of the data are essential parameters. In this chapter, both the important parameters are evaluated for the system. Afterwards, addition of Passive Infrared (PIR) sensors in parallel with CCTV cameras as shown in Figure 30, is discussed that how addition of a hardware can decreases the computational cost. Then the nomenclature for CCTV frames to be stored to the database is defined in this section. This concludes the first computing stage. Second computing stage is dedicated for face detection using high performance computers. All the area under the surveillance is divided into zones to distribute the computing tasks. Specifications for distributed computing system for stage two and three are explained. GPUs are used for parallel computing; multiple GPUs are overviewed, and GPU rig is proposed. For

efficient detection and recognition pre-processing option is introduced, fog/smog removal technique along with histogram equalization is explained. Then face detection technique used in the research is discussed. Dataset collection and region of interests' generation are described. Afterwards this section explains the deep learning model training details. Dimensions of the input layer is an important part to considered, it is then calculated for optimized results. Frame resizing for face detection and face cropping regions are estimated. Pixel allotments for face and facial hair region are calculated while explaining details of the processing stage two. Then processing stage three is explained by defining the classification layers and dataset collection techniques. Transfer learning is used for classification stage, famous models like AlexNet, GoogLeNet, ResNet-50 and VGG-16 based VGG-Face are included in the research. Architectures, layers settings and training parameters are discussed for each CNN model. At the end this chapter the target search system that interacts with the user via GUI is described. In addition to image search system other options like sketch to colour image conversion, low resolution/blur image restoration and face alignment are also part of this artificial assistance system research. Finally, summary is presented to conclude the chapter.

4.02 Parallel Computing Stage I

The first processing stage includes the imaging devices, PIR detectors and the processing unit, to perform the necessary operations at this stage as shown in Figure 30. Each node has its own processor that gives the whole system the capability that multiple devices work together and upload the processed data to the cloud by performing multiple operations as shown in Figure 31. Details of this stage is given in this section.

I. Imaging Device(s) Selection Parameters

In all image recognition systems, the acquisition device plays a key role. To make a system adaptable and make it compatible with the existing hardware installed in various places like the Safe City Project, the test equipment should also be similar. That is why system is designed to work with existing security network installed in Lahore city thus IP cameras were used for prototyping. It should be noted that every

image acquisition device has few parameters that makes it distinguishable from the other devices. These parameters determine the quality and capabilities of the imaging device.

Non changeable lens types of cameras were used. The important parameters for selection of the camera includes resolution, shutter rate, video frame rate, range, view angle, focal length and aperture of lens, video compressions, Day/Night capability, noise reduction, connectivity, operating conditions, and power consumptions.

While designing the system many high-resolution cameras were on the list, but the resolution required was to be like the resolution of the already installed cameras. No doubt that high resolution gives better results of the recognition process but consumes more bandwidth while transmission and generate frames of larger size. This causes more computational cost making the process slow. Shutter rate describes the hardware capability to get an image and exposure time. Slow shutter rate results in leaving trails and making image blur. Cameras with lower shutter rates cannot capture fast moving objects like running human. These images are taken and displayed in a sequence to create a video. Video frame rate is a software control of utilizing the images per second. In this criterion the processing capability of the Complementary Metal-Oxide Semiconductor (CMOS) of the camera is involved. All the modern security cameras give variable frame rate so the user can set according to the requirement. Video with higher frame rate needs more memory for storage and consumes more bandwidth. That is why lower frame rate is used to increase the video backup time without increasing the storage memory and of course compromising the video quality.

Another parameter that is usually ignored by the domestic users, network of cameras for security and monitoring applications, is range of the camera. British Security Industry describes selection resolution should take care of the camera's resolution power to Detect, Observe, Recognize, and Identify. For each of the above, specific pixels per foot (PPF) are required. For detection, it is necessary to distinguish between humans and other objects. It requires at least 10 PPF for a general overview. Detection level gives details about the characteristics of the target like clothing, activities etc. of a person. Successful detection requires 20 PPF. Recognition is the level for determining if the same individual was shown in images taken by the same device. It needs a minimum value of 40 PPF for effective recognition. Then finally

comes the authentic recognition known as Identification having no doubt in the result that needs 60-80 PPF. Pixel density is found by dividing the pixels in horizontal axis by the Field of View also termed as FOV as given in

Eqn. 4-1. For an 8 Mega Pixel camera having 3840 horizontal pixels covering 100 ft can give 38 PPF. Giving pretty much satisfactory results while recognition [241]

$$PPF = \frac{\text{Number of Horizontal Pixels}}{\text{Horizontal FOV}}$$

Eqn. 4-1

View angle has a close relationship with the Detection, Observation, Recognition and Identification (DORI). According to the **Theia Technologies** there is trade between the field of view and the resolution. Pixel density is reduced when the FOV is increased. It is clearly shown in the Figure 32 that how result enhances when angle of the view is reduced [242].

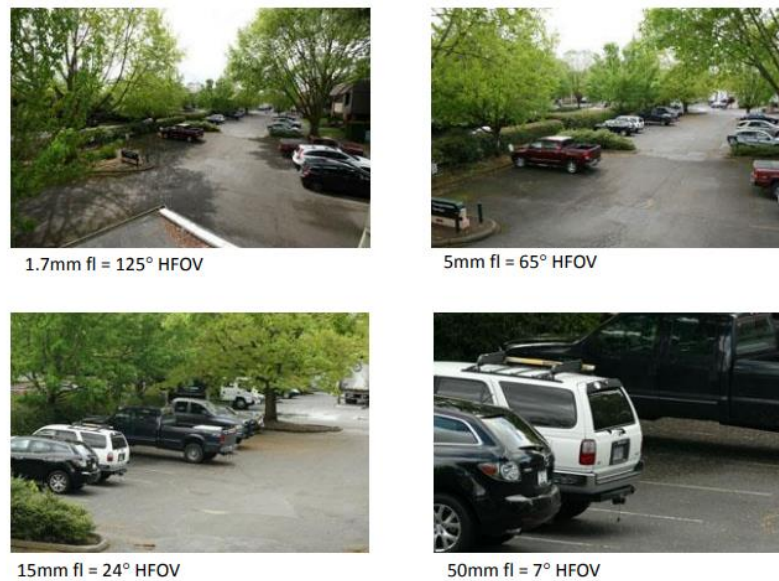


Figure 32 Resolution vs FOV [Source: Theia Technologies, 2018] [242]

Cameras having lens with narrow field of view are preferred in Identification applications like we see at immigration counters at airports and detection application in passenger lounges. Multiple cameras are used instead of one fisheye camera to enhance the detection rate. It is therefore selected carefully in this research to get better results in recognition and covering all area. In the safe city project, multiple cameras are used at specific location and keeping lesser field of view. Height of the camera

mount is kept in such a way to keep it out of reach of the public and giving minimum hypotenuse value.



Figure 33 CCTV Cameras PSCA Lahore

As shown in the Figure 33, two cameras from the front side covering 120 degrees where it can be easily done by one camera. Using lens for lesser field of view of around 60 degrees gives more resolution resulting in better surveillance results. Figure 34 shows the details of the face image enhances with the increase of resolution and relationship between pixels per foot with the forensic detail of the image. Image with 60 PPF shows much more detailed features to identify a person from a CCTV footage.

<i>Resolution level</i>	<i>Pix/ft</i>	<i>Pix/m</i>
Identification (high detail)	60	200
		
<i>Resolution level</i>	<i>Pix/ft</i>	<i>Pix/m</i>
Detection (general security)	20	70
		

<i>Resolution level</i>	<i>Pix/ft</i>	<i>Pix/m</i>
Recognition (forensic detail)	40	130
		
<i>Resolution level</i>	<i>Pix/ft</i>	<i>Pix/m</i>
Overview (motion, tracking)	10	30
		

Figure 34 Resolution vs FOV [Source: Theia Technologies 2018] [242]

Another parameter is about the lens. All the lenses cannot be used in the megapixel cameras as high performance lens is required to project image on the sensor. Lens projects a spot on the sensor array, this spot must be smaller or equal to the size of the pixel of sensor. If does not happen, then the image produced will be blurred [242]. According to section 1.3 of the Imaging Resource Guide, Angular field of view (AFOV) can be found using Eqn. 4-2.

$$AFOV [degrees] = 2 \times \tan^{-1} \left(\frac{h}{2f} \right) \quad \text{Eqn. 4-2}$$

Where h is horizontal dimension size of the sensor, f is focal length of the lens and $AFOV$ is angular field of view. Both quantities are taken in millimetres. We are generally concerned with horizontal field of view as shown in the figure below. Horizontal field of view or just field of view can be found as given in Eqn. 4-3.

$$Horizontal\ FOV [mm] = 2 \times WD \times \tan \left(\frac{AFOV [degrees]}{2} \right) \quad \text{Eqn. 4-3}$$

Where FOV is horizontal field of view in millimetres and working distance (WD) in millimetres.

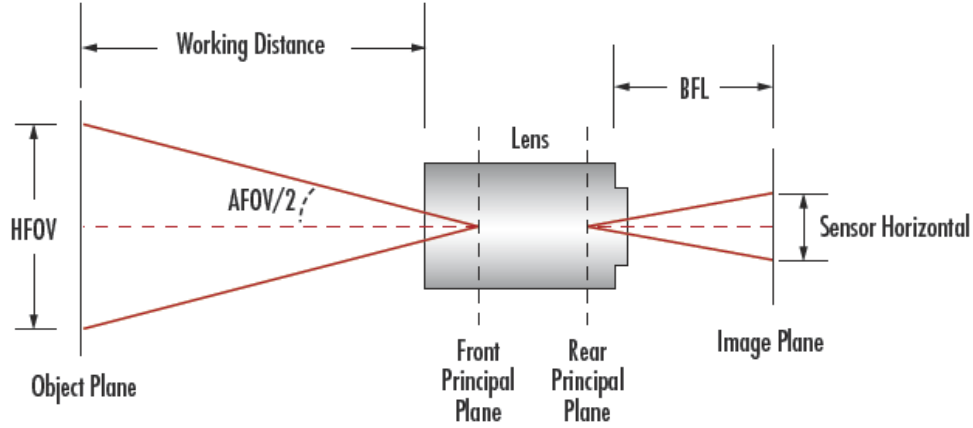


Figure 35 HFOV, Sensor Size and WD for AFOV [Source: Edmund Optics, n.d.] [243]

For calculating the focal length while selection of camera, Eqn. 4-4 can be used.

$$f = \frac{h \times WD}{\text{Horizontal FOV}}$$

Eqn. 4-4

Figure 35 shows the arrangement of all discussed parameters in pictorial form. All these calculations were used for the system design and experimental setup [243]. As discussed earlier, more the smaller the pixel size more the resolution, keeping area same. It's very important to know about the lens capability and the pixel sizing of the sensor for designing a machine vision system specially for the biometrics. To support a high-performance lens an efficient sensor with small size of pixel is required. Otherwise, image will be blurred. Resolution of the digital sensor is measured as using Eqn. 4-5.

$$\text{Resolution} = \frac{1}{2 \times \text{Pixel Size}}$$

Eqn. 4-5

Pixel is multiplied by 2 because 1 cycle is equal to 2 pixels and 1 pixel is 1 sample. Resolution is also termed as number of Nyquist cycles per millimetre. It is maximum number of cycles a digital sensor can sample at. As the sensor is digital and lens gives the analogue signal. There is a sampling rate with which the signal is sampled by cycles per millimetre. Analog signal is in the form of cycles per millimetre or number of

trough and crests per millimetre in the wave or it can be termed as frequency. Sensor samples the wave at Nyquist cycles per frequency.

If a lens produces 100 cycles analogue signal that is to be sampled at 100 cycles, it means that it will miss the middle data, so if 50 cycle analogue signal is sampled at 100 cycles it will give considerably well results [244]. Another part that is particularly important for selection of an imaging device for machine vision is video compressions. As discussed, earlier video is series of frames joined together. For latest surveillance systems H.264 codec is used. It is most efficient codec as it evaluates the group of frames and removes the duplicate frames (no change) to save space. MJPEG (Motion JPEG, Motion Joint Photographic Experts Group) is known as motion JPEG, it evaluates each frame and sends separate JPEG images [245]. For a machine processing an individual frame is to be dealt so JPEG is used, and the frame repetition issue is overcome by the PIR sensor. Otherwise, two consecutive frames had to be compared in the system to finding out the faces.

II. Video Quality Settings

Various parameters are to be considered before selection of CCTV camera and designing a surveillance system, like operating setup, light condition, and placement of camera to get best results. Various times of the day have different system requirements for outdoor monitoring systems depending on the amount of illumination, both natural and artificial. Auto lens cameras are used in the outdoor applications to cope with light variances. Sometimes images from the cameras are not much clear, as the lighting conditions keeps on changing with time and the reflection from the moving bodies [246]. In addition to the lighting another parameter that is important specially for the moving targets is shutter rate, more the shutter rate better the detail of the features of a face. The camera shutter speed on the left-hand image is 1/30s, which is probably typical for all installed cameras with the default camera settings. The shutter speed on the right-hand image was purposefully reduced to 1/125s [247]. Appropriate shutter rate must be set so that detection and recognition can be performed effectively. Figure 36 shows the difference between the 1/30s (on the left side) vs 1/125s (on the right side) shutter rates, and it's evident that the left image is blurred as compared to the image on the right.



Figure 36 Shutter Rate Comparison: 1/30s vs 1/125s [Source: Simon L, 2016] [247]

Modern cameras give access to the settings that includes the frame rate, video format, brightness level, hue, contrast, and saturation of the image quality. Usually, every modern CCTV camera provides user an option that the basic parameters of the video can be altered. Figure 37 shows the settings pane of one of the CCTV cameras used in the research.

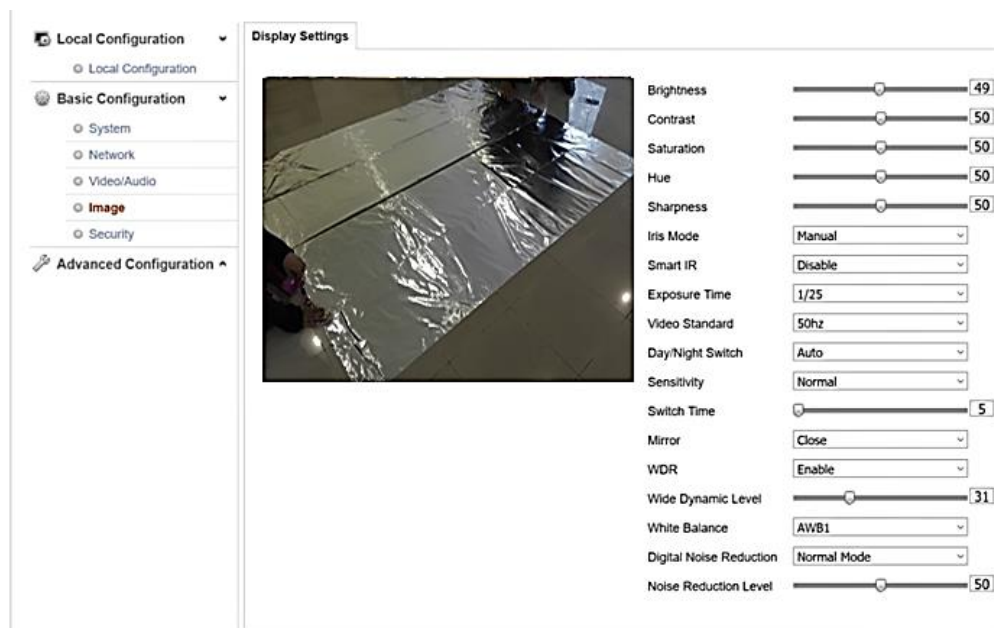


Figure 37 IP Camera Image Settings Panel

All the parameters can also be changed in the image by using mathematical operations on the captured images. But doing it using Raspberry Pi will consume lot more time in transforming the image into a revised image. For changing Hue Saturation Value (HSV) of an image requires transforming Red, Green and Blue (RGB) image to HSV and must convert back to RGB after changing the HSV matrix.

To deal with this issue saturation, hue, and value (brightness) of the image are read by the computing device. All the three values are stored in the separate matrix. Hue is ignored. Where mean of brightness and saturation are found. After the series of experiments on the CCTV footages threshold limits for saturation and brightness. The minimum threshold for the average saturation was set to be 0.25 and maximum was 0.45. If the level is found deviated from this value, then the system changes the saturation to nearest limit.

Brightness can be found with the same method by calculating the mean value of the brightness (Value) matrix form HSV matrix. Brightness can also be increased or decreased by the help of histogram shifting but we can save the computation cycles by leaving the task to automated feature of the cameras. Limits for the mean brightness of HSV is set to be between 0.3 and 0.9. These values are extracted by the experiments performed on the 100 CCTV images. RGB to HSV conversion of 3840 x 2160 image using Intel Core i7 9750H @ 2.60 GHz requires 0.06 seconds whereas back conversion takes 0.6 seconds. The brightness and saturation level adjustments require 0.06seconds for each change.

III. Raspberry Pi as Parallel Computing Stage I

As discussed earlier that the system is designed to cope with the challenge of dealing with thousands of cameras at time. These cameras are distributed in the various geographic locations of the city and the command centre is situated at some other part of the city. For paralleling processing, a low cost but efficient computing device was required to help the main system by distributing its workload. Cost was the major issue that was to be taken care of. Now the other task that was to be done was to capture the frames from the cameras and upload to the main server. It was to be decided what tasks can be done at the nodes. It was to be determined by knowing the processing capability of the computing device. The computing device at the front node must have to be compact in size and high processing speed. It is attached to the network switch to access IP camera and server. PIR sensor attached at General Purpose Input/Output (GPIO) pins is used to detect any change in the frame to save the computation cost. In our system this stage is used for image acquisition and preliminary image processing.

While selecting the device the one more parameter to be considered was the cost. In the literature section, various single board computing devices are discussed including Raspberry Pis, NVIDIA Jetson, HiKey970 and Intel Neural Stick. The selection of Raspberry Pi 4 as the preferred platform for the task of frame capturing and colour to grayscale conversion was based on a careful consideration of various factors. Firstly, Raspberry Pi 4 offers a cost-effective solution without compromising on performance. Given that cost was a significant constraint, Raspberry Pi 4 provides a budget-friendly option compared to alternatives like NVIDIA Jetson and HiKey970, which tend to be pricier. Additionally, for the specific tasks of frame capturing and colour to grayscale conversion, Raspberry Pi 4's processing capabilities were more than adequate, meeting the project requirements efficiently. Moreover, the Raspberry Pi ecosystem is well-supported with a large community and a wide range of resources, making it easier to find solutions and troubleshoot any potential issues that may arise during the implementation. Overall, considering the balance between cost-effectiveness and performance for the defined tasks, Raspberry Pi 4 emerged as the optimal choice for the project.

IV. Frame Dimensions Optimization

Resolution of the cameras varies with the variation of the user requirements. Size of each frame increases with the increase of the camera resolution. Better resolution gives better results but on the other hand high resolution images requires more computational power. To create a real-time detection and recognition system, each pixel of the image matters. For such system, frame size should be optimized in such a way that minimum size can be calculated without effecting the efficiency. For resizing the frames, multiple parameters like Field of View, Horizontal Range, Vertical Range, Focal Length etc. are to be taken care of. For the calculations, CCTV camera installations in the city network had to be analysed. Cameras with different heights and angles are installed. For the reference, one scenario is taken as an example, whereas cameras with varied parameters can be evaluated with the established formulae in this section.

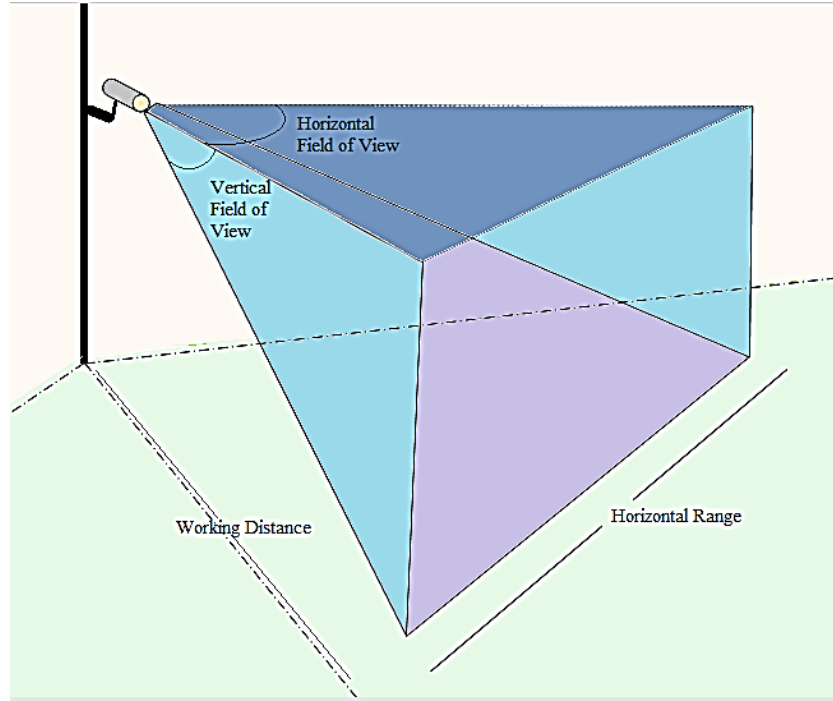


Figure 38 Camera Installation with Field of View

Figure 38 shows the setup of the camera installation. For the calculations to find the image quality at distances it is required to know the vertical and horizontal field of view, consider the working distance of the camera and height of the camera installed. For a recognition, the target must be present in the coloured pyramid showing the camera field of view. In this example, camera is installed at a height of 13 feet like the safe city cameras. There is a blind spot under the camera due to the angle of depression. To cover the range of 50 feet, angle of depression of the camera was found in such a way that camera should give view 4.25 feet above the ground that is the average shoulder height of an adult in Pakistan [248].

8 MP camera was used for the experiments, and it has a horizontal field view of 50 degree using 6mm lens.

Using the Eqn. 4-6, $A = \pi r^2$

$$\text{Horizontal Range} = 2 \times WD \times \tan\left(\frac{HFOV [\text{degrees}]}{2}\right)$$

Eqn. 4-6

$$\text{Horizontal Range} = 2 \times 50 \times \tan\left(\frac{50}{2}\right)$$

Eqn. 4-7

$$\text{Horizontal Range} = 46.63$$

Eqn. 4-8

In Eqn. 4-8, Horizontal range was found to be $46.63 \approx 46.5$ feet at working distance of 50 feet. Ranges and angles are shown in Figure 39.

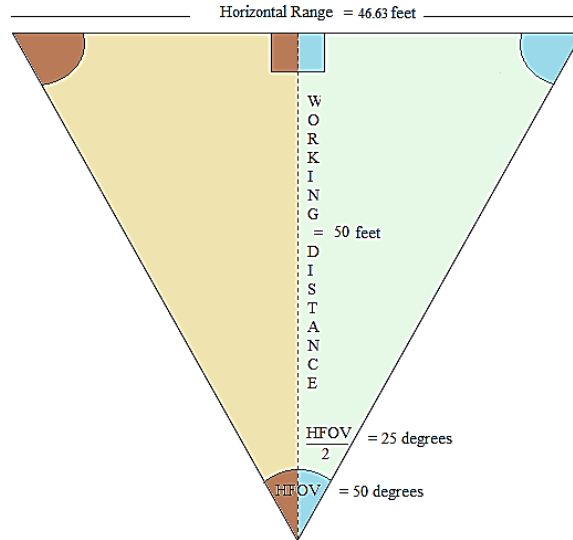


Figure 39 Plan view of HFOV

This system is only interested in the images with humans in it. There are various cameras covering the same place. If person of interest is missed by one camera while crossing the camera sideways, then the frontal image must be captured by the other camera. So, the face re-alignment is not performed for all the targets. Manual face alignment feature is available in the system and is discussed in the later stages of this Chapter. By the calculations it was found that 46.5 feet are represented by 3840 pixels that means one foot is represented by 82.55 at the maximum distance.

Horizontal angular field of view of the camera is known to 50 degrees. Vertical field of view of camera can be found by using the ratio of the horizontal pixels and vertical pixels as in Eqn. 4-9.

Vertical Angular FOV

$$= \frac{\text{number of vertical pixels}}{\text{number of horizontal pixels}} \times \text{Horizontal Angular FOV}$$

Eqn. 4-9

$$\text{Vertical Angular FOV} = \frac{2160}{3840} \times 50$$

Eqn. 4-10

$$Vertical\ Angular\ FOV = 28.125$$

Eqn. 4-11

Using the Eqn. 4-9 vertical angular field of view is found to be 28.125 degrees in Eqn. 4-11. For finding the minimum distance Now the task was to find the blind spot or the minimum distance to detect a face and the maximum distance. As per the arrangements and calculation shown in the previous section the working distance or range of the camera was set to be 50 feet. Both quantities were set to detect the face of a person with minimum height of 5 feet at distance of 50 feet. Arrangement of the side view is shown in Figure 40 and plan view in Figure 41. In the diagram the average shoulder height of a person C is kept being 4.25 feet to detect face of 5 feet tall person. As stated in earlier section, average face height B is 0.75 feet. Now if we look at the upper most triangle the perpendicular of the triangle A was 13feet – 5 feet i.e., 8 feet. It is the blind region at WD. Base of the triangle WD is equal to the working distance i.e., 50 feet. Using the trigonometric formula in Eqn. 4-12, the angle A was found to be 9.09 degrees as in Eqn. 4-14.

$$\angle A = \tan^{-1} \frac{Blind\ Height\ (A)}{Working\ Distance\ (WD)}$$

Eqn. 4-12

$$\angle A = \tan^{-1} \frac{8}{50}$$

Eqn. 4-13

$$\angle A = 9.09$$

Eqn. 4-14

Angle of depression was found to be 23.2 degrees in Eqn. 4-17 using Eqn. 4-15.

$$Angle\ of\ Depression\ (\angle D) = \angle A + \frac{\angle AVFOV}{2}$$

Eqn. 4-15

$$Angle\ of\ Depression\ (\angle D) = 9.09 + \frac{28.13}{2}$$

Eqn. 4-16

$$Angle\ of\ Depression\ (\angle D) = 23.2$$

Eqn. 4-17

To find the value of Z that is the minimum distance on which camera can capture a face, in Eqn. 4-20 angle C was calculated to be 38.68 degrees by Eqn. 4-18.

$$\angle C = 90 - (\angle A + \angle AVFOV)$$

Eqn. 4-18

$$\angle C = 90 - (23.2 + 28.13)$$

Eqn. 4-19

$$\angle C = 38.68$$

Eqn. 4-20

Angle D found out to be 51.32 degrees in Eqn. 4-23.

$$\angle D = 180 - 90 - \angle C$$

Eqn. 4-21

$$\angle D = 180 - 90 - 38.68$$

Eqn. 4-22

$$\angle D = 51.32$$

Eqn. 4-23

Solving triangle for finding X through Eqn. 4-24 equals to 10.4 feet as given in Eqn. 4-26.

$$X = \frac{H}{\tan \angle D}$$

Eqn. 4-24

$$X = \frac{13}{1.25}$$

Eqn. 4-25

$$X = 10.4$$

Eqn. 4-26

Similarly, Y found to be 3.6 feet in Eqn. 4-29 using Eqn. 4-27.

$$Y = \frac{C}{\tan \angle D}$$

Eqn. 4-27

$$Y = \frac{4.5}{1.25}$$

Eqn. 4-28

$$Y = 3.6$$

Eqn. 4-29

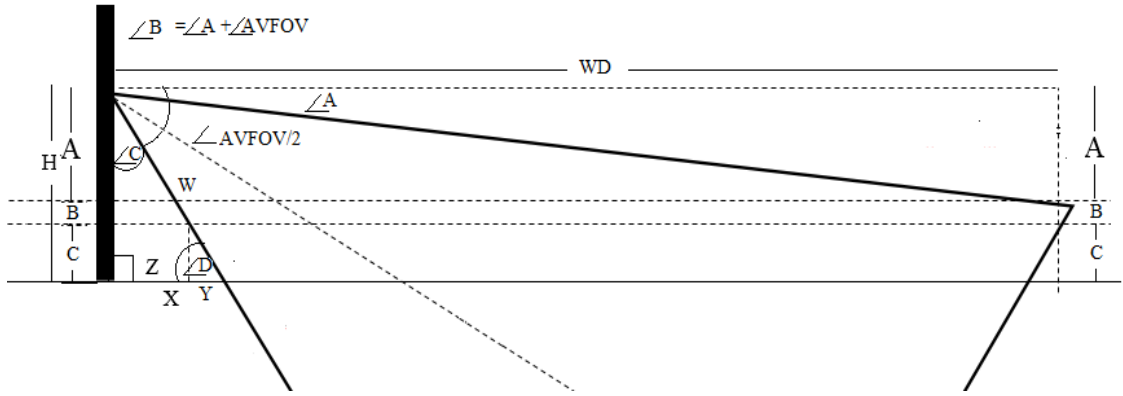


Figure 40 Field of View Calculation

Finally, using Eqn. 4-30, the minimum distance covered by the camera Z is found to be 6.8 as given in Eqn. 4-32.

$$Z = X - Y$$

Eqn. 4-30

$$Z = 10.4 - 3.6$$

Eqn. 4-31

$$Z = 6.8$$

Eqn. 4-32

Giving a margin of 0.2 feet and was treated as 7 feet. Now the shortest distance on the horizontal plane needed to be calculated to estimate the number of frames per seconds settings. In Eqn. 4-32, the value of Z has already been found as 6.8 feet. D is the minimum distance that can be travelled in one direction of the camera. $\angle AHFOV$ is already known to be 50 degrees.

$$\angle A = \frac{180 - \angle AHFOV}{2}$$

Eqn. 4-33

$$\angle A = \frac{180 - 50}{2}$$

Eqn. 4-34

$$\angle A = 65$$

Eqn. 4-35

Using Eqn. 4-33, $\angle A$ is calculated to be 65 degrees.

$$D = 2 \times \frac{Z}{\tan \angle A}$$

Eqn. 4-36

$$D = 2 \times \frac{6.8}{2.14}$$

Eqn. 4-37

$$D = 6.35$$

Eqn. 4-38

Finally, the minimum distance was calculated by Eqn. 4-36 to be approx. 6.5 feet.

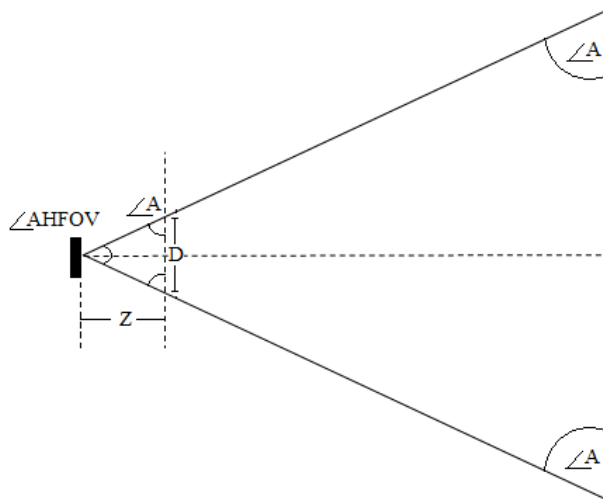


Figure 41 HFOV for Horizontal Distance calculation

All the above calculations can be verified using **IP Video System Design Tool**. Camera parameters were set in the software as Figure 42, 8MP camera with 6mm lens and 1/2.5” sensor.

Field of View & Lens Focal Length

Camera

Installation Height (ft)
13

Manufacturer

Model

Sensor Format
1/2.5 "

Resolution
3840x2160 (8MP 16:9)

Focal Length (mm)
6

Figure 42 Camera Configuration

Target maximum and minimum height were set to 5 feet and 4.5, respectively. Maximum range (distance from camera) was set to be 50 feet as shown in Figure 43. The field of view was found to be 46.5 feet as proven mathematically.

Field Of View

Distance from Camera (ft)
50

Installation Height (ft)
13

Target Height (ft)
5

FOV Width (ft)
45.7

Viewing Angles, °
50 29.4

4.5 ☒ 5 ☒ 6 ☒

Figure 43 Target and View Setting

The horizontal distance calculations are verified by the software.

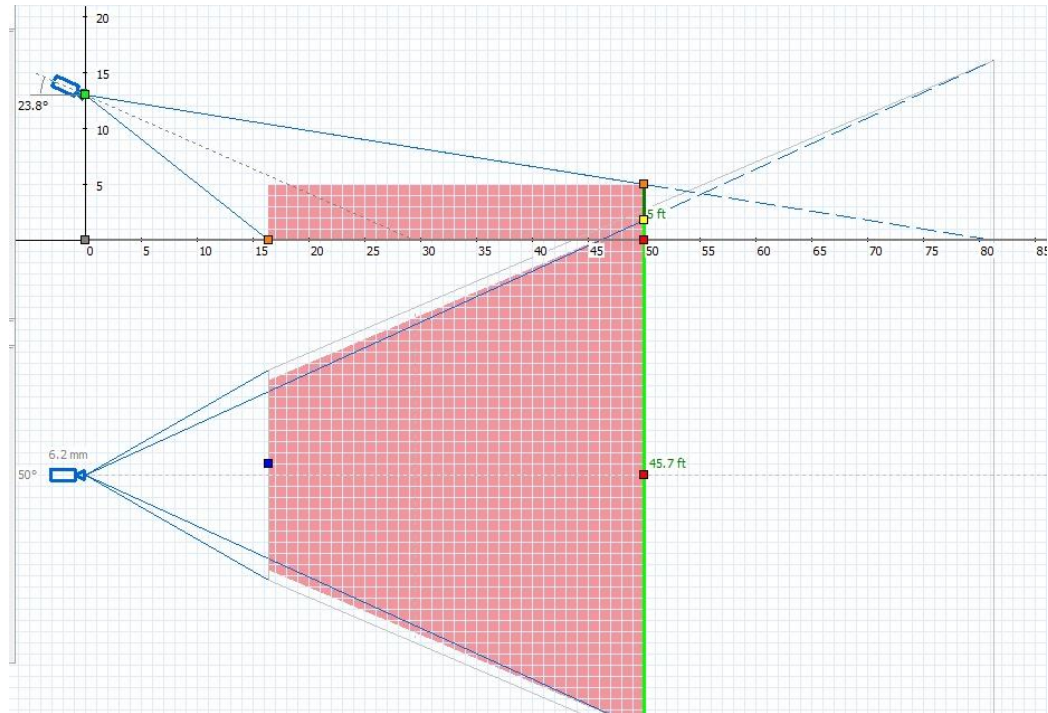


Figure 44 Camera Views

The results of PPF are shown in Figure 45 and matches with the calculations shown above.



Figure 45 Expected image PPF

By repeating all the calculations for 3.6mm lens with 104 degrees FOV, with same working distance the horizontal distance will be increased. It means more area represented by same number of pixels or decreasing pixel per foot. PPF is changed with the same ratio as of the lens focal length as given in Eqn. 4-39.

$$\frac{Focal\ Length_a}{Focal\ Length_b} = \frac{PPF_a}{PPF_b}$$

Eqn. 4-39

For 3.6mm lens PPF for same working distance is calculated using Eqn. 4-39.

$$\frac{3.6mm}{6mm} = \frac{82\ ppf}{PPF_{3.6mm}}$$

Eqn. 4-40

$$PPF_{3.6mm} = \frac{3.6mm}{6mm} \times 82.5\ ppf$$

Eqn. 4-41

$$PPF_{3.6mm} = 49.5\ ppf$$

Eqn. 4-42

For the identification purposes in challenging conditions the pixel per foot requirement increases. It will be requiring more pixel density.

The original resolution from the camera was 3840 x 2160. For the horizontal view 3840 pixels are covering maximum of approximately 47 feet at the maximum working distance of 50 feet. It is providing resolution of 82 pixels per foot. As shown in the Figure 43 and Figure 44 a person of maximum height of 5 feet can be detected at maximum range. Total vertical distance covered by the camera can be calculated from the same figure. It can be found using Eqn. 4-43.

$$Vertical\ Distance = WD - X + Y$$

Eqn. 4-43

$$Vertical\ Distance = 50 - 10.4 + 3.6$$

Eqn. 4-44

$$Vertical\ Distance = 43.2$$

Eqn. 4-45

Total vertical distance covered is calculated to be 43 feet and is covered by 2160 pixels as given in Eqn. 4-45. For the recognition criteria set by the Theia Technologies [242]. 40 pixels are required to represent a foot for recognition and 60 pixels are

required for identification. We have 60 pixels to represent a foot. For the increasing the efficiency of the system 15 pixels per foot margin was found. Image was rescaled in such a way to get the 60 PPF.

$$\text{Scaled Horizontal Pixels} = \frac{\text{Original Horizontal Pixels} \times \text{Required PPF}}{\text{Original PPF}} \quad \text{Eqn. 4-46}$$

$$\text{Scaled Horizontal Pixels} = \frac{3840 \times 60}{82.5} \quad \text{Eqn. 4-47}$$

$$\text{Scaled Horizontal Pixels} = 2793 \quad \text{Eqn. 4-48}$$

From Eqn. 4-46, scaled horizontal pixels found to be 2793. For vertical pixels Eqn. 4-49 was used.

$$\text{Scaled Vertical Pixels} = \frac{\text{Scaled Horizontal Pixels} \times \text{Original Vertical Pixels}}{\text{Original Horizontal Pixels}} \quad \text{Eqn. 4-49}$$

$$\text{Scaled Vertical Pixels} = \frac{2793 \times 2160}{3840} \quad \text{Eqn. 4-50}$$

$$\text{Scaled Vertical Pixels} = 1571 \quad \text{Eqn. 4-51}$$

Vertical pixels should be scaled to 1571 pixels. Finally, it is calculated that the final image for face detection can be reduced to 2792 x 1571 without affecting the result with 60 PPF at the maximum range.

For the original resolution contained 8,294,400 whereas the reduced image consists of 4,386,232 pixels. Resolution reduction will lead to save the system memory, computations, and data bandwidth for uploading the images to cloud. Size of the file is now reduced to 52.88% of the original resolution.

It must be noted here that this reduction will downgrade the system capability in the identification. DORI rule says that for the Identification purposes we need 60 PPF. This system will be able to attain 60 PPF at working distance of 50 feet. For the indoor use, this range is enough to identify targets at entry and exits points.

Above calculations are made for specific resolution, resizing for other resolutions can be done using the same formulae. These equations can help to calculate the image size to be sent for efficient recognition with changing the variables like height, working distance, HFOV, angle of depression, resolution of camera etc. Variables can be changed for the cameras with different resolution. Reason for making all the calculations is to make an adaptive system that can alter the image size at first computation stage in such a way that detection and recognition can be performed with minimum size of the frame correctly and the all the nodes of the cameras behave in similar way.

V. Grayscale Conversion

Face detection and recognition does not involve colour or skin tone. Image in grayscale can serve equally good as of colour image. Converting the image to grayscale can reduce the size of matrix by three times. To resize an image of 3840 x 2160 to 53% of its original size on Intel(R) Core (TM) i7-9750H CPU @ 2.60GHz it takes *0.08 seconds*. This was a time-consuming task for the system. As discussed in the literature that the face detection algorithms do not require coloured image. The latter stages of the system also do not need any criteria based on the skin colour for the recognition of a target. A coloured image is made by combining three matrices of the same dimensions. One dimension represents levels of red, second represents levels of green and third represents levels of blue. Whereas a grayscale image contains a single matrix of same dimension representing 256 levels of grey from 0 to 255.

Converting the same resolution image into grayscale using the same machine, it takes approximately *0.004 seconds*. After converting the image to grayscale, when the image was resized, the time consumed was found to be *0.02 seconds*. It is proved to be 4 times faster than that of the coloured image. Now the total consumed time is *0.024 seconds*. Still *3.33 times* faster and saving valuable time for performing other tasks. Implementing the same code to Raspberry Pi with Quad core Cortex-A72 (ARM v8) takes around 1.6 of the time taken by Intel(R) Core (TM) i7-9750H CPU.

VI. Memory Allocation

Raspberry Pi writes the image to the cloud in JPEG format that occupies more space than the frame size of H.264 video frame. Grayscale frame in JPEG format having dimensions of 2792 x 1571 occupies around 480KB whereas 3840 x 2160 in grayscale is around 741 KB large. Frame rate is set to two frames per second so, 2 x 480 KB data is to be written by each Raspberry Pi every second. Details are given in later sections.

In a frame of 8 MP camera with 60 PPF maximum number of detected faces can reach up to 6,000 recognizable faces. But the average number of faces in a CCTV footage in normal conditions at peak hours are not more than 200 from each camera.

In the main computing system faces are to be detected in the frames. Overall head breadth is referred to the distance between extreme horizontal points of the head. The maximum horizontal distance between the left and right facial boundaries is referred to bizygomatic breadth. Whereas bi-gonial width of face is little less than bizygomatic breadth. In the study conducted by Joseph W Young in 1993, results shown in Table 5, the average bizygomatic breadth for females is 131.81 mm or 5.18 inches and for males its 140.7 mm or 5.5 inches. Whereas the average head breadth for females is 145.8 mm or 5.7 inches and males have average head width of 152.4 or 6.0 inches [249].

Table 5 Both Genders Mean Bizygomatic & Head Breadths [Source: J. W. Young 1993] [249]

Bizygomatic Breadth	Females		Males	
	mm	inch	mm	inch
Mean	131.8	5.2	140.7	5.5
Standard Deviation	5.1	0.2	6.0	0.2
Min	119.1	4.7	126.0	4.96
Max	148.1	5.8	152.9	6.0
Head Breadth	Females		Males	
	mm	inch	mm	inch
Mean	145.8	5.7	152.4	6.0
Standard Deviation	5.6	0.2	5.3	0.2
Min	128.0	5.0	139.0	5.5
Max	161.0	6.3	166.1	6.5

Facial features are present in the bizygomatic breadth, from the above data presented the females have lesser width than males. 6 inches wide face must be represented by

40 pixels [250]. This pixel density is true for the worst scenario. Whereas for the recognition layer the input size is set to 224 x 224. That is why face image can be resized to 224 x 224. There are three types of recognition layers are present in the system, explained in Categorization System section. Two matrices are required for these two layers, face matrix is used for gender and face type whereas lower matrix is used for facial hair classification. Details are given in Face Cropping section.

As proved in the Overall dimension of the stored face is 224 x 365 pixels. Details regarding the face size and cropping calculation are given in Face Resizing Calculation section. For storing a jpeg image of 224 x 365 image consumes around 16 KB of size on disk.

VII. Networking

As discussed earlier, computer networking is important part of the system. This subsystem is responsible for uploading the images to the cloud. Each node may contain 1 to 8 cameras and each camera sends the video stream in the H.264 format to the processing unit. As discussed in the previous section the frame rate is high to track the target in a reasonable manner. For this purpose, high speed networking device was required. As it is being used in the actual safe city project. Huawei has completed the project and laid down dedicated optical fibre to provide high speed video streaming for surveillance centre. This is totally secured telecommunication system based on LTE-A technology. It has a capability to integrate the other sources too. Moreover, a redundant wireless path has also been deployed [251].

For a selecting a networking device few things are to be considered. There are generally two types of communications used for connecting the nodes. Wireless communications and wired communications. In wired communication ethernet and optical fibre can be used. Wireless communication option has already been rejected due to the speed. 5G is not yet implemented in Pakistan so current system cannot support large amount of data that has to be sent to the command centre. Bandwidth, data rate and data security are the three major concerns for the system.

Restricting to 4G-LTE Advanced, as it is widely available in Pakistan and been implemented by PSCA as a backup. It is convenient to implement but it cannot match

the speed of the wired connections. Maximum bandwidth available for an LTE-advanced system is 5 to 20 MHz [252]. Table 6 shows the data rate comparison in theory and the speed practically available in UK.

Table 6 Data Speed of Wireless Theoretical vs Practical [253]

Network Type	Theoretical		Practical	
	Upload Speed (Mbps)	Download Speed (Mbps)	Upload Speed (Mbps)	Download Speed (Mbps)
3G	2	7.2	0.4	3
3G HSPA+	22	42	3	6
4G LTE	50	150	10	20
4G LTE-Advanced	150	300	25	42

For a wired communication, the optical fibre has few advantages over the ethernet. With the increase of the use of network devices ethernet has been upgraded to Gigabit Ethernet that can provide speed up to 1000 Mbps. Cat-6 can support 10Gbps of transfer rate. Optical fibres use light instead of Electromagnetic (EM) waves. For transmitting the data to longer distances optical communication is preferred. Data rate is many times faster than that of the ethernet in the case of multimode fibre cables. Data rate reaches up to 100 tera bits per second. Ethernet is exposed to electromagnetic interference and strong fields generated around the transformers may affect the transmission. On the other hand, optical fibre uses light for communication and total internal reflection phenomenon. Privacy and security are other reasons why optical fibre is preferred in the surveillance applications. Ethernet can easily be intercepted where optical signals are difficult to intercept [254]. The other networking switch proposed for final system is Cisco® Catalyst® 2960-X series switch. 48-port Gigabit Managed SFP Switch. It supports wide range of fibre optics. It has 600 MHz dual core CPU i.e., APM86392. Total 128 MB of flash memory. It has total 512MB of RAM. It has Static IPv4 routing with 4K active VLANs capacity. It has 108 Gbps of forwarding bandwidth and 216 Gbps of switching bandwidth. It gives maximum capacity of 1023 active VLANs [255].

VIII. Bandwidth Estimation

As multiple cameras are being used in parallel a good bandwidth is required. In addition to number of cameras installed, the other parameters include resolution, frame rate and video compression codec used. For calculating the bandwidth required for transmission of the video Eqn. 4-52 is used:

$$\text{Bandwidth (Mbps)} = \text{Bitrate (Main)} \times N + \text{Bitrate (sub)} \times M$$

Eqn. 4-52

Where N is number of mainstream cameras and M is number of sub stream cameras. Bitrate depends upon the resolution and frame rate of the video [256]. As discussed in previous section more the frame rates more images are to be sent and compression is another parameter to decide the frame to be sent. Multimode optical fibre has bandwidth above 1000 MHz using the cameras having 2 MP and 8 MP resolution with the frame rate 20 fps. Both are used as mainstream cameras. System needs to send the detected faces over the cloud in addition to the live video to command centre. Using the chart for bit rate developed by Hikvision [257]. (See Annexure B) 8 MP camera with 20 fps requires average of 5468 bps and maximum of 12160 bps. Where 2 MP camera with the same frame rate need 1856 bps on average and maximum of 3072 bps. Maximum bit rate for both comes to be 17,408 bps and average of 7960 bps.

In a camera video stream, Raspberry Pi reduces the frame rate to 2 frames per second to give time to all the other algorithm to work. Trigger is also controlled by the PIR sensor to avoid repetition of the same face with the same angle using hardware instead of consuming machine cycles of Raspberry Pi for frame comparison. Recognition system available at the main processing node does not require colour images as no algorithm need skin colour detection. In this regard, greyscale images are sent to the cloud. Grayscale images from each node at the rate of 2 frames per second are sent to cloud. Frame size of the camera is 184 KB at H.264-10 high quality but for image storage JPG format is used. Frame size dimensions reduction is performed and set the frame size to 2792 x 1571 that is 53% of the original image size. Raspberry Pi writes the image to the cloud in JPEG format that occupies more space. Grayscale frame in JPEG format having dimensions of 2792 x 1571 occupies 480 KB whereas 3840 x 2160 in grayscale is around 515 KB large. 2 x 480 KB data is to be written by each Raspberry Pi every second. Bitrate will be 7680 kbits per second. Bandwidth required

will be 7.68 Mb/s per camera in addition to the bandwidth requirement of video stream from camera.

IX. PIR Sensor for Frame Triggering

Designing a system that can process a huge amount of data from 12,000 cameras was a big task. These cameras are distributed in city of Lahore having area of 404 sq.-km. Lahore is further divided into towns. It is located at a strategic point north and west by Sheikhpura district and shares boarder with India in the east. Where another important city, Kasur is in the south of Lahore [258]. According to census held in 2017, it has a population of 19,398,081 people [259]. Huge amount of data is to be sent to command centre for live video streams and frames for storage/recognition purposes. In video with H.264 compression sends the optimized updated frame with tracking the region where there is a difference from the previous frame. This codec helps to reduce the bit rate but with a huge computational cost at both encoding and decoding ends [260]. There was an important parameter of frames selection and frame rate that needed to be taken care of before going for the detection and recognition stage. Cameras are generally set to provide 20-30 frames per second. But this is an extremely high rate for dealing with the real-time recognition process. Frame rate for the system is kept 2 frame per second as humans are slow moving targets and field of view is wide and targets traveling towards the camera are to be observed only. Processing every single frame from the video consumes plenty of machine cycles. Multiple frames with the same images will result in detecting same face with the same angle or no human in the frame. This can consume processing cycles of the system without any additional benefit. This issue had to be resolved. Solution for reduction in the computing operations is made with the help of hardware-controlled triggering mechanism for frame difference assessment. As per the frame rate selection, rate of 2 fps was decided to make the system efficient in terms of compute complexity. PIR with single output is connected to Raspberry Pi's GPIO to detect any change in infrared signature of the camera view. It helps to keep Raspberry Pi in idle state if there is no change. It also avoids re-processing of similar frames multiple times. Module used provides a 5V pulse for detection.

PIR sensor generates a pulse when there is a change in the infrared signature. It rejects the smaller changes in the pattern due to the environmental changes and due to the intruding small animals. As shown in Figure 46, the algorithm works in such a way that after completing the initial setup, it keeps on pooling for the trigger at GPIO pins connected to PIR sensor output and if it finds the required pulse it goes inside the loop. Inside the loop it captures the frame from the corresponding camera, from which PIR trigger is received, then it converts the image to grayscale. Then the image is downsized and uploaded to relevant location on the cloud.

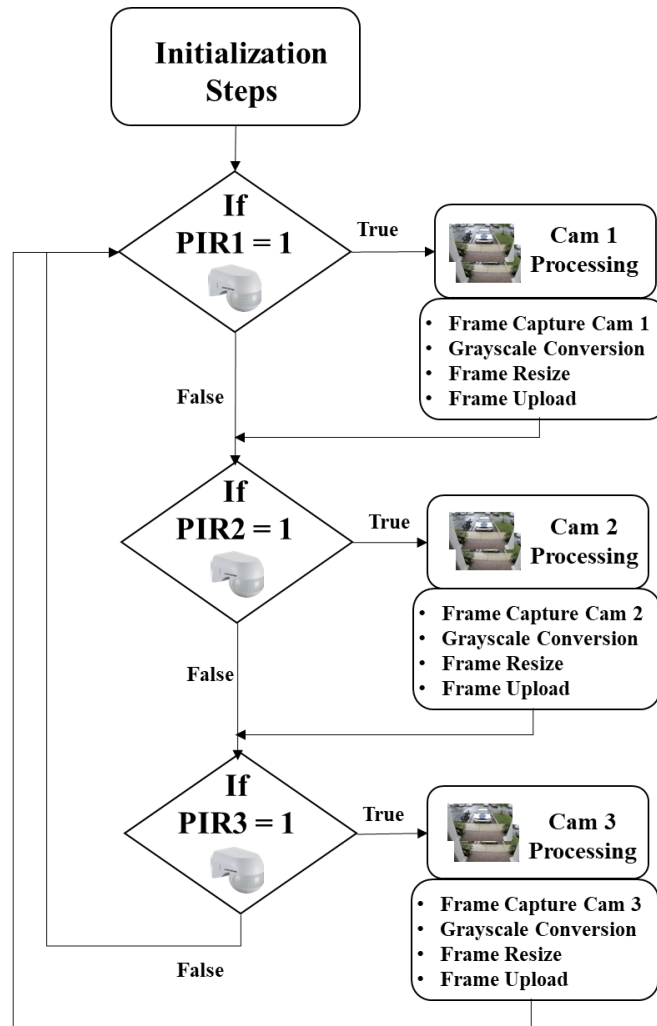


Figure 46 PIR Triggered Raspberry Pi Computations

As shown in Figure 46, this system one Raspberry Pi can handle maximum of three cameras at 2 frames per second. Number of cameras can be increased if the resolution

or frame rate is decreased. The generated pulse is kept less than one second, as two frames are to be captured per trigger. The prototyping unit has 2-second-long delay pulse for output, but the proposed design should have delay pulse less than one second. If the pulse is kept longer, it will trigger the next cycle, resulting in capturing additional frames.

X. Nomenclature for Frame Images

After dealing with all the procedures the last step was to write the images to the destination folder. From that location it will be accessed by the high-performance computing devices for further processing that is to detect faces from the image and then classification into categories. For the convenience, these images had to be saved in the JPEG format for easy retrieval and processing. There can be multiple cameras working in parallel and can communicate to the other computing devices through local processing unit that is Raspberry Pi. The files that must be uploaded should have unique name for easy understanding and avoid overwriting. There was another issue that was addressed here is to retrieve the time, date and location of the image taken and saved from the file name. It was solved by issuing unique name to all the frames. Nomenclature is defined in such a way that name contains the camera ID of the camera, date and time and frame number. Frame rate is set to be 2 that's why the time is taken till seconds and the last digit shows whether its first image or the second image of each second. To deal with all the city, system has prototyped the city model in a specific way. For prototyping the city of Lahore is divided into 60 zones. Each zone contains various locations at which camera poles are installed. Same pole site contains multiple cameras covering specific field of views (50 degrees used in the calculations).

Camera ID is defined with the help of nine characters, first three characters represents zone, next four numbers represent camera location and last two numbers for the camera number of that location. For example, in camera ID GBR023101, GBR is representing the zone, 0231 is the location number and 01 is the camera number at location 0231.

Next part of the frame name is date and time of the frame captured in DDMMYYHHMMSS format. There is last digit that shows the frame number of that second. Right now, frame rate is set to be two, that is why it can be one or two only.

Final frame name will look like GBR0231012201221312451, it means that this image is uploaded from camera ID GBR023101 at 13:12:45 on 22-01-2022 at it is first frame of the second.

XI. Frame Uploading/Writing

Final task for the subsystem at acquisition node was to write the frame to a location from which it can be accessed for face detection. There is a sperate folder at the cloud location for each camera. Images are sorted in hourly manner for each day of the year. Details of the folder management is given in the Face Categorization section.

4.03 Parallel Computing Stage II

The second stage of computing is dedicated for face detection from the frames those are uploaded by the multiple imaging nodes connected through cloud as shown in Figure 30. There can be hundreds of imaging nodes in the network and must be divided into zones to split the work. In addition to face detection this stage pre-processes the images if required as displayed in Figure 31. As discussed in the previous section the output of the CCTV cameras is adjustable and usually does not require many transformations. But being an artificial assistance system, it gives the user option to transform the images to achieve better results of recognition. Details of this stage is given in this section.

I. Zone Division

Till the previous section, frames captured are ready for face detection. For further tasks it is not possible for a computing system to deal with 10000 cameras at the rate of 2 frames per second. To cope with this issue, this research proposes multiple computing devices dealing with the cameras parallely. For designing and calculations of computing machines, geographical and population distribution of the city must be observed as displayed in Figure 47.

LAHORE POPULATION DENSITY

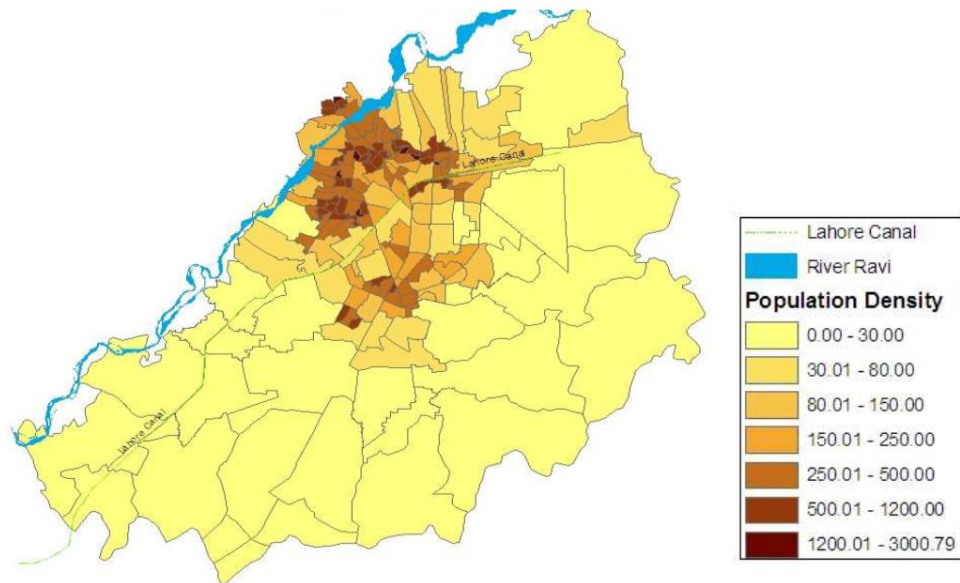


Figure 47 Population Density Map of Lahore [Source: Khadija M, 2019] [261]

Just like any other big city of the world, city of Lahore also has different population density. More population is concentrated in a specific part of the city. Majority of the business and commercial activities are done in northwest sectors. Industrial zones are situated outside the city and away from the residential zones. Majority of the public offices are also situated in the central region of the city [261]. With the more density the more surveillance is required, and more cameras are being installed in the region. With the increase of the cameras more of the back-end machinery is occupied for dealing with these regions.

Zones are made on the basics of number of cameras installed. This division is made in such a way that one zone will consist of 200 camera nodes. Total 10000 cameras can be dealt with only 500 zones. One system needs to oversee the 30 frames per second to fetch faces out of the frame. This is a very heavy computational task in terms of time. It cannot be managed by the conventional CPU. At this spot, the need for the parallel computing devices jumps in. It requires extra ordinary system to work with in parallel to perform time complex tasks like face detection, face categorization and face recognition. CPU is slower than GPUs for performing this type of tasks.

This division can be changed as per the requirements and demographic distribution in the camera network. This research gives a platform only that can be adjusted as per the needs.

II. Computing System Design Parameters

This computing stage is important in terms of compute capability. As discussed in earlier sections that complex high-speed computations are needed.

The first thing that had to be decided before designing the computing system was that what type of computations are expected from the device. Deep learning algorithms were required to run on the systems. Input rate was high and for this regard parallel computing was required with the help of GPUs. CPU is in charge for assigning the tasks to the GPUs. That is why a capable CPU that can manage all the GPUs was required. While selecting a CPU there are few parameters to be taken care of. First thing is the number of cores. Number of cores describes the parallel computing capability of the CPU.

(A) Parallel Computing via GPU

Keeping the results by [262] and [263] in view, GPU is proposed for parallelizing of the tasks. Above researchers used NVIDIA GTX 470 with CUDA toolkit 4.0 and NVIDIA GTX 770M. Face detection is found to be multiple times faster on GPU as compared to CPU. Tasks must be segmented before assigning to the GPU. Few tasks are to be dealt by CPU where all the repetitive tasks. Detection system was implemented by [264] on Intel Core i5 4210 U CPU @1.70 GHz and GPU of Nvidia GeForce 820. System had 4GB RAM. GeForce 820 has 2 GB memory and 2 multiprocessors with maximum of 1024 threads per block and 1536 active threads per multiprocessor. Both CPU and GPU based versions of Viola Jones algorithm were used. Figure 48 shows the summary of CPU vs GPU Comparison performed by Bhutekar et. al. 2014 [262].

Time (in <u>ms</u>)	Image Size					
	480 * 360		650 * 400		700 * 580	
	GPU	CPU	GPU	CPU	GPU	CPU
Face Detection	102	780	160	900	350	1406
Face Recognition	90	207	127	279	312	300

Figure 48 CPU vs GPU Comparison [Source: Bhutekar et. al. 2014] [262]

Developer has evaluated face detection for six image sizes. For detecting face from a colour image of 2500 x 1667 resolution, CPU took *1595.02 ms* whereas GPU completed the same task in *312.66 ms* making the GPU 5 times faster than the CPU. Whereas this time was *1227.40 ms* and *82.16 ms* for CPU and GPU respectively for 1280 x 626 resolution image. Proving the GPU 15 times faster [264].

Now the next task was to selection of the GPU. First step is to compare the GPUs on which experiments have been performed for face detection with the available newer versions. Initially one GPU per system was to be evaluated. The main parameters for selection of GPU includes, graphic processor used in the card, its architecture, clock speed of the processor including base clock and boost clock. The other commonly used term and known to the users is memory. For dealing with the memory, its size, type, memory bus size and bandwidth is to be considered. At last, but most important is the number of the CUDA cores. Already been discussed in the literature. It defines the capacity of the GPU to manage the parallel processes. As the multiple cores addition in the CPUs brought a revolution in the field of computing, similarly GPU's CUDA cores provides more power for the dealing with the parallel processes. CUDA libraries, drivers and runtime are managed by CPU. As shown in the Table 7, the GPUs used by the researchers in the mentioned research are far less capable than the NVIDIA GeForce GTX 970 and NVIDIA GeForce RTX 2060. The major thing here to be noted is the difference in the number of cores. System was designed using the massive GPUs to deal with the huge amount of data in real time.

Table 7 NVIDIA GPUs Comparison [Source: TECHPOWERUP 2019] [265]

Graphic Card	NVIDIA GEFORCE 820M	NVIDIA GTX 470	NVIDIA GTX 770M	NVIDIA GeForce GTX 970	NVIDIA GeForce RTX 2060
CUDA Cores	96	448	960	1664	1920
CUDA	2.1	2.0	3.0	5.2	7.5
Architecture	Fermi 2.0	Fermi	Kepler	Maxwell 2.0	Turing
Base Clock	625 MHz	608 MHz	706 MHz	1050 MHz	1050 MHz
Memory Size	1 GB	1 GB	3 GB	4 GB	6 GB
Memory Type	DDR3	GDDR 5	GDDR 5	GDDR 5	GDDR 6
Memory Bus	64 bits	320 bits	192 bits	256 bits	192 bits
Bandwidth	14.04 GB/s	133.9 GB/s	96.19 GB/s	224.4 GB/s	336 GB/s
L1 Cache	64 KB	64 KB	16 KB	48 KB	64 KB
L2 Cache	128 KB	640 KB	384 KB	2 MB	3 MB

Using the NVIDIA GeForce GTX 2060 the face detection was found to be around 20 times faster than the CPU. The amount of data from the database to be processed is too much that is why one GPU cannot be sufficient for our processing. More GPUs working in parallel can solve this issue. GPU rig was used to connect multiple units with one CPU. It must be noted here better GPUs alone cannot manage the data; appropriate CPU is also required as few of the tasks in this stage is to be done by CPU as GPUs are designed for performing less complex but repetitive tasks. CPU used for the Core i7 7th Generation with eight cores. GPUs are designed specifically for graphical operations and our research includes similar operations, working with the pixels and their transformations. As we know computer gaming requires continuous image updating and transforming for the effects like shadows, flares etc. These tasks are very well dealt with the multicore GPUs. For extreme machine NVIDIA GeForce RTX 2080 or above can be used with 2944 CUDA cores with 8 GB memory.

(B) Proposed Multi-level GPU Rig

For prototyping the GPU has been extended to the Asus Nitro 5 using with GeForce RTX 2060 and Core i7 desktop with GeForce 960. Due to cost constraints only, the rig is proposed for the implementation. That is why complete design and parameters are given in the thesis. For designing a rig few things are considered. First thing is to define the number of the GPUs to be added to the machine. Then the type of the GPU. As discussed earlier the final GPU proposed for the real time system is Nvidia RTX

2060. In coming section, requirement of the total number of the GPUs are proved for the dealing with the 30 fps for face detection tasks.

This rig is proposed to build on aluminium frame. For selection of the motherboard, the first criterion was set to the number of Peripheral Component Interconnect Extended (PCI-X) slots available as we need to attach total eight GPUs with this motherboard. In this regard C.B250A-BTC PLUS V20 is proposed. It has 16 PCI Express slots. It supports DDR4 RAM. 500 watts power supply is required for powering up the motherboard and basic peripherals to power up.

Intel Core i9-9980 HK is a 9th generation CPU with base frequency of 2.4 GHz. It has option of turbo mode with 5.0 GHz. It can be used with turbo mode and heating can be taken care of using ultra cooling with circulating liquid in addition to heat sink and fan. This CPU has total 8 cores and 2 threads each for parallel working. 16 MB cache makes this CPU performance superior. It supports DDR4 memory up to 128GB. Memory is not much required as GPUs have their own memory as well. Total 3 DDR4 RAMs of 32 GB each is proposed.

The most powerful part of the rig is the GPU. As discussed in previous section Nvidia RTX 2060. It has 1920 CUDA cores and works on *Turing* architecture. It has base clock of 1050 MHz with 6 GB GDDR 6 memory. 192 bits memory bus provides 336 GB/s bandwidth that is ideal for fast data communication between layers. 64 KB per SM L1 cache and 3 MB L2 cache gives an outstanding capability of handling process ques. Each GPU consumes 160 watts of power and is recommended by the Nvidia to have 500 watts system power when used in conventional manner. Separate power supply of 2000 watts is proposed for feeding 8 GPUs with 8 pin power connectors. Each GPU generates lot of heat and temperature rises to 88 degrees Celsius. Moreover, one card is 2 slots wide. To deal with the area and heating issues, slot extension is required to keep safe distance and proper cooling using power air blowers.

All the data is written on main server, but it must be downloaded first to avoid processing delays. Data is continuously read, removed, and written by the algorithm. It needs a fast storage device with efficient seek time with durability. Intel Optane SSD 905P Series - 1.5TB can serve the purpose.

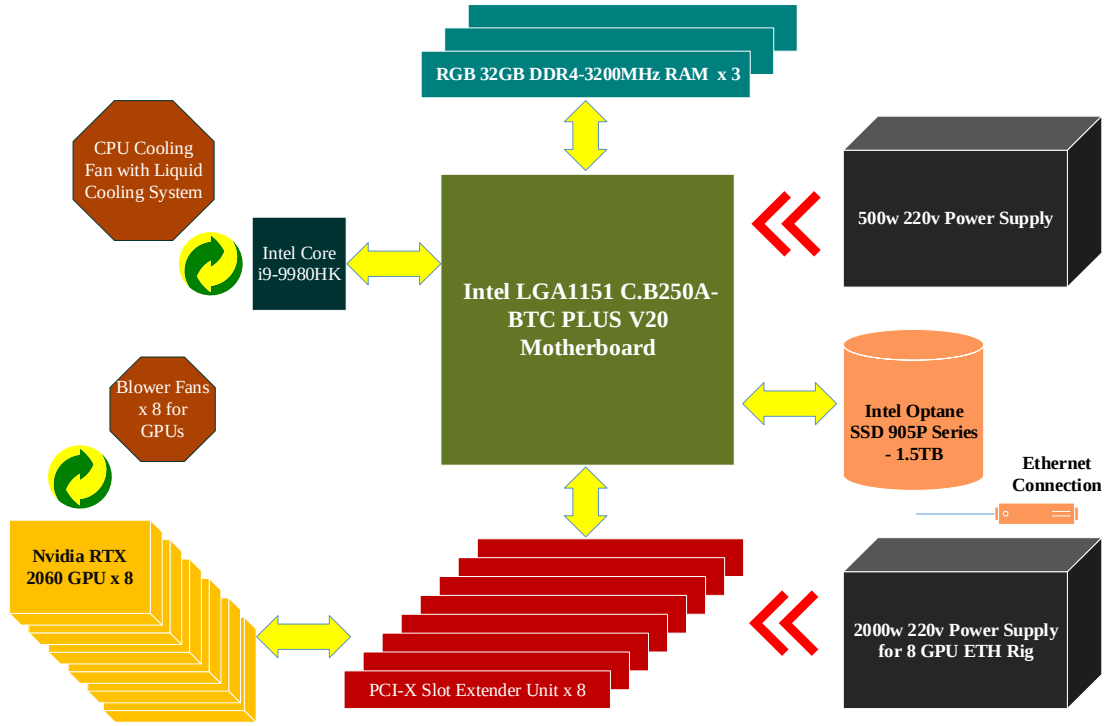


Figure 49 Rig Overview with Components

Combining all the hardware makes a powerful system that can serve one zone of the city with real time processing capability as shown in Figure 49.

III. Fog/ Smog Removal & Histogram Equalization

Lahore is in geographical area where fog persists for 3 to 4 months of a year. With a highly urban population this fog is converted to smog making AQI of the city one of the worst in the world [266]. This badly effects the visibility in the city causing road accidents. Sometimes visibility range drops to few meters [267].

This can directly affect the performance of the system. It required a layer that can remove / reduce the fog element. Level of fog and blur is evaluated with the vertical and horizontal variance in the pixel levels of the image [268]. If the image is blurred, then it must be addressed. In this regard Retinex Algorithm was used for fog removal. This algorithm was developed to remove fog for monitoring traffic. It uses Retinex Algorithm in combination with Wavelet transform [269].

$$R_l(x, y) = \log I_l(x, y) - \log[F(x, y) * I_l(x, y)] \quad l = 1, \dots, n$$

Eqn. 4-53

In Eqn. 4-53, $R_I(x, y)$ is the output image after Retinex transform, $I_I(x, y)$ is the input image, $F(x, y)$ is centre function given by Eqn. 4-54.

$$F(x, y) = Ke^{\frac{-(x^2+y^2)}{\sigma^2}}$$

Eqn. 4-54

Whereas x and y are the coordinates and σ is responsible for sharpness. In wavelet transform, vertically low pass filter is used, and horizontally high pass filter is used. Two image layers of high frequency and low frequency are generated. Low frequency is layer is suppressed to achieve the target [270].

In the [270] fog removal was proposed and proved that Retinex algorithm and wavelet transform algorithm showed better results for fog removal. This technique is adapted in the system to attain promising results.

Dark channel prior was applied to image before applying the Retinex Transform. It removes the haze from the image. It follows four steps, estimation of light in atmosphere, transmission map estimation and refinement and then reconstruction of image [271]. It deals with the low intensity pixels in at least one channel of colour.

$$I(x) = J(x)e^{-\beta d(x)} + A(1 - e^{-\beta d(x)})$$

Eqn. 4-55

In Eqn. 4-55, $I(x)$ represents the image with haze, $J(x)$ is without haze image, atmospheric light is given by A . β is the atmospheric scattering coefficient that is equal to 0 in clear conditions and d is the depth.

$$t(x) = e^{-\beta d(x)}$$

Eqn. 4-56

By keeping $t(x)$ as transmission map in Eqn. 4-56.

To estimate the $J(x, y)$ in hazy condition we use Eqn. 4-57.

$$J(x) = \frac{I(x) - A}{t(x)} + A$$

Eqn. 4-57

This transform can be applied when there are foggy conditions. This stage will be used in unusual circumstances only.

Some images under blurred conditions over heterogeneous regions are homogeneous and do not have much variation in their grey levels. Histogram of an image plots the occurrence frequency of every grey level. It ranges from 0 to 255. Here it is important

to discuss the histogram parameters. For example, the Figure 50 is manually captured by keeping the lighting conditions in the view. The histogram shown in the figure expresses that this image has enough number of each grey level providing the maximum information.

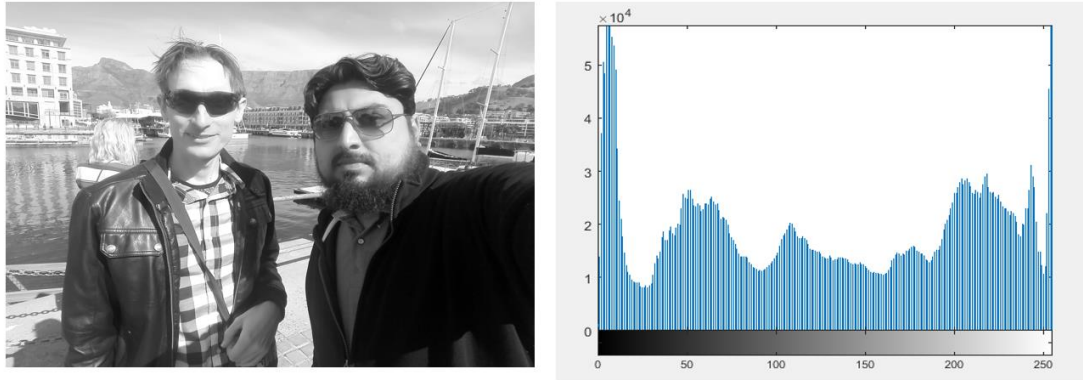


Figure 50 Histogram of A Normal Image

But it is not the case with the images taken by the live video streams because it has a general view not focusing a specific person or the object. It results in the non-uniform histograms. As shown in the Figure 51 there is absence of pixels with many levels of grey. This is evident from the image that it is not clear and does not represent much data for the computations.

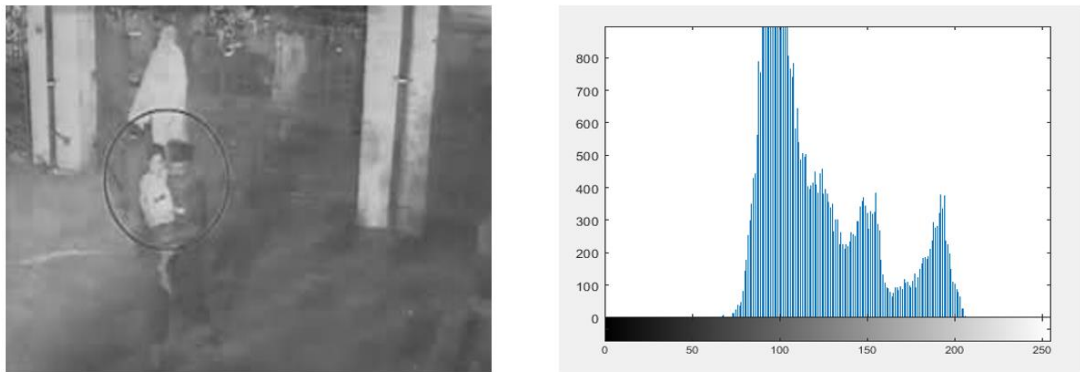


Figure 51 Compressed Histogram Image

For the solution of this problem the first thing that can be done to the image is to stretch the histogram, so it distributes the current pixels in 255 levels of grey. As shown in the Figure 52 image has become much clearer and easier to understand. Now the histogram is spread between 0 to 255.

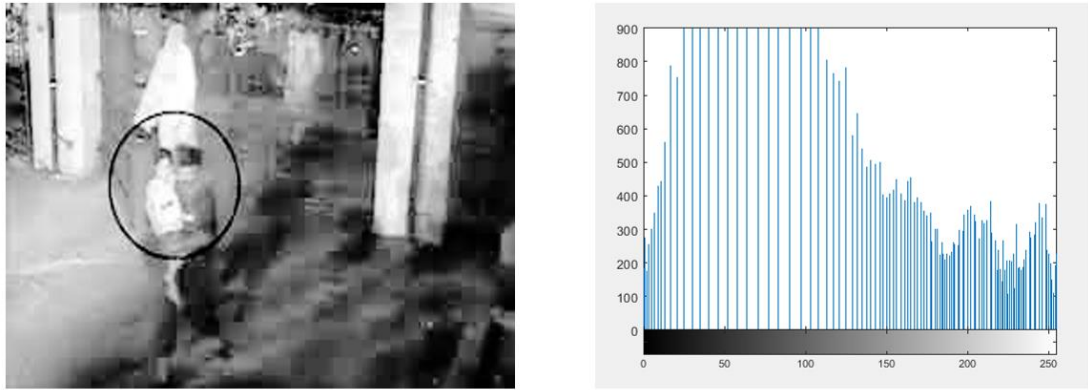


Figure 52 Histogram Equalized to 255 Levels

As shown in the above figure there are abrupt changes in the levels. Image histogram distribution can be changed so that to address the above-mentioned issue. For an image with already controlled brightness, features of the human face are represented by the middle levels of the grey of the histogram. Type of distribution was selected with series of experiments, and it was established that the Rayleigh distribution gives the best results for detection of human faces from the CCTV images. Rayleigh distribution is applied on the images to transform the histogram into bell shape. Another issue that can arise after the equalization of the histogram contrast factor. It can be termed as the high peaks of the image. By the experiments if this factor is not taken care of then it may result in bad results while detecting faces from the images. Clip limit of the image is set to 0.01 to achieve the best results after applying on the 100 test images for detection of faces using the viola jones algorithm. Experimental results are shown in following sections. After applying the Rayleigh distribution with clip limit of 0.01 the histogram becomes smooth and ready for the further processing stages.

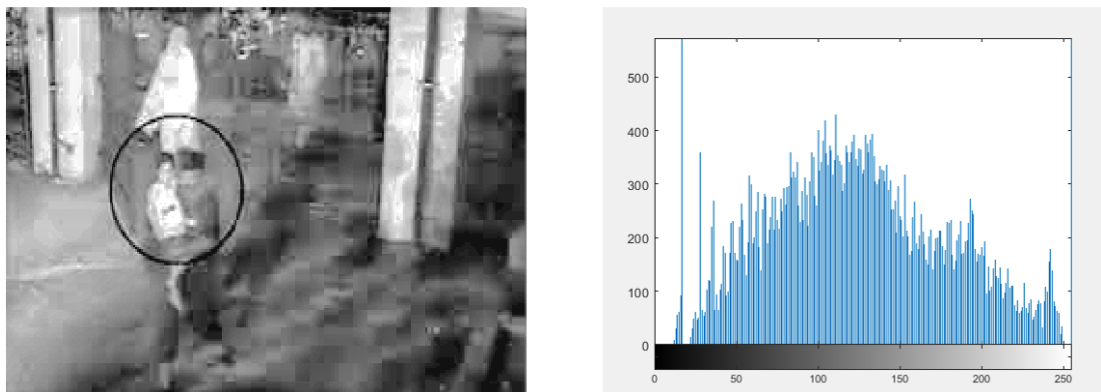


Figure 53 Corrected Histogram

As shown in the Figure 53 the final transformed histogram is an incredibly attractive shape for the face detection step. All the above tasks consume *0.03 seconds* with CPU per frame.

IV. Face Detection

Initial testing was done on the Viola-Jones object detection framework. Its working has already been discussed in literature. Algorithm has following stages shown in Figure 54:

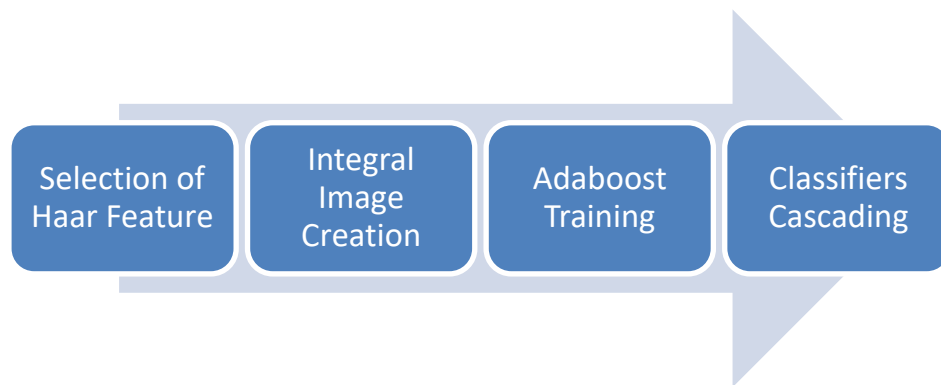


Figure 54 Face Detection Process using Machine Learning

While selecting the algorithm to detect faces out the frames, the main thing to be taken care of was the type of images we are dealing in. As stated above the first choice of every face detection system can be Viola Jones but while using the viola jones for face detection using *object detection* in MATLAB, it was found to very fast but not effectively translated dealing with the small images that a CCTV image contains. To cope with the situation another technique was required perform efficiently for CCTV images. As discussed in the literature *Deep Learning* is the solution to this problem. MATLAB being the main tool for the development of the system, has majority of the algorithms those are required for the designing system.

The other option was CNN based detectors. Since 2013 till date, we can see advancements in CNN based systems. Two major types of detectors are one stage and two stage. In two stage detectors one stage is for region of interest and the other stage is for classification. Whereas in the one stage detector, localization and classification is done at the same time. Problem is solved by performing the classification and

bounding box regression directly on feature maps making it regression problem instead of detection problem. This solution reduces detectors inference time.

In a study that was done for Face and Person Detection algorithms comparison by Belhassen et al. (2019) [171] gives an overview of frame rate and accuracies of various algorithms. Dataset used was near to our dataset as it includes WIDER, CHUCKFACE and self-gathered dataset from CCTV images. Both speed and accuracies were evaluated on the dataset using GPUs. In comparison viola jones has a limitation of detection side poses and small faces, whereas Faster RCNN can resolve the issue by detecting small faces. But the Faster RCNN is slow as compared to other methods. YOLO shows better results than Faster RCNN. YOLO v2 is faster than YOLO v3 but in terms of accuracy YOLO v3 outperforms YOLO v2 [272].

Ullah et. al. 2022 [273] presented a pioneering approach in the development of a real-time framework employing machine learning and deep learning techniques for the detection and recognition of human faces within CCTV images. Leveraging principal component analysis (PCA) and convolutional neural networks (CNN) for feature extraction, the study evaluates their performance in conjunction with K-nearest neighbour (KNN), decision tree, random forest, and CNN algorithms. With a dataset comprising over 40,000 real-time images captured under diverse conditions, including varying light levels, rotations, and scaling, the system achieves recognition with high accuracy exceeding 90% [273]. Faster-RCNN method is employed by Halawa et. al. 2019 [274] for both face detection and recognition tasks. Inception V2 architecture was used in the study. The study further optimizes crucial parameters such as learning rate and epoch settings for the Faster R-CNN model to enhance face recognition performance on CCTV footage. The research dataset comprises 6 individuals with 50 face images for each individual, which are utilized for training, testing, and validation purposes. It reached an accuracy of 89% for detection [274].

Deep Learning is faster than the machine learning. But it has variety of options while selecting the algorithm. As discussed in the literature four algorithms were evaluated and applied. Time for training and detection for our test images were found using the same dataset. RCNN, Faster RCNN and YOLOv3 were evaluated one by one. For using the Deep Learning functions in the MATLAB few toolboxes were required. It includes Deep Learning Toolbox.

(A) Dataset Collection for Face Detection

The input frame images are from the CCTV cameras and the datasets selection was the major concern as the lighting conditions, view and the resolution of the images are different from the normal photograph. Picture that has been taken manually by keeping all the participants in the frame and lighting is adjusted by placing them under best lighting condition. Keeping in view this issue, algorithms were trained with the help of self-created in combination of open-source dataset of *WIDER FACE* for human face that can be used for CCTV image faces. This huge dataset contains more than 12000 images from events like festivals, gatherings, parade etc. In these images more than 150000 faces are present. It includes around 90000 blurred tiny faces. Till this stage, the only concern was the face detection only that is why no pre-processing on the face dataset was performed. Images of the participants were taken and published with their consent. (See Annexure C and D)

More images for the dataset were gathered by locally made arrangements as shown in Figure 55 and Figure 56 for day and night respectively. Frames from the CCTV video footages acquired from the experimental setup were added. These images are taken with the consent of the participants. These participants are students at University of the Punjab. The setup is shown in the experimental setup section.



Figure 55 CCTV Face Detection Dataset (Day)



Figure 56 CCTV Face Detection Dataset (Night)

Samples of training images are shown in Figure 57.

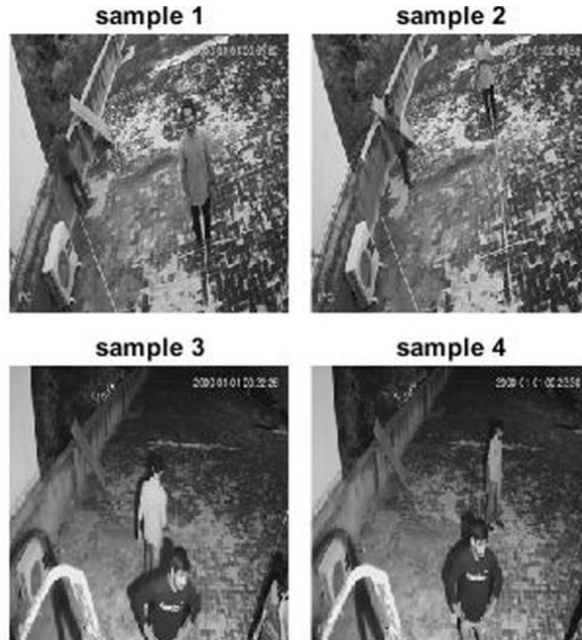


Figure 57 YOLO Training Dataset Sample

In addition to images from *Large Scale Multi-Camera Detection Dataset Kaggle* were included.

(B) ROI Generation

Next challenge was to create ROIs for each image for the indigenous dataset. Creating manual ROIs for all the images for this huge data was not possible. In this regard *Ground Truth Labeller* tool was used to automate the process. Viola Jones algorithm was used as base algorithm to automate the process as shown in Figure 58. Viola Jones is efficient in memory allocation on the other hand CNN is faster [275].

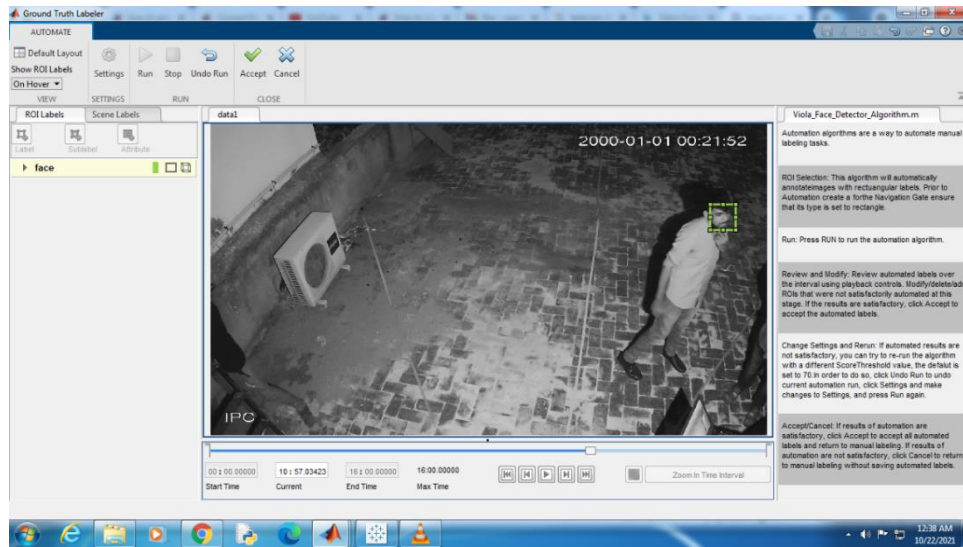


Figure 58 ROI Automation with Custom Algorithm

As discussed in the earlier section, Viola Jones has a limitation that it is unable to detect the small faces. So, where it misses the faces, manual ROIs were created. CCTV footage of humans was captured and 0.5 frame per second was captured. The process of creating ROIs from the video streams is shown in Figure 59 and Figure 60.

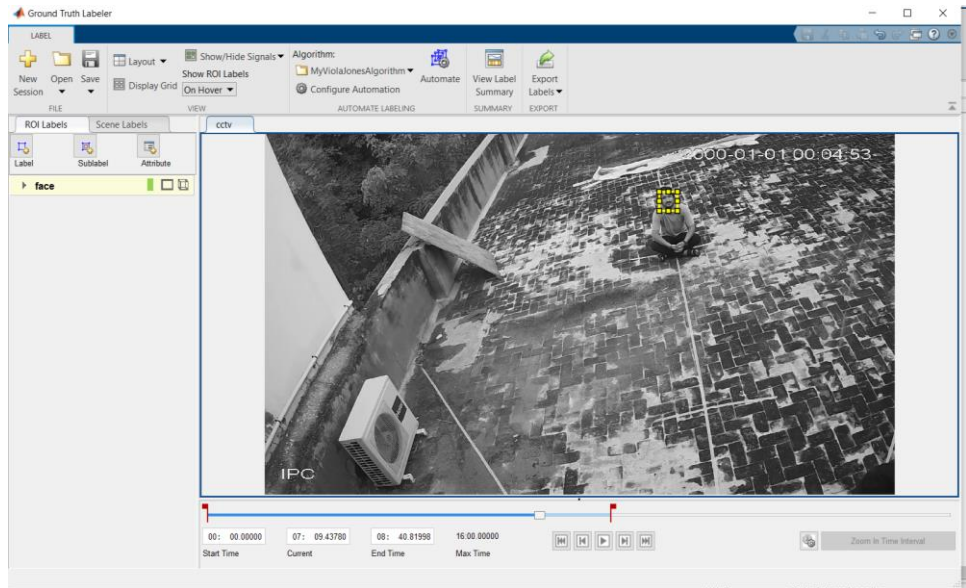


Figure 59 ROI Generation in CCTV Footage

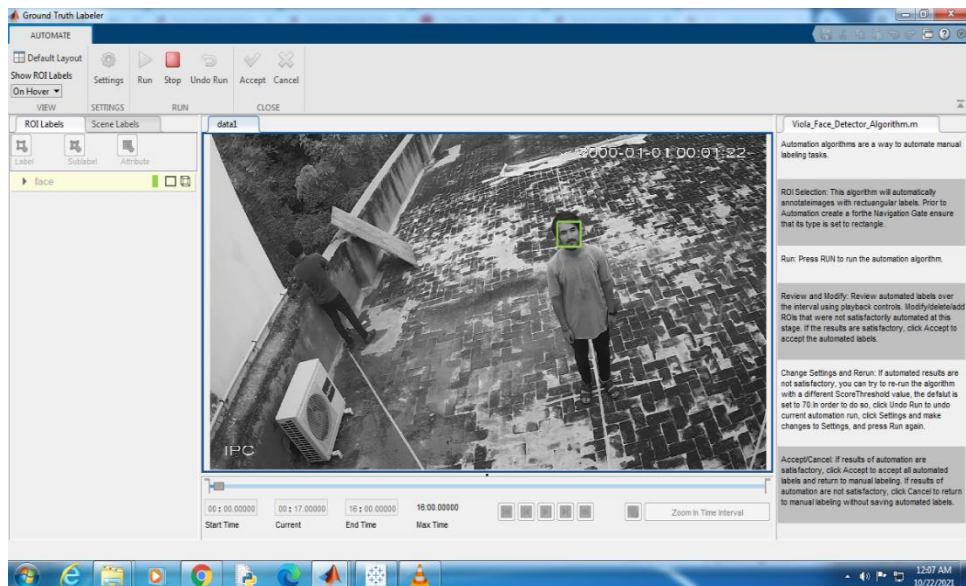


Figure 60 ROI Generation in CCTV Footage B

Training network with the dataset having dimensions of 2792 x 1571 had a high computational task. For solving this issue images were downsampled to 2161 x 1216 keeping 46 PPF at the maximum range. Figure 61 shows the rescaling of the image and rescaled ROI. Boundary box parameters were divided by 1.2919 to get the ROIs for the downsampled image of training. Image names were kept same, whereas ROI column for image labelling was generated transforming the extracted ROIs matrix.

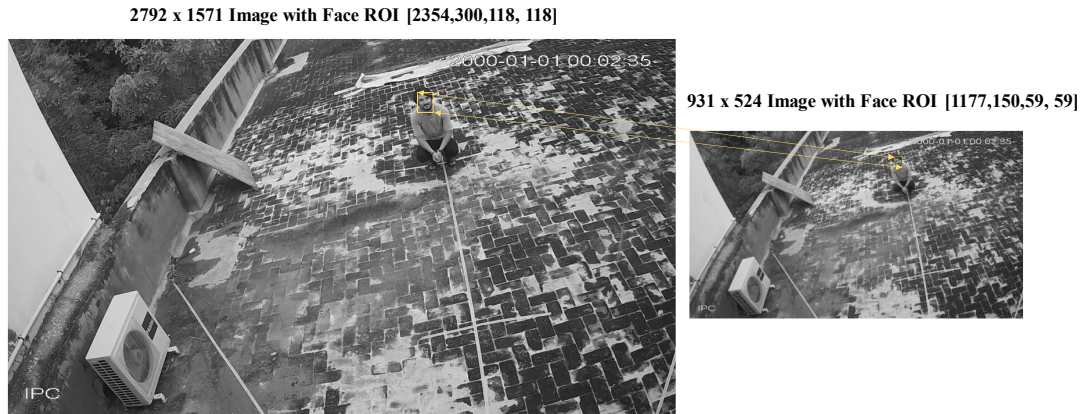


Figure 61 ROI Rescaling

(C) Training the Network

To detect face images from CCTV camera, the challenge is fast detection and small faces. This task is quite complex as detection of small objects is challenging and detection time is also critical.

i. Input Layer Dimension Setting

YOLOv3 network input size should be kept smaller to avoid huge computations. In this case the training images from the dataset and CCTV camera, size of the image was 2792 x 1571. The network input size cannot be made equal to 2792 x 1571 due to high computation cost. The ideal YOLOv3 input size can be either 416x416 or 608x608. One solution is to resize the image to 608x608 which is not recommended as it will cause not only the loss in image resolution but also affect ROI, as small ROIs will be eliminated. The solution is to break the input image in size of 608x608. This will not affect the ROI and the image resolution will also remain intact. As a result, detection of small faces can be easily done. If the image resolution is kept 2792x1571 then it will be divided into 15 images of 608 x 608 with paddings added in the images 5, 10, 11, 12, 13, 14 and 15 as shown in Figure 62. The frame rate for YOLO v3 for 608 x 608 image using RTX 2060 Super was found to be 55 frames per second. If the image is kept at the original dimensions, then it needs to be broken into 15 frames as shown in Figure 62. The frame rate to detect faces using YOLO v3 from 2792 x 1571 with the same GPU was found to be 18 fps.

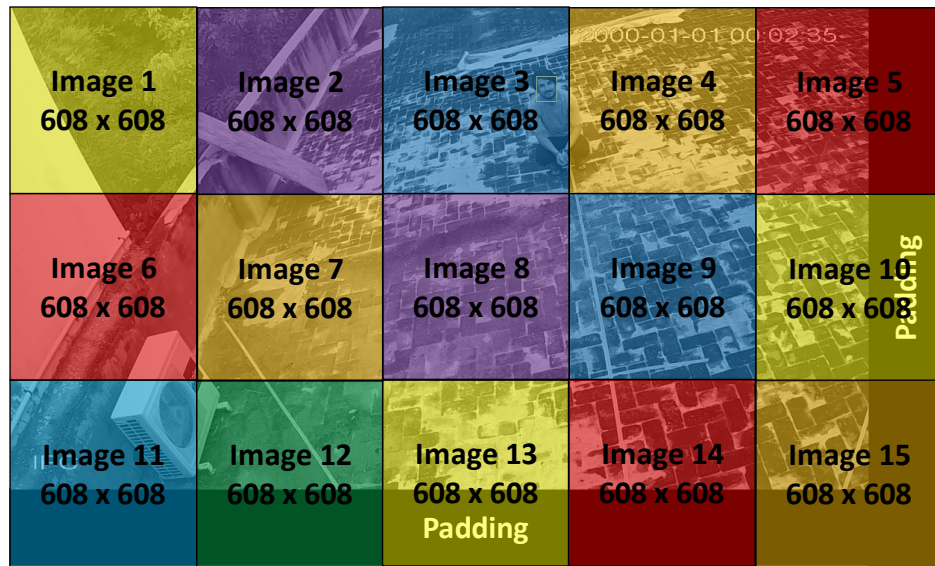


Figure 62 2792 x 1571 Image Breakdown Scheme

This frame rate can be improved by reducing the dimensions in such a way that the 20 PPF rule for face detection is not violated and the size is kept at 46 PPF. One dimension is made exact multiple of 608 to avoid padding on one side with the same aspect ratio. The new dimension is calculated to be 2161 x 1216. That is 1.292 times smaller than the original input image. Now the image has 46 PPF at the maximum range. 2161 x 1216 image is divided into 8 images of 608 x 608 with paddings added in the images 4 and 8 as shown in Figure 63. 271 x 608 paddings are added in both the images. The data is prepared in this way such that no ROI is lost.

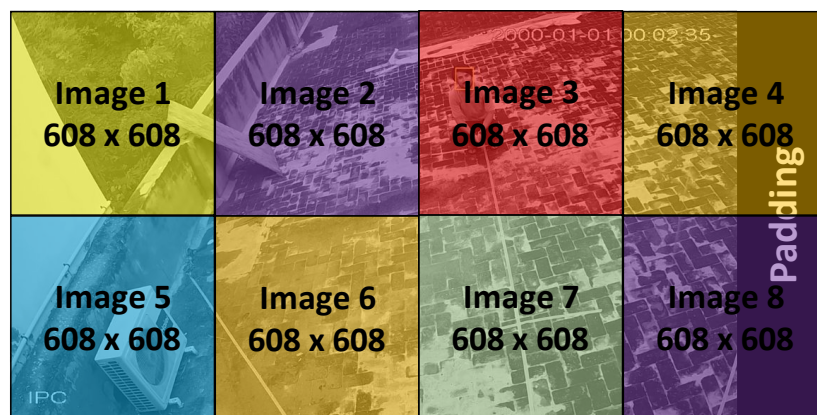


Figure 63 2161 x 1216 Image Breakdown

With the reduction of the dimensions the frame rate for detection was increased to 30 fps. In the worst scenario, that every second Raspberry Pi receives a trigger, GPU can handle 15 cameras with frame rate of two frames per second. There are eight GPUs

per GPU rig proposed. One GPU rig can handle 120 cameras. For a camera network like Lahore having 12000 cameras, can be handled by 100 distributed high-performance computers for detection. The number of computing devices can be decreased by using superior GPUs.

ii. Training Parameters

To implement YOLOv3, Darknet is used which is an open-source neural network framework which is C based and supports CUDA for processing. Darknet provides a building block for implementing architecture of YOLOv3. For the training of the network, Network parameters are given in Table 8.

Table 8 Network Parameters

Parameter	Value
Batch	64
Subdivisions	16
Height	608
Width	608
Channels	3
Momentum (optimizer)	0.9
Decay (optimizer)	0.0005
Angle (Image Augmentation)	0
Saturation (Image)	1.5
Exposure (Image)	1.5
Hue (Image)	0.1
Learning rate	0.001

V. Frame Resizing for Face Detection

First task to detect faces out of the already stored images by the parallel computing nodes distributed in the zone. For detection YOLO v3 trained model was used. As proven earlier the frame stored was of the dimension 2792 x 1571. YOLO v3 is used for real time face detection, input layer settings are discussed in the above section.

As explained earlier, referring to the DORI. Image should be represented by 60 PPF for identification, 40 PPF for recognition and 20 PPF for detection. 2792 x 1571 frames

are already stored for identification, but the dimensions of the image can be reduced for detection purposes.

was then designed that rescale the image to 2161 x 1216, keeping the same aspect ratio. This image is downscaled to 1.292 times of the original image and representing 46 PPF at the maximum range. The input image is then divided to eight parts of 608 x 608 and padding of black is added to the additional part. As shown in the Figure 64, the input image is the frame that is uploaded by the Raspberry Pi's having dimensions of 2792 x 1571 is resized to 2161 x 1216. This image is divided into eight sub-images of 608 x 608 pixels each. Face is detected in sub-image 3, having ROI of [110, 168, 66, 66] or x=110, y=168, h=66 and w=66.

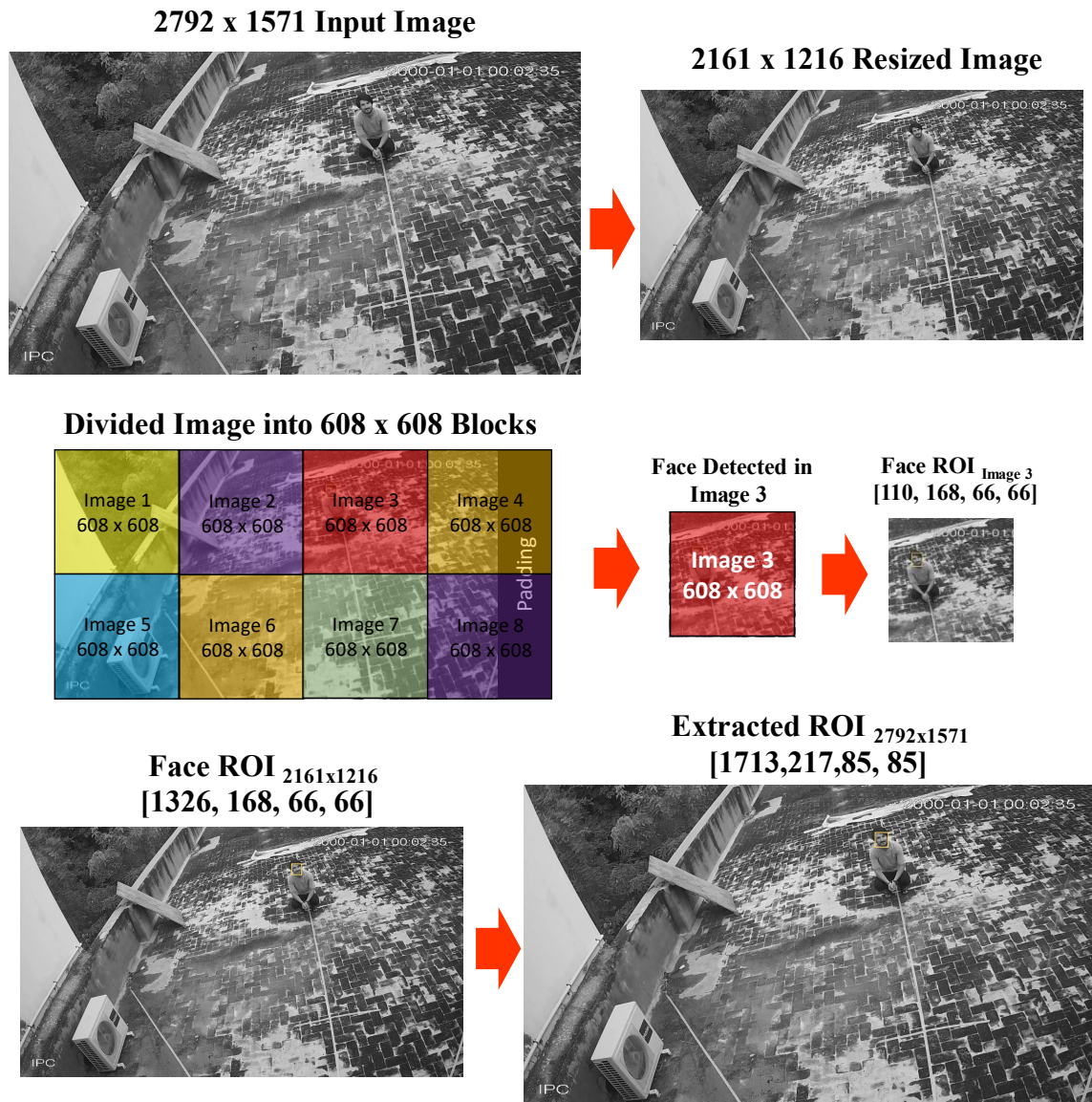


Figure 64 YOLO Face Detection Process

As this face is extracted from third sub-image then to find the coordinates for 2161 x 1216, following expressions Eqn. 4-58, Eqn. 4-59, Eqn. 4-60 and Eqn. 4-61 were used.

$$x_{2161 \times 1216} = \begin{cases} x_n + [608 \times (n - 1)], & n \leq 4 \\ x_n + [608 \times (n - 5)], & n > 4 \end{cases} \quad \text{Eqn. 4-58}$$

$$y_{2161 \times 1216} = \begin{cases} y_n, & n \leq 4 \\ y_n + 608, & n > 4 \end{cases} \quad \text{Eqn. 4-59}$$

$$h_{2161 \times 1216} = h_n \quad \text{Eqn. 4-60}$$

$$w_{2161 \times 1216} = w_n \quad \text{Eqn. 4-61}$$

Where n is sub-image number.

Till this stage the image of 2161 x 1216 resolution, can give 46 PPF at the maximum range, if face is cropped from this resolution it can lead to lesser accuracy for recognition. That is why the face is cropped out from the original stored image. Eqn. 4-62, Eqn. 4-63, Eqn. 4-64 and Eqn. 4-65 are used to calculate ROIs for 2792 x 1571 image.

$$x_{2792 \times 1571} = x_{2161 \times 1216} \times 1.292 \quad \text{Eqn. 4-62}$$

$$y_{2792 \times 1571} = y_{2161 \times 1216} \times 1.292 \quad \text{Eqn. 4-63}$$

$$h_{2792 \times 1571} = h_{2161 \times 1216} \times 1.292 \quad \text{Eqn. 4-64}$$

$$w_{2792 \times 1571} = w_{2161 \times 1216} \times 1.292 \quad \text{Eqn. 4-65}$$

VI. Face Cropping

It is already discussed in detail in earlier section. Middle face represents eyes, forehead, lips, and cheeks. As mentioned in the introduction, this system contains various layers of face recognition used for distribution of the detected faces into categories that makes the computation tasks less expensive. Faces are to be categorized so that the target must be searched from the relevant category. There are three categories or layers from which each face must be pass through. First layer is for gender, then face type and last one is the facial hair style. As this research is focused on Pakistan, that is why face types and facial hair styles are modelled as per the target scenarios.

For detection of first two categories, the face is needed to be cropped to extract features. But the issue is with the last stage. Beard is distinguished based on their style as well as lengths. Beards can be of many sizes and types. In this research only four categories are introduced depending upon the moustache and length of the beard. It cannot be calculated without having the lower matrix as discussed above. To calculate how long this matrix should be depending upon the face data.

No.	Anthropometric dimension	Measurement location
1	Head circumference	
2	Head height	
3	Head length	
4	Head breadth	
5	Face width	
6	Bitragion-menton arc	
7	Bitragion-subnasale arc	
8	Bizygomatic-menton arc	
9	Chin-menton length	
10	Face length(sellion - menton)	
11	Face length(sellion – bottom lip)	
12	Nose length	
13	Nose protrusion	
14	Menton-nasal bridge length	
15	Lower face length	
16	Menton-bottom lip length	
17	Menton-chin length	
18	Anterior chin projection-nasal bridge length	
19	Maximum nasal bridge breadth	
20	Nose width	
21	Lip width	
22	Chin width	

Figure 65 Anthropometric Proportions [Source: Lee et. al., 2012] [276]

Various research has been done for estimating the face/head dimensions. Facial anthropometric data was analysed for design oxygen masks for Korean pilots. Variance in the lengths of Korean and American pilots was determined [276].

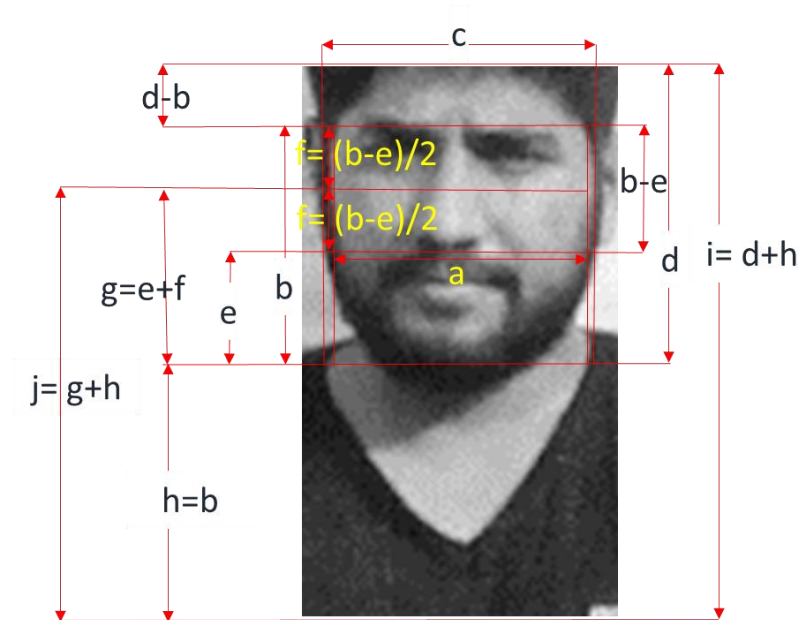


Figure 66 Facial Dimensions Key Points

Inspired by Figure 65, in Figure 66 key areas of a face are described. Width “a” is face breadth (bizygomatic), Height “b” is known as Menton-Sellion/ Face Length, “c” is Bitragion Breadth, dimension, “d” is Menton-crinion Length and “e” is known as Menton-sub-nasale length.

For face recognition part in which the face type and gender is classified is width “c” and height “d”. The average values are as; value of length face breadth and Menton-Sellion/ Face Length 5” and 4.4” respectively. Whereas average of Bitragion Breadth is 5.2”. Whereas Menton-crinion Length represents complete face including forehead having value of 7”. Menton-Subnasale length average is 2.6”.

For front aligned face, the detected face with features covers face length as “d” Menton-crinion Length and Face width as “c” Bitragion Breadth. For facial recognition 60 PPF was required, now the issue is how large the facial image would be to get recognized by the system without wasting the system time and space resources. Increasing the size for feature extraction can make computations slow and

lowering the size that the threshold will make less efficient in terms of detection. Moreover, how much should be cropped to compensate the alignment losses.

For cropping out the face, additional space on the top and sides is required. The overall width of humans including ears is known as Head Breadth having average value of 5.7". Whereas width without ear i.e., Bitragion Breadth has an average value of 5.2". And Face breadth where bizygomatic is 5". The detected face covers Bitragion Breadth. Now the additional area percentage will be calculated based on detected face breadth i.e., Bitragion Breadth. By the percentile values on both sides 0.25" will be required. It is almost 5% on each side.

On the safety margin double the percentage i.e., 10% on both sides is included to cover ears. If the cropped face already includes this area additional pixels will be there not affecting the system performance.

For the forehead height, it can be found by the differences of averages of Menton-Sellion/ Face Length (5") and Menton-crinion Length (7"). The forehead percentile is found to be 2". Detected face includes this region but for the margin 10% of the detected length is added on the top.

After deciding the face region matrix, the next part of the research is to find the region where facial hairs can be present. One solution was to just include an additional area in the lower side, but the issue is with the computations, pixels not representing the facial hairs should not be the part of the recognition stage. To estimate the points of the region, the average values are used in the research. For facial hairs analysis region of the interest is the overall length of the face, menton-sellion length (length from nose top to chin) and sub-nasle-sellion length (height of nose) to estimate the ratio of overall face length with the expected facial hairs area. Using the data for lower side limit, double the detected face length is taken for long bearded person. For upper limit it is required to estimate the position with safety margin. Midpoint of the nose length can serve the purpose for beard detection. Calculating the ratio with the mean values for the estimation.



Figure 67 Rhinion Point

Point of interest shown in Figure 67 is the Rhinion point where half nose may be captured. For this “f”; Rhinion- Subnasale length (half nose length) is found as follows in Eqn. 4-66.

$$f = \frac{b - e}{2}$$

Eqn. 4-66

Using the average values for “b” and “e”

$$f = \frac{5 - 2.6}{2}$$

Eqn. 4-67

$$f = 1.2 "$$

Eqn. 4-68

And “g” Rhinion-Menton Length (Mid nose to Chin) is using Eqn. 4-69.

$$g = e + f$$

Eqn. 4-69

Using the average values,

$$g = 2.6 + 1.2$$

Eqn. 4-70

$$g = 3.8 "$$

Eqn. 4-71

Lower Matrix length “h” for managing the long beards, extra length is kept same as of Face Length (b). Height of facial hair type matrix (j) will be found using Eqn. 4-72.

$$j = g + h \quad \text{Eqn. 4-72}$$

$$j = 3.8 + 5 \quad \text{Eqn. 4-73}$$

$$j = 8.8" \quad \text{Eqn. 4-74}$$

Here for the ratios of the values, the detected face length is Menton-crinion Length (d) that from which we need to find the Rhinion spot. By calculating the ratio of g to d in Eqn. 4-75, we can find the percentage of the area to be considered for the facial hair matrix.

$$Percentage_{Lower} = \frac{g}{d} \times 100 \quad \text{Eqn. 4-75}$$

$$Percentage_{Lower} = \frac{3.8}{7} \times 100 \quad \text{Eqn. 4-76}$$

$$Percentage_{Lower} = 54.2\% \quad \text{Eqn. 4-77}$$

Above result concludes that facial hair matrix will be cropped at lower 54% area and additional length equals to b at lower side. As per the data available of Pakistan the average beard size varies and in few cases people with long beard lives in country [277]. This issue made the cropping of the face different from the conventional method.

As shown in Figure 68, boundary boxes for the face are detected. A clear image is used here for better understanding. Two of them are the x and y coordinates of the face present in the frame and the dimensions of the detected face. Now the algorithm adjusts the dimensions making the recognition robust against the alignments and the addition of the facial hair styles.

First, the face is detected form the image and then it is cropped by adding 10% pixels one both sides and top. Ratio of averages of Menton-Sellion to Menton-crinion Length is used to find the region to be added below the menton point as Eqn. 4-78.

$$Beard Compensation = \frac{b}{d} \times 100 \quad \text{Eqn. 4-78}$$

$$Beard\ Compensation = \frac{4.4}{7} \times 100$$

Eqn. 4-79

$$Beard\ Compensation = 62.8 \%$$

Eqn. 4-80

Pixels from top edge of the forehead to the bottom edge of the chin are calculated to find the lower portion to be added for long beard people. It is to be noted that this area of the face is the region where majority facial features are present and here, we are taking same pixels additional downwards. 63% vertical pixels are added at the bottom side of the detected face. Percentage lower along with the beard compensation makes the total facial hair matrix. 10% of the pixels are added on the upper side for forehead margin. More portion of the forehead was required for the better results. 10% of the compensation area has been cropped on each side of the face to deal with the alignment issues.

Final coordinates of the complete image are found using Eqn. 4-81, Eqn. 4-82, Eqn. 4-83 and Eqn. 4-84.

$$x_{final} = x_{2792 \times 1571} - 0.1 \times w_{2792 \times 1571}$$

Eqn. 4-81

$$y_{final} = y_{2792 \times 1571} - 0.1 \times h_{2792 \times 1571}$$

Eqn. 4-82

$$w_{final} = 1.2 \times w_{2792 \times 1571}$$

Eqn. 4-83

$$h_{final} = 1.63 \times h_{2792 \times 1571}$$

Eqn. 4-84

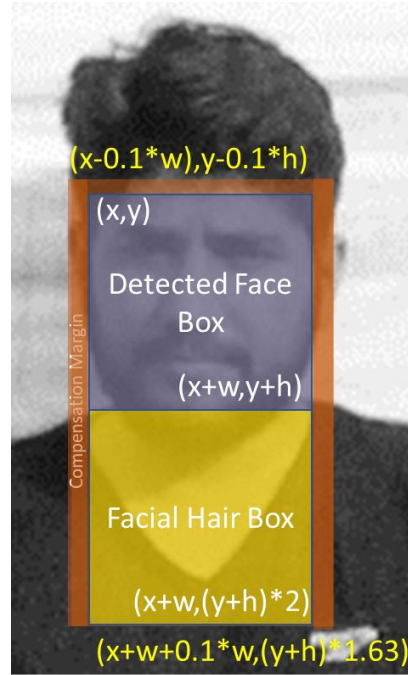


Figure 68 Compensation Matrix Calculation

Face matrix coordinates is cropped as.

$$x_{face} = x_{final} \quad \text{Eqn. 4-85}$$

$$y_{face} = y_{final} \quad \text{Eqn. 4-86}$$

$$w_{face} = w_{final} \quad \text{Eqn. 4-87}$$

$$h_{face} = h_{2792x1571} \quad \text{Eqn. 4-88}$$

Facial hair matrix coordinates are cropped as.

$$x_{f_{hair}} = x_{face} \quad \text{Eqn. 4-89}$$

$$y_{f_{hair}} = y_{face} + 0.46xh_{face} \quad \text{Eqn. 4-90}$$

$$w_{f_{hair}} = w_{face} \quad \text{Eqn. 4-91}$$

$$h_{f_{hair}} = 0.54xh_{face} + 0.63xh_{face} \quad \text{Eqn. 4-92}$$

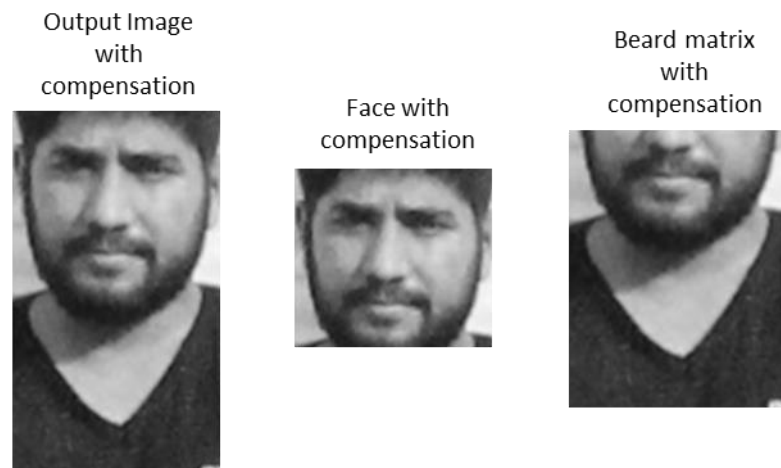


Figure 69 Compensation Face Matrices

Output of the system after cropping is shown in Figure 69. The trained system cannot detect faces with horizontal rotated having Yaw more than 40 degrees, it is shown in results section. This issue can be resolved by the presence of multiple cameras covering the area with different angles. Face was not detected from image shown in Figure 70, having yaw more than 40 degrees.



Figure 70 70 Degrees Rotated Face

In Figure 71, the system successfully able to detect face from the input image. Face in the image was rotated horizontally by 30 degrees off camera. For face alignment extra pixels are required to get frontal face. This feature of the research can be used for assisting the officials to estimate the frontal face.



Figure 71 30 Degrees Rotated Face

Figure 72 shows the output of the system with and without compensation regions.

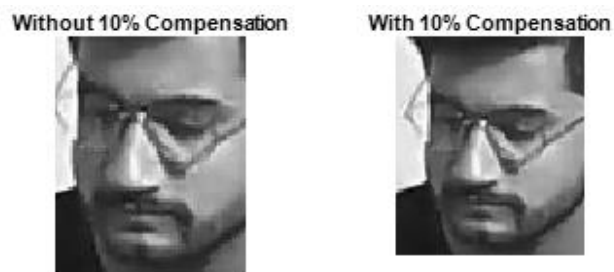


Figure 72 System Results with and Without Compensation Regions

Here an issue comes up. If the face is present in any of the edges as in Figure 73 and compensation pixels are not present in the frame, then pixels are added to the sides of the face. To make it like the normal image in dimensions.

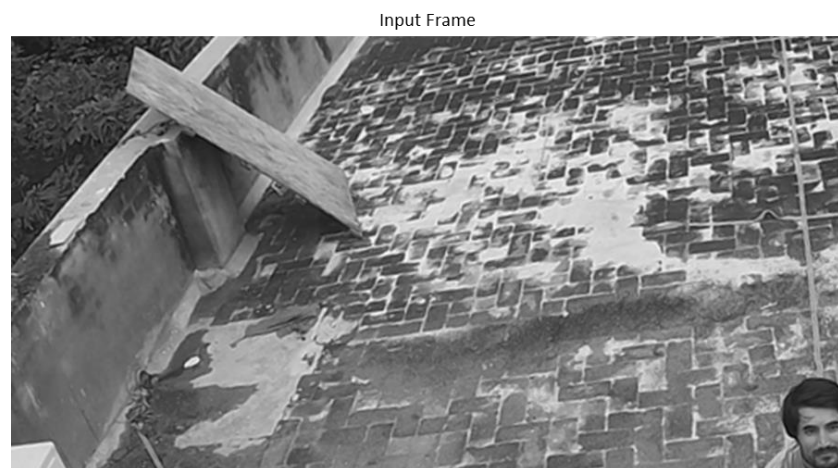


Figure 73 Input Frame with right bottom face

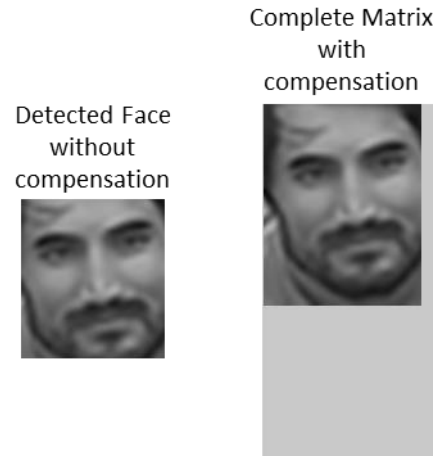


Figure 74 Output Images with and without compensation



Figure 75 Input Images for Gender/Face Type and Facial Hair Stage

Figure 74 and Figure 75 shows a cropped face and facial hair matrix respectively from a frame placed at the bottom right corner where there was no space for compensation rectangles at right and bottom. Algorithm checks the position of the detected face coordinates and performs following actions.

if : $x_{final} < 1$ then add pixels on left

if: $y_{final} < 1$ then add pixels on top

if: $x_{final} + w_{final} > 2792$ then add pixels on right

if: $y_{final} + h_{final} > 1571$ then add pixels at bottom

VII. Face Resizing Calculation

Now the concern is what should be the size of the face to be stored that can be processed for feature extraction. What we need is minimum size that can be attained without affecting the performance. For recognition purposes the 40 PPF was required and Identification 60 PPF. The proposed system has been designed to scale down the image in such a way that person at the maximum range can be recognized i.e., 50 feet with existing hardware. All values can be calculated by the help of the equations stated in previous sections.

It should be noted here that for facial features; Bizygomatic breadth/ face width is important. It's already been discussed with the help of literature. Applying the DORI rule 60 PPF are required to represent this section, not the overall head width or more PPF will be required if we want to include head width.

As described and proved in earlier sections, the range of the camera in such a way that one foot must be represented by at least 60 pixels. As per research survey conducted with total of 3000 samples having 974 females and 2026 males, the mean width for males were 147.6 mm (6" approx.) and 140.1 mm (5.5" approx.) whereas mean length for males was 117 mm (4.6" approx.) and 109.7mm (4.3" approx.) for females. This value can vary with region and BMI [278]. This length is the menton-sellion length and width is bizygomatic breadth.

We can estimate the final size of the image to be stored by using the averages or percentile values. These calculations only require the width measurements.

Worst scenario calculation includes the smaller size faces. That is why calculations for resizing should be made by using small size face data. As per research conducted for small and medium size faces by Y C Lin and C P Chen uses mean value of bizygomatic breadth from data of U.S. "National Institute for Occupational Safety and Health" generated from people living in U.S. Mean bizygomatic breadth for males was $128.2\text{mm} \pm 6.6$ (5" ± 0.25) and $122.7\text{mm} \pm 7.0$ (4.8" ± 0.28) for females [279].

In previous calculations the overall cropped face includes additional area of the face that does not include the facial features. For this the ratio of the overall width bitracion breadth and bizygomatic breadth was found. For various percentiles, it was found that

the bizygomatic breadth remains around 95-96% of bitragion breadth. Now we must find the number of pixels to be available to represent the final face to be stored.

With the data discussed in the previous section the percentiles of Bizygomatic breadth and Bitragion Breadth shows that Bitragion Breadth is around 4% more than that of Bizygomatic breadth.

For assignment of pixels to the face, what we must estimate is the percentage of Bizygomatic breadth of foot. For dedicating more pixels to a face, we must consider the largest value the 99th percentile, which is 6.1". For easier understanding the 51% of a foot should be dedicated to representing a face.

Axis Communications also defines the pixel density concept in their article "*Perfect Pixel Count, Meeting your operational requirements*". It states that the average face width is 6.3" and keeping in view the Swedish National Laboratory of Forensic Science verified by the tests conducted by Axis Communications, following pixel density requirements were set [280].

Table 9 Pixel Density Requirements [Source: Axis Communications 2022] [280]

Requirement	Pixels per Face
Identification (Difficult Conditions)	80
Identification (Favorable Conditions)	40
Recognition	20
Detection	4

Table 9 by shows Axis Communications 2022, similar requirements for recognition but 16 more pixels for identification. We can use the 40 pixels per face or 60 pixels per foot requirement. Here it shows that face is represented by 52.5% of a foot. 1.5% more than our above shown calculations.

For the pixel allocation, 40 pixels are required for Bizygomatic breadth, 4% is additional to represent Bitragion Breadth i.e., 1.6 pixels. Now the total will be 42 pixels to represent the Bizygomatic breadth. For the sides compensation 10% pixels are added on each side. Making total 50 horizontal pixels to represent the complete head breadth.

For vertical pixels, consider Menton-crinion Length, and like the horizontal pixel density, we are using the 99th percentile i.e., 8.5" long face. If we consider 76 pixels

per foot density, the overall length will be requiring 54 pixels to represent a face. We can use 50 vertical pixels without affecting much ratio.

For the recognition of beard additional pixels would be required at the bottom side. As per the calculations, the ratio of Menton-Sellion to Menton-crinion was found to be 1:1.63. That is why here we need to add 63% (32 pixels) more vertical pixels. Making a total of 82 pixels.

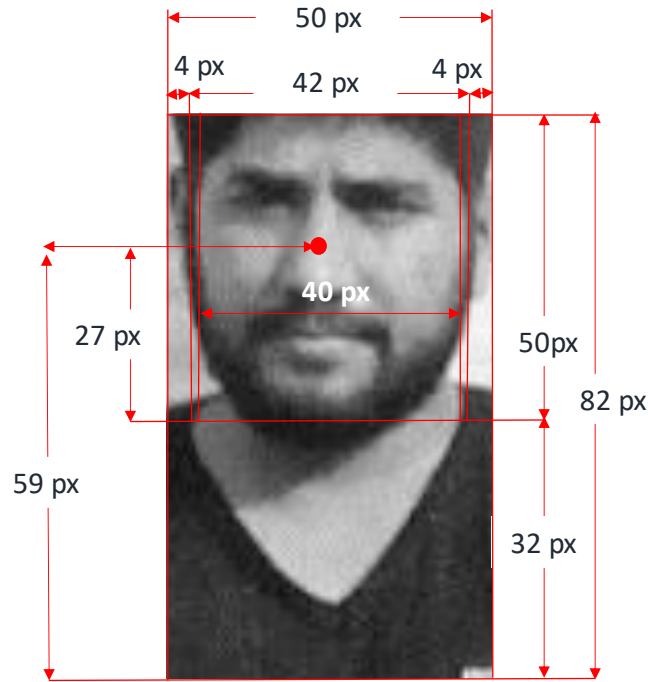


Figure 76 Pixels Allocation for 50 x 82 Image

As per the DORI rule, Face Breadth must be represented by 40 pixels. Following calculations are made in Eqn. 4-93, Eqn. 4-94, Eqn. 4-95, Eqn. 4-96, Eqn. 4-97, Eqn. 4-98, Eqn. 4-99 and Eqn. 4-100.

$$\text{Face Breadth} = 40 \text{ pixels}$$

Eqn. 4-93

$$\text{Bitracion Breadth} = 1.04 \times \text{Face Breadth}$$

Eqn. 4-94

$$\text{Bitracion Breadth} = 42$$

Eqn. 4-95

$$\text{Side Compensation} = 0.1 \times \text{Face Breadth}$$

Eqn. 4-96

$$\text{Side Compensation} = 4 \text{ pixels}$$

Eqn. 4-97

$$Width_{total} = 42 + 4 \times 2$$

Eqn. 4-98

$$Width_{total} = 50 \text{ pixels}$$

Eqn. 4-99

$$Height_{face} = 50 \text{ pixels}$$

Eqn. 4-100

Figure 76 represents the image to be stored in the database for 60 PPF. Two matrices are extracted from the image. From this image two matrices are extracted, one for facial features extraction and the other one for facial hair analysis. For the facial features extraction top matrix of 50 x 50 pixels is used. This matrix is used to detect the gender and the face type. Whereas for the facial hair, a lower matrix with additional pixels was added. As discussed earlier the facial hairs detection does not require complete face. That is why the Rhinion average spot was calculated to include the sub-nasle area with a safety margin. Calculations and data discussed in the previous section state that Menton-crinion Length was 7” and the Rhinion to Menton Length average is 3.8”. Now we need to find the vertical pixels to be cropped using Eqn. 4-101.

$$Facial \ Hair \ Vertical \ Pixels_{Face} = \frac{Rhinion - crinion \ Length}{Menton - crinion \ Length} \times Height_{face}$$

Eqn. 4-101

$$Facial \ Hair \ Vertical \ Pixels_{Face} = \frac{3.8}{7} \times 50$$

Eqn. 4-102

$$Facial \ Hair \ Vertical \ Pixels_{Face} = 27$$

Eqn. 4-103

$$Facial \ Hair \ Vertical \ Pixels_{Additional} = 50 \times 0.63$$

Eqn. 4-104

$$Facial \ Hair \ Vertical \ Pixels_{Additional} = 32$$

Eqn. 4-105

$$\begin{aligned} Facial \ Hair \ Vertical \ Pixels_{Total} \\ = Facial \ Hair \ Vertical \ Pixels_{Face} \\ + Facial \ Hair \ Vertical \ Pixels_{Additional} \end{aligned}$$

Eqn. 4-106

$$Facial \ Hair \ Vertical \ Pixels_{Total} = 27 + 32$$

Eqn. 4-107

$$Facial\ Hair\ Vertical\ Pixels_{Total} = 59$$

Eqn. 4-108

Image size was optimized using the 60 PPF at the working distance of 50 feet but the case here is that is the worst scenario i.e., the size of the human face for the maximum range. But for the targets near to camera can give better results.

For the next processing stage, the image must pass through the recognition layers. The input size for the layers is 224 x 224. If the target is present closer to the camera more pixels can represent the face.

Till now 2793 horizontal pixels are represented at 60 PPF at the range of 50 feet. Face is complete face is represented by 50 horizontal pixels. Keeping the same ratio, PPF for 224 pixels face can be found using Eqn. 4-109.

$$\frac{Pixels_{full\ face\ @\ 224}}{Pixels_{foot\ @\ 224}} = \frac{Pixels_{full\ face\ @\ 50}}{Pixels_{foot\ @\ 50}}$$

Eqn. 4-109

$$\frac{224}{Pixels_{foot\ @\ 224}} = \frac{50}{60}$$

Eqn. 4-110

$$Pixels_{foot\ @\ 227} = \frac{60}{50} \times 224$$

Eqn. 4-111

$$Pixels_{foot\ @\ 224} = 268.8$$

Eqn. 4-112

Finding horizontal range represented by 2793 pixels at 269 PPF using Eqn. 4-113.

$$Horizontal\ Range_{@227} = \frac{Horizontal\ Pixels}{Pixels_{foot\ @\ 224}}$$

Eqn. 4-113

$$Horizontal\ Range_{@227} = \frac{2793}{269}$$

Eqn. 4-114

$$Horizontal\ Range_{@227} = 10.38$$

Eqn. 4-115

Using Eqn. 4-116, working distance for horizontal range of 10.38 feet is found to be 11 feet.

$$Horizontal\ Range = 2 \times Working\ Distance \times \tan\left(\frac{HFOV\ [degrees]}{2}\right)$$

Eqn. 4-116

$$Horizontal\ Range_{@224} = 2 \times Working\ Distance \times \tan\left(\frac{50}{2}\right)$$

Eqn. 4-117

$$10.38 = 0.9326 \times Working\ Distance$$

Eqn. 4-118

$$Working\ Distance = 11.13 \approx 11\ feet$$

Eqn. 4-119

Above results show that targets present between 6.8 and 11 feet will not experience pixel loss in downscaling. 6.8 feet is the minimum distance calculated for the camera to cover. These calculations are for specific camera resolutions and pixels per foot capacity. Working distance and horizontal distance can be increased by changing the variables as explained in Frame Dimensions Optimization section. But with a trade-off with the computation speed.

The final image must be resized to 224 x 224, target closer to the camera means clearer the image will be. Number of pixels dedicated to each part of the image is calculated for 224 x 224 image as follows.

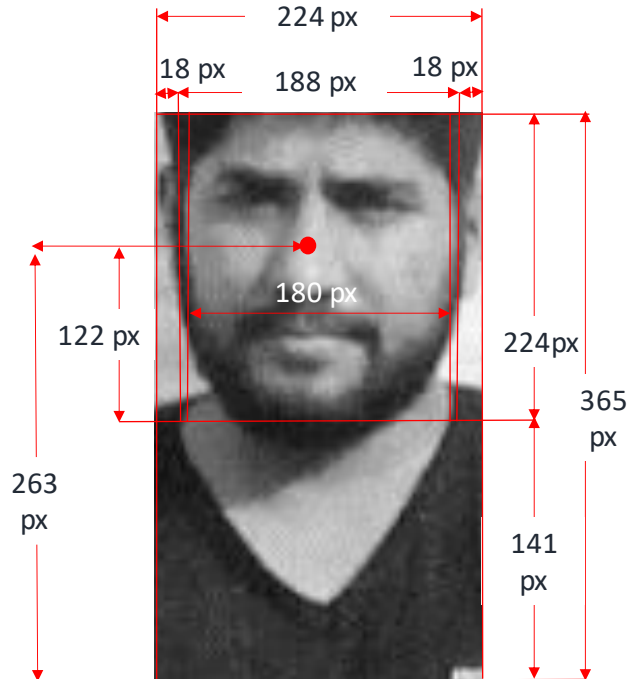


Figure 77 Pixel Division for 224 x 365 Image

In previous calculations, with the average human face sizes, it was found that average Bitracion Breadth is 4% more than face breadth (bizygomatic) and 10% additional pixels are included for tilt compensation. $Width_{total}$ is found using Eqn. 4-120, Eqn. 4-123 is used to find Face Breadth, Bitracion Breadth is calculated using Eqn. 4-126 and Eqn. 4-128 finds the Side Compensation. These parameters are represented in Figure 77.

$$Width_{total} = Bitracion\ Breadth + 2 \times (0.1 \times Face\ Breadth) \quad \text{Eqn. 4-120}$$

$$Width_{total} = 1.04 \times Face\ Breadth + 0.2 \times Face\ Breadth \quad \text{Eqn. 4-121}$$

$$Width_{total} = 224\ pixels \quad \text{Eqn. 4-122}$$

$$224 = 1.24 \times Face\ Breadth \quad \text{Eqn. 4-123}$$

$$Face\ Breadth = \frac{224}{1.24} \quad \text{Eqn. 4-124}$$

$$Face\ Breadth \approx 180\ pixels \quad \text{Eqn. 4-125}$$

$$Bitracion\ Breadth = 1.04 \times Face\ Breadth \quad \text{Eqn. 4-126}$$

$$Bitracion\ Breadth \approx 188\ pixels \quad \text{Eqn. 4-127}$$

$$Side\ Compensation = 0.1 \times Face\ Breadth \quad \text{Eqn. 4-128}$$

$$Side\ Compensation = 18\ pixels \quad \text{Eqn. 4-129}$$

$$Height_{face} = 224\ pixels \quad \text{Eqn. 4-130}$$

Image shown in Figure 76 is the final image that is stored in the database. From this image two matrices are extracted, one for facial features extraction and the other one for facial hair analysis as shown in Figure 78

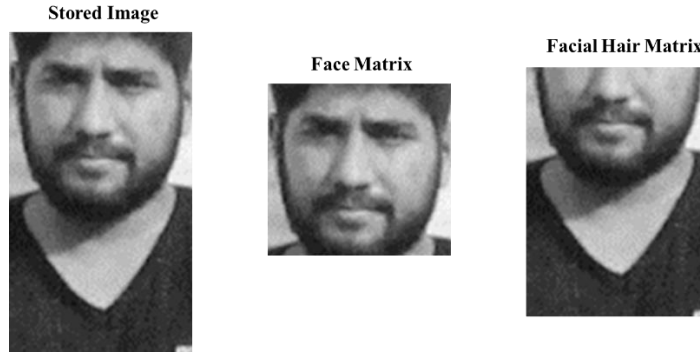


Figure 78 Face and Facial Hair Matrices Pixel Allocation

For the facial features extraction, the top matrix of 224 x 224 pixels is used. This matrix is used to detect the gender and the face type. Whereas for the facial hair, lower matrix with additional pixels is added. As discussed earlier the facial hairs detection does not require complete face. Facial hair matrix for 224-pixel wide face is calculated in the same way using Eqn. 4-131, as of the 50-pixel wide face already explained.

$$Facial\ Hair\ Vertical\ Pixels_{Face} = \frac{Rhinion - crinion\ Length}{Menton - crinion\ Length} \times Height_{face} \quad \text{Eqn. 4-131}$$

$$Facial\ Hair\ Vertical\ Pixels_{Face} = \frac{3.8}{7} \times 224 \quad \text{Eqn. 4-132}$$

$$Facial\ Hair\ Vertical\ Pixels_{Face} \approx 122 \quad \text{Eqn. 4-133}$$

$$Facial\ Hair\ Vertical\ Pixels_{Additional} = 224 \times 0.63 \quad \text{Eqn. 4-134}$$

$$Facial\ Hair\ Vertical\ Pixels_{Additional} = 141 \quad \text{Eqn. 4-135}$$

$$\begin{aligned} Facial\ Hair\ Vertical\ Pixels_{Total} \\ &= Facial\ Hair\ Vertical\ Pixels_{Face} \\ &+ Facial\ Hair\ Vertical\ Pixels_{Additional} \end{aligned} \quad \text{Eqn. 4-136}$$

$$Facial\ Hair\ Vertical\ Pixels_{Total} = 122 + 141 \quad \text{Eqn. 4-137}$$

$$Facial\ Hair\ Vertical\ Pixels_{Total} = 263 \quad \text{Eqn. 4-138}$$

The final dimensions for the facial hair matrix will be 224 x 263. That will be then transformed into 224 x 224 by the recognition layer.

VIII. Face ID Stamping

Till above step, the new face is available with dimensions of 224 x 365 in grayscale. Faces are cropped from the frames with the names tagged with the details of the date, time, and camera IDs. Now each frame is checked for the presence of the face and faces are cropped out from the frame. Each frame may contain multiple faces. Now these face images are stored to the cloud that can be accessed by the other computing system. Face number is added at the end of the file name and rest of the file name remains same.

$$filename_{face} = filename_{frame} \cup face_{number}$$

For example, two faces are detected from a frame having name of, **GBR0231012201221312451**. This image is uploaded by camera having ID of **GBR023101** at **13:12:45** on **22-01-2022** and it is first frame of the trigger. Three digits are reserved to represent the face number, which means number of faces may reach maximum of 999 faces. The two face images will have names of **GBR0231012201221312451001** and **GBR0231012201221312451002**.

4.04 Parallel Computing Stage III

Till the previous stage face images are uploaded to the cloud database and resized to appropriate size. Now, the third computing stage is responsible for passing the face image through series of classification layers and categorizing them. Face images are accessed from cloud database as shown in Figure 30 and perform cascaded classification operations as shown in Figure 31. As discussed in the introduction there are three layers made for categorizing. It is done to make the final matching less complex and the final matching for input target image should be compared with fewer images. Before recognition we need to train a system with strong classifiers having capability of recognizing IDs in an efficient way. Hence this block got two steps, first is training a model and the second one is recognition through the model. Face Categorization Workflow is shown in Figure 79.

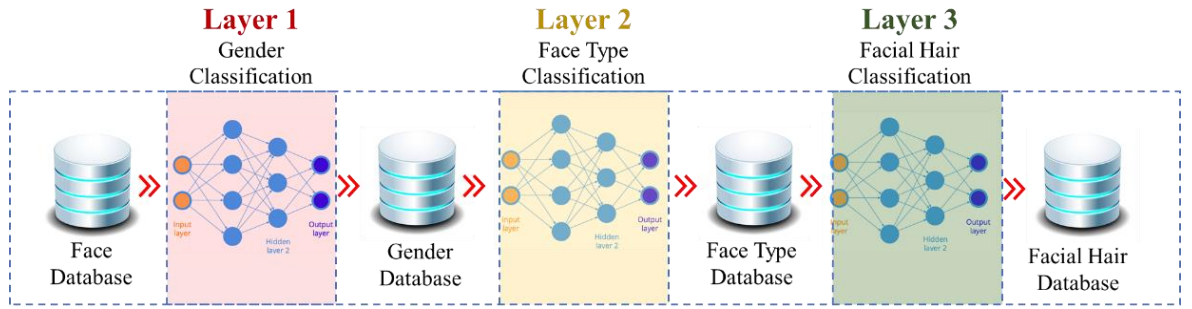


Figure 79 Face Categorization Workflow

I. Selection of Classification Layers

While looking at someone's face, we usually focus on eyes, lips, eyebrows, and hair. We also do a quick assessment on the facial framework, which includes the bone structure and ratios of various parameters [281]. Faces are distinguished by the races, ethnicity and gender [282]. Various features of the face play their role in defining a face type. Importance in terms of percentages is given in Figure 80.

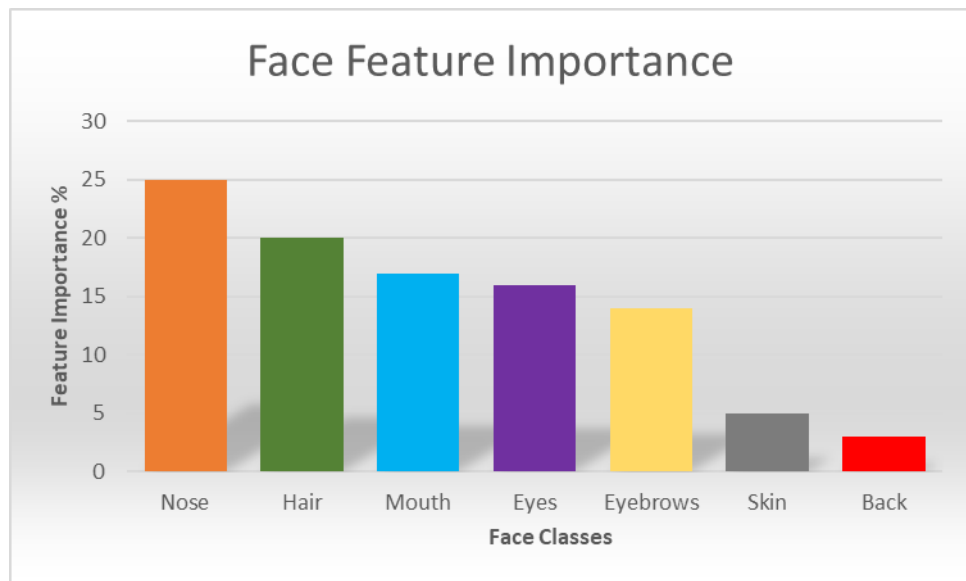


Figure 80 Feature Importance by Face Elements [Adapted from Khan et. al., 2021] [294]

Keeping in view the above parameters, total three stages of the face tagging were established. The first step is to determine the gender of the face. Pakistan has total population of 250 million and from which around 20 million people are associated to Lahore city. Lahore has around 50% of gender parity. If face is detected and classified into the gender, half of the recognition computations can be saved while looking for some specific target [283]. Multiple works have been done in the field of gender

detection for various applications. *Deepgender*: real-time gender classification using deep learning for smartphones is developed by Haider et. al. [284] and achieved an accuracy of 97.09%. It has ability to deal with the misaligned faces. Kumar et al. [285] proposed a novel algorithm for automatic live Gender Recognition utilizing Support Vector Machine (SVM) classification. Their approach integrated feature extractors like Bag of Words (Bow), SIFT, and K-means clustering to achieve efficient results on different standard datasets. Through testing on FEI, Live Images, and SCIEN databases, their methodology demonstrated high detection rates: 98% in FEI dataset, 94% in Live/Own dataset, and 91% in SCIEN dataset, respectively It has ability to detect face from the image first then classify its gender.

Thangaraj et. al. [2021] [286] proposed to enhance object identification and recognition through CCTV cameras by implementing a motion-based recording approach, thereby reducing data storage costs. Gender classification was fundamental aspect of this research. Factors including lighting, poses, and expressions can lead to challenges in accurate gender classification. To address this, the paper employs Haar cascades for real-time human face detection, followed by cropping and processing the detected face with Inception V3 for gender classification. The research utilizes the IMDB datasets for both training and testing, achieving an impressive gender classification accuracy of 97.4% [286].

The second step is to mention the face type of the detected face. Gene has an impact on deciding the facial features/shape [287]. Face style and shape is dependent upon gender and ancestry. Scientists have modelled the face shape by DNA. They also made it possible to estimate facial features/ shape of even extinct humans [288]. Basic face architecture by gender and ancestry is shown in Figure 81.

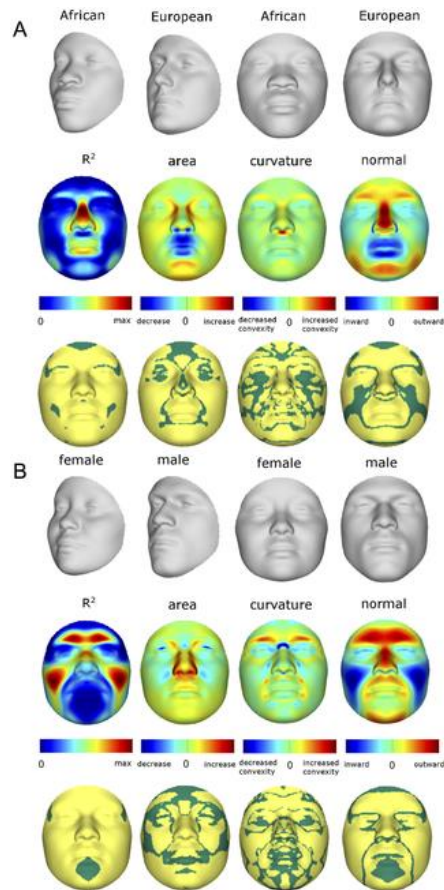


Figure 81 Face Architecture by Gender and Ancestry [Source: Claes et. al., 2018] [288]

Pakistan is a country with multiple ethnic groups living side by side. If Pakistan is observed geographically, it is in the middle of the four different types of sub-races. Pakistan shares its border within the north.

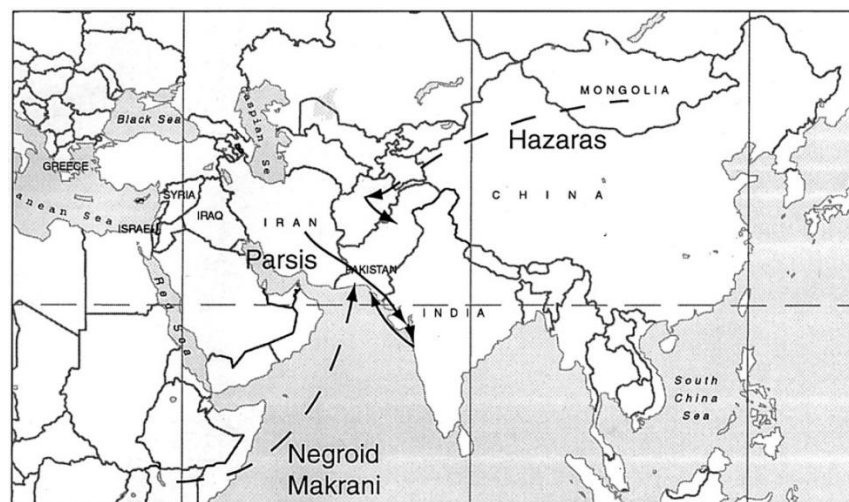


Figure 82 Subcontinent Population Origin [Source: Qamar et. al., 2002] [289]

It is a known fact that Asian race is present in China (Tibet) and at the Pakistani side they are often known as Balti people. Hazara people are also from the same area and tribe of Genghis Khan. If we move to eastern side of the country most common ethnic group is present, usually called Punjabis in Pakistan, those are basically Indo-Aryan race. Afghanistan is present in the northwest region Afghanistan and province bordering has Pashtun people. The race is derived from Jewish and Greeks. Look more like Europeans [289]. Migration pattern making the ancestry of the region is displayed in Figure 82.

These people spread down to north of the Baluchistan. Iran and Pakistan share an exceptionally long border in the west of the country. People with Persian roots in some parts of the Baluchistan province. Brahui people are another ethnic group living in Baluchistan and Sindh [290]. Another type of people living in Pakistan are of the local roots specifically from south of the Indian subcontinent. These are usually belonging to Dalit group of India [291]. Makrani people living in coastal areas of Baluchistan are descendants of Africans. They migrated as merchants or servants [292].

Ethnic Group	Location ^a	Language Family	Census Size ^b	Suggested Origin(s)
Baluch	Baluchistan	Indo-European	4,000,000	Syria: Aleppo (Quddus 1990)
Balti	Eastern Baltistan	Sino-Tibetan	300,000	Tibet (Backstrom 1992)
Brahui	Baluchistan: Kalat State, Sarawan and Jhalawan regions	Dravidian	1,500,000	West Asia (Hughes-Buller 1991)
Burusho	Karakorum Mountains: Hunza, Nagar, and Yasin	Language isolate	50,000–60,000	Alexander's army (Biddulph 1977)
Hazara	Southern Baluchistan: Quetta and NWFP: Parachinar	Indo-European	NA	Genghis Khan's soldiers (Bellew 1979)
Kalash	NWFP: Hindu Kush Mountains: Bumburet, Rambur, and Birir valleys	Indo-European	3,000–6,000	Greeks (Robertson 1896) or "Tsyam," possibly Syria (Decker 1992)
Kashmiri	Kashmir Valley	Indo-European	NA	Jewish (Ahmad 1952)
Makrani Baluch	Makran coast	Indo-European	NA	West Asia (Hughes-Buller 1991)
Negroid Makrani	Makran coast	Indo-European	NA	Rajput (Quddus 1990); Africa?
Parsi	Karachi	Indo-European	A few thousand	Iran, via India (Nanavutty 1997)
Pathan	NWFP and Baluchistan	Indo-European	17,000,000	Jewish (Ahmad 1952), Greek or Rajput (Bellew 1998; Caroe 1958)
Sindhi	Sindh	Indo-European	15,300,000	Mixed (Burton 1851)

^a NWFP = North West Frontier Province.

^b NA = not available.

Figure 83 Study of Pakistani Ethnic Groups [Source: Qamar et. al., 2002] [289]

Out of all the above mentioned major ethnic groups living in Pakistan in large numbers as shown in Figure 83, we can classify four major face types having distinct face styles. Type I is for Balti and Hazara people. Type II faces are for Pashtun and Baloch people. Type III combines Punjabi and Brahui people. Whereas Type IV

represents the Makrani and South Indian race. In Pakistan, still cultural barriers are present, and it is rare that people marry out of their ethnic group. So, the facial features remain very distinctive among these groups. These are the typical types of the faces while reporting or filing any police sketch classified as Type I-IV in Figure 84.

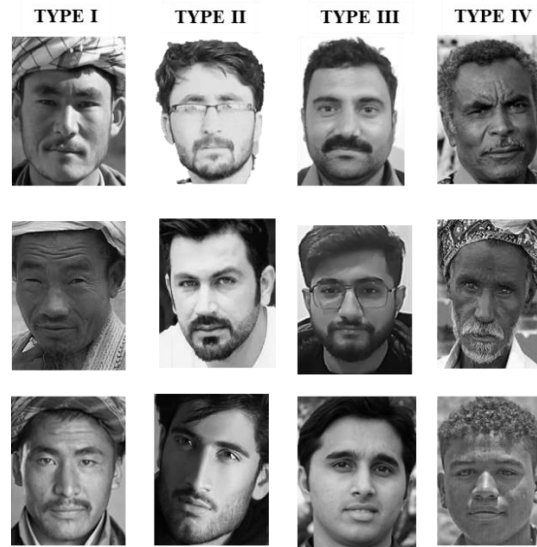


Figure 84 Face Types Samples

Various works has been done in the race classification. Gao et. al. [293] using PCA and SVM, Khan K et. al. [294] using Facial feature discovery framework, Salah et. al. [295] using Gabor and SVM and Roomi et. al. [296] using Regression created models trained on FERET and CAS-PEAL datasets for classification of Asians and Non-Asian faces. Antonio et. al. [297] using DCNN and Khan K et.al. using Facial feature discovery framework trained model on VNFACE dataset to classify Vietnamese and others. They achieved accuracies of 88.8% and 92% respectively. Models are trained on FERET and CAS-PEAL datasets to classify Asian, Caucasian, and African races are developed by Yiting et. al. [298] using Skin information and Adaboost, Chen et. al. [299] using DCNN, Geng et. al. [300] using Gradient and Gabor pattern and Khan K et. al. [294] using Facial feature discovery framework. Sarfaraz et. al. [301], Qiong et. Al. [302] and Khan K et. al. [294] trained model on FERET and VMER datasets to classify African American, Asian Indian, Caucasian Latin and East Asian races and achieved accuracies from 93% to 96%.

The last stage for the classification is the facial hair styles. Ubaid et. al. [303] presents an advanced system for detecting and segmenting beards and hair in

challenging facial images, which is crucial for tasks like gender identification, age assessment, and facial recognition. The proposed system also allows for changing the colour of segmented hair and beards. The authors utilized a modified version of the Mask R-CNN model for this purpose and prepared a dataset of 1500 images, evenly distributed between hair and beard images, which is publicly available on the NCAI1 website. Through transfer learning on this dataset, the modified Mask R-CNN achieved an impressive accuracy of 91.2%, outperforming previous systems designed for the same task. This research significantly advances the state-of-the-art in beard and hair detection and segmentation [303]. Now the challenge was to further classify the people based on various facial hair styles. No such work has been done in this field before. In Pakistan men grows their beard. Pakistan has Muslim population of 97%. Growing beard is one important part of Muslim community [304]. People from various sects follow different type of beard having lengths and styles [305] But this practice is not only driven by the religion. It is also due to cultural influences and fashion. Facial hairs not only include the beard but the moustaches as well. It is little difficult and new to understand for the reader from other parts of the world. They are categorized by the length, starting limit of the beard, spread and presence of the moustaches. For any of the sketch maker it is vital area of the identification of a person. Type I is for the faces with no facial hairs at all, Type II is for the faces having moustaches only. Type III contains faces with long beard and Type IV represents faces with short beard covering jaw line only. Figure 85 shows samples for each facial hair type.



Figure 85 Facial Hair Types

Detailed comparison of these models on gender, face type and facial hair classification is shown in results section. All these processes have been done on grayscale images keeping the size constraints in real-time process. Moreover, no classification was made based on colour tone. One more reason is that if colours were considered then the image captured by night vision camera cannot pass the criteria.

MATLAB 2020a and python with TensorFlow, Keras etc. were used to perform training. RTX 2060, GTX 950 and Google Colab were used for training the large dataset to gain maximum accuracies.

II. Image Collection

For development of any of the deep learning model and recognition system the most important thing is development of correct dataset. This dataset should be rich enough to classify the input data. This research has been done for the region of Pakistan and focused only type of people living in this country. No similar work has been done in this area for this region. Being research with new dimension, new dataset was required to be built. That is why for all the stages datasets had to be prepared. This is the most tedious and time-consuming part of the research. Following steps participated in the preparation of the datasets.

There are three types of layers for recognition. Open-source gender recognition dataset was available. But for more accurate results for real time scenario the dataset was transformed by the pre-processing stages to make the recognition efficient in test environments. Open-source images from *Cleb_A*, *Wider Face*, *CUHK Person Re-identification Datasets*, *Kaggle Uzbek People Faces*, *YFCC-100M Flickr dataset* and open-source *Google search images* containing relevant images were gathered. In addition to these images, more images are captured in the premises of university. Total 10000 images per gender were prepared from which 50% data was used for training and 50% for testing. Then the images of four types of ethnicities or face types were collected. It was a hard job to find open-source images for Pakistan region. Major local ethnicities are descendants of ethnicities like Persians, Asians, Europeans, South Indians, North Indians Africans etc. Images of the people belonging from the root of the face types are also added to provide the characteristic features of the type to the respective model. For example, Central Asian face images are added to define the Face

Type I, European faces are utilized in addition to Pathans for Face Type II, Face III includes typical North Indian faces those live in Punjab region and East African and South Indian face images are used to define Face Type IV. In addition to the open source google images, dataset was generated by gathering the images from the university students. University of the Punjab is the biggest university of the Pakistan having thousands of students enrolled on campus. People from all over the country studies in the university. Moreover, images were collected from University of Central Punjab with the same procedure. Cell phone with [4:3] 13MP resolution was used for collection of images. As described in the methodology section student images were acquired after their filling the consent form and no picture has been published without their approval. Similarly, images of subjects including all types of facial hair styles were obtained from Google image gallery and university student's pictures. Finally, it was possible to gather 4000 images for each category of face and 4000 for facial hair from all the sources.

III. Dataset Population

It was not possible to populate the dataset manually with the thousands of the images manually. Therefore, an efficient algorithm fulfilling system requirements was created. Following steps shown in the figure were followed.

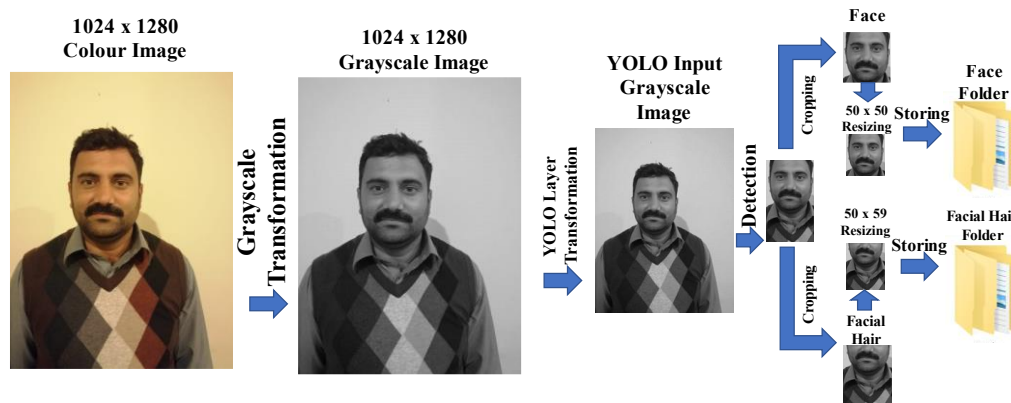


Figure 86 Dataset Generation Process Flow

All the pictures acquired, contained backgrounds whether those are from online sources or the direct acquisition and had different dimensions. As shown in the Figure 86 the dataset images are passed through few steps before using for model training. First step is to convert the colour image to grayscale. The input image may have

different dimensions. These images are fed to YOLO v3 trained model, and its dimensions are changed as per the input layer parameters. The images acquired are focused on the subject's face, so no further enhancement was performed before face detection except the grayscale conversion. YOLO v3 model detects the face, it is cropped with the same way as mentioned in the Face Cropping section. Two parts of the images are selected from the cropped image. One is for the face dataset and the other is for the facial hair type dataset. Images are downgraded to 50 x 50 for face and 50 x 59 for facial hair type, to reduce pixel density before training the system to make system robust against the worst scenario that is targets presents at the maximum range that is 50 feet. These images are then again scaled to 224 x 224 x 3 by the training algorithm before passing it to the input layer of the network.

(A) Image Augmentation

Larger the size of the of images in the dataset, more accurate will be the model. The larger dataset will produce better result than model trained on less dataset. For increasing the number of the images in our dataset “Image Augmentation” was used. Image augmentation algorithm add distinct image to our dataset so that our data set become larger. All images are different as some of them are rotated by small degree (0-5 degree) as shown in Figure 87. As alignment is already performed so the rotation is not more than 5 degrees. The final image is cropped from the inner sides and gives the final image of the central face.



Figure 87 Image Rotation at 5 degrees

Some images were mirrored as shown in Figure 88.



Figure 88 Image Mirroring

(B) Storing images

Last step was to store the images with a unique name. Algorithm was run for each category and named the images as of iteration number. All the images were stored in the .jpg format. Figure 89 shows the final face pictures produced. These pictures are intentionally blurred to keep the privacy of the people volunteered in the dataset development procedure.



Figure 89 Dataset Images Output

4.05 Face Categorization System

This system sorts the images into the gender, face type and facial hair type as shown in the Figure 90.

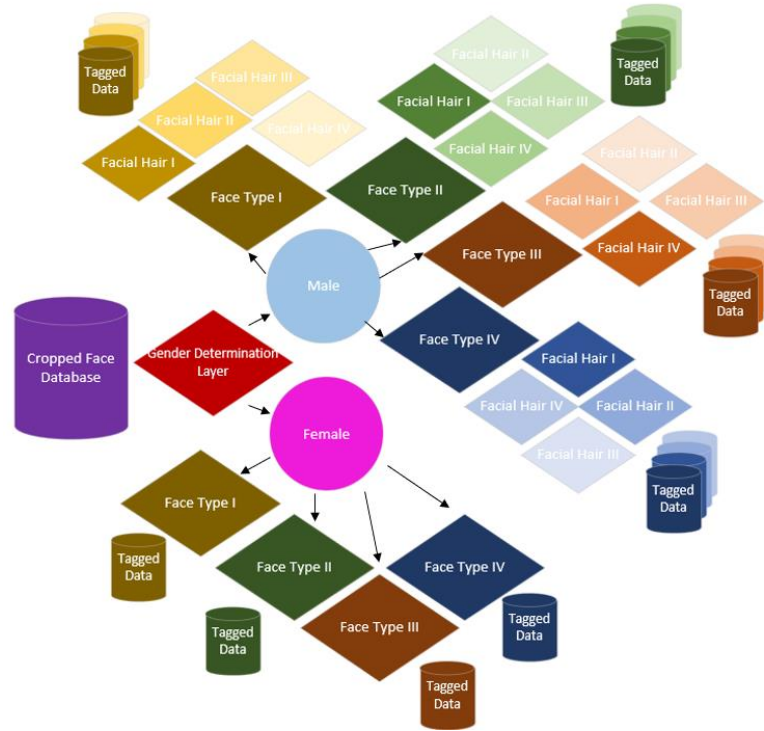


Figure 90 Person Categorization System High Level Flow

I. Frames Placement

The first step is performed by Raspberry Pi, s by creating folder for each camera added to the node. Each camera has its node number, longitude and latitude that is available in the lookup table in .csv format and accessible from the cloud. Each camera folder has subfolders representing each day of the year. That can be further sub divided in months but in this research, it is on daily basis. In the day folder there are twenty-four folders representing each hour of the day. The total number of folders in each camera is 365 x 24 per year. Initially only one year is accommodated. Figure 91 shows the folder distribution in the frame folder. More years can be added in a similar manner.

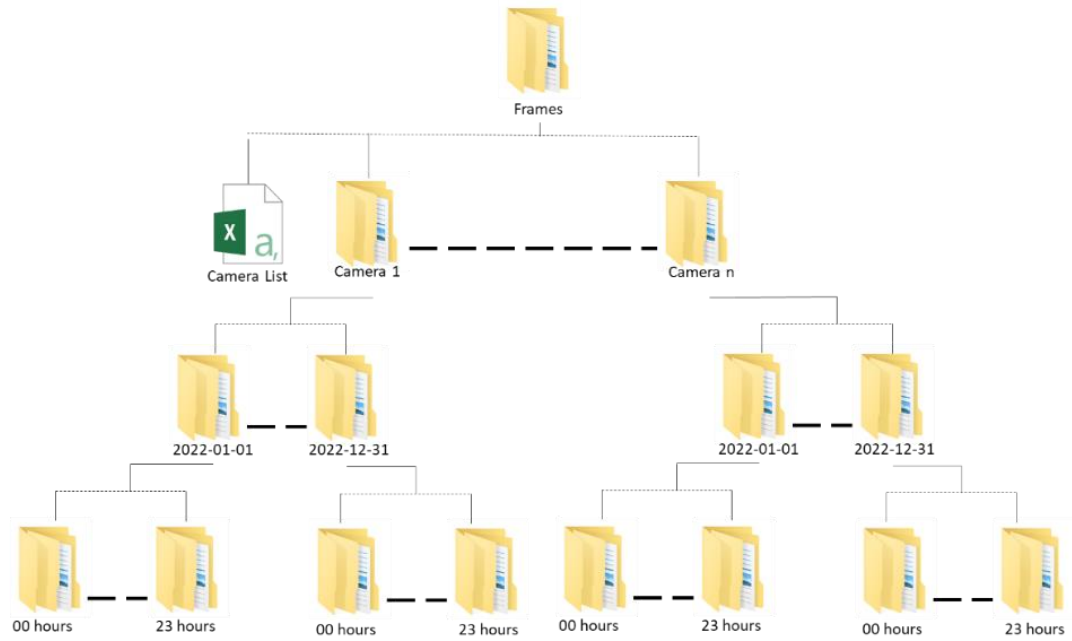


Figure 91 Frame Folder Distribution

As discussed in the earlier section the frames are captured when there is a trigger signal from PIR. These frames are downsized and converted to grayscale and stored into relevant folders available at cloud. Multiple processing nodes work together to populate the data.

II. Face Placement

There is the same structure, as of the frames in the *Face_Data* folder. It contains the same folder names but this time the computing system(s) writes the cropped faces from the corresponding frame folder. Nomenclature for the face images is already discussed in the previous section. Figure 92 shows the upper tier of the folder distribution for the cropped faces.

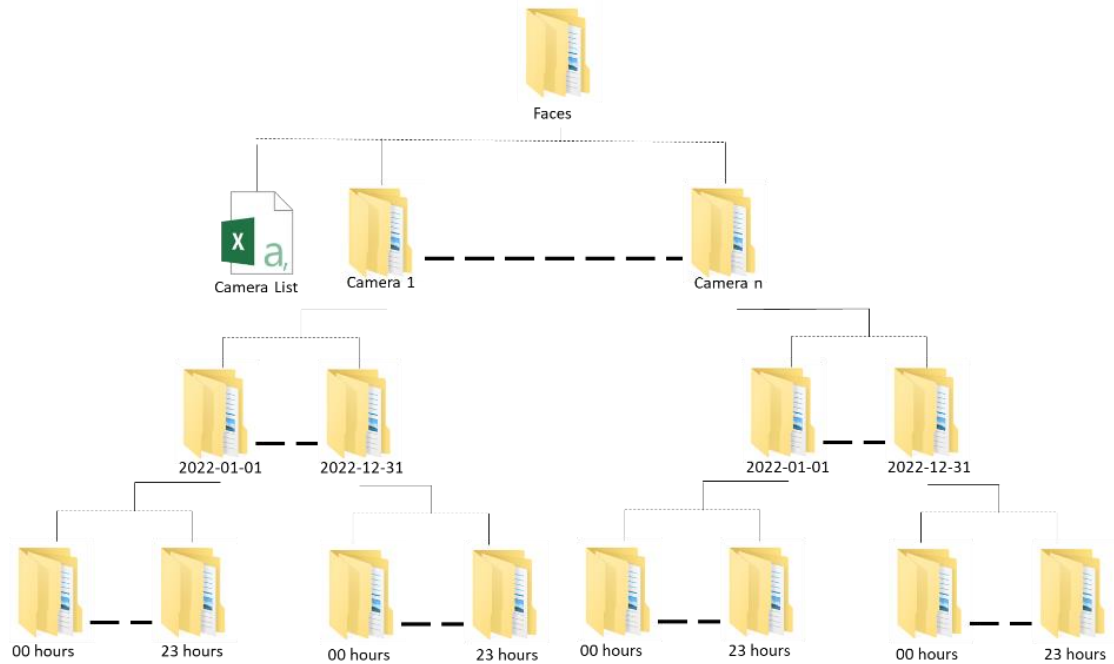


Figure 92 Face Folder Distribution

Once these folders are created, the recognition models start working. Now the system manages to reach each image from the subfolder of each camera, then these images are passed through the classification layers. In this part feature extraction is done for the upper matrix that contains face. Extracted features are saved against each image. As shown in the results section the VGG Face was selected as the final model due to its promising results. Once the gender and face types are evaluated for each image, lower facial hair matrix of the males are then extracted. This part is then evaluated for the facial hair type.

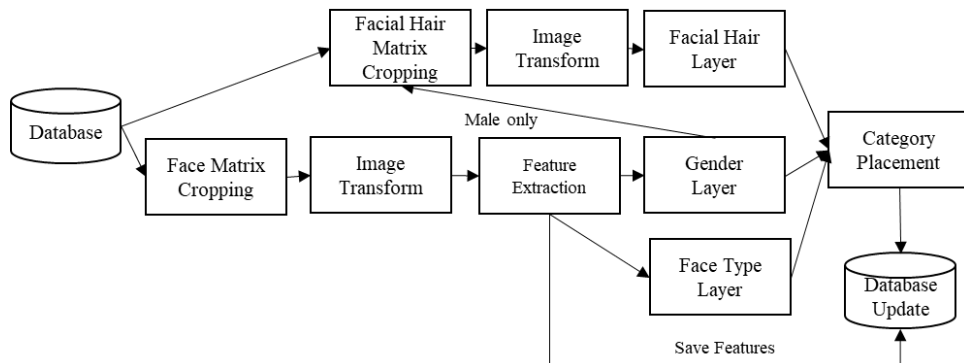


Figure 93 Categorization System Low Level Flow

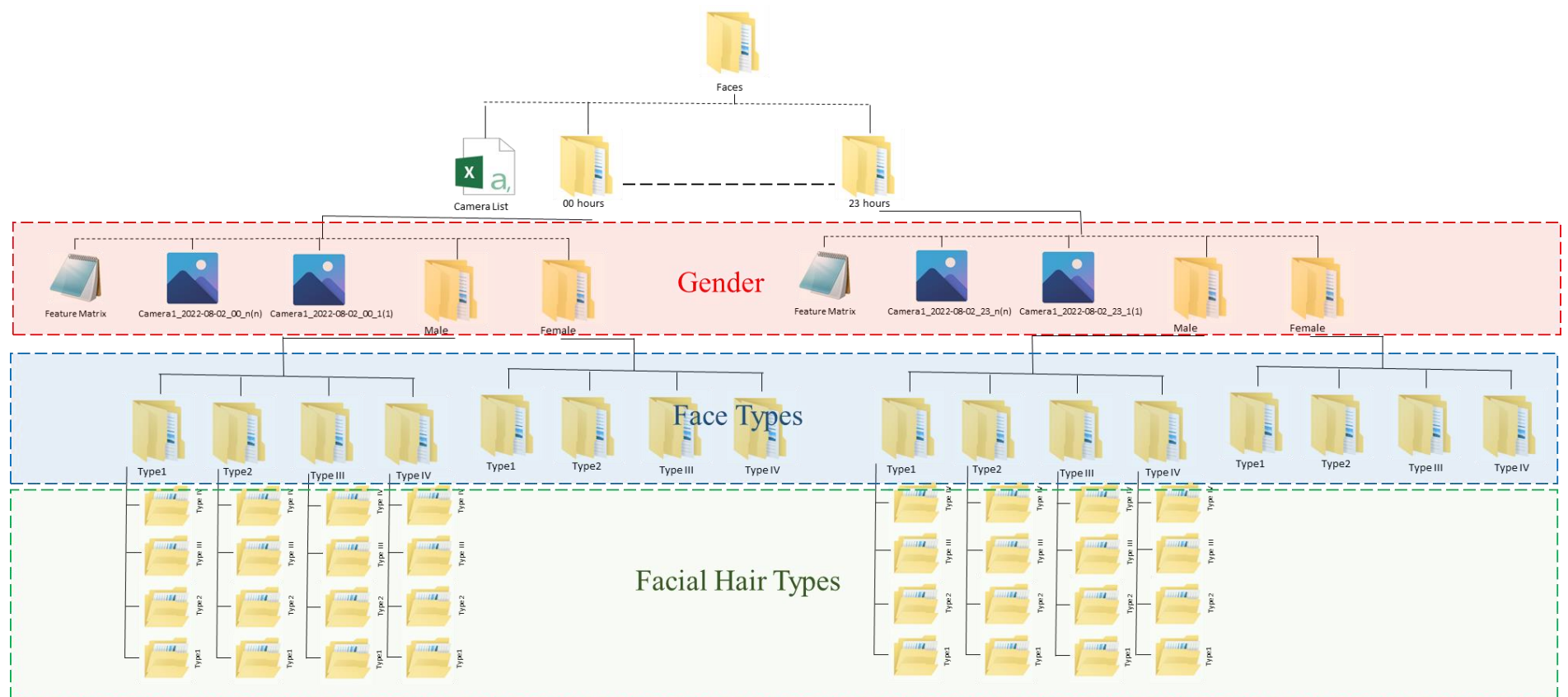


Figure 94 Classification Folders Scheme

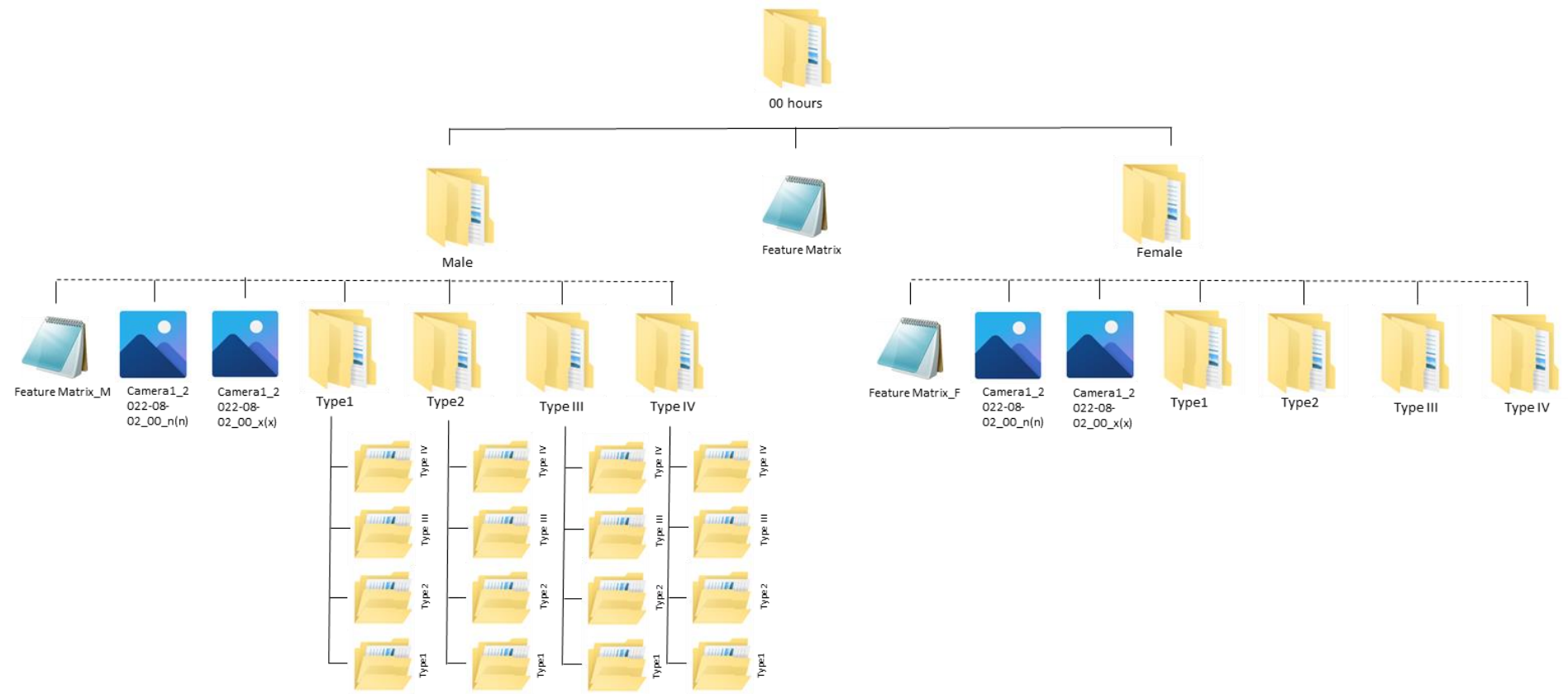


Figure 95 Classification Folders Scheme till Gender Classification

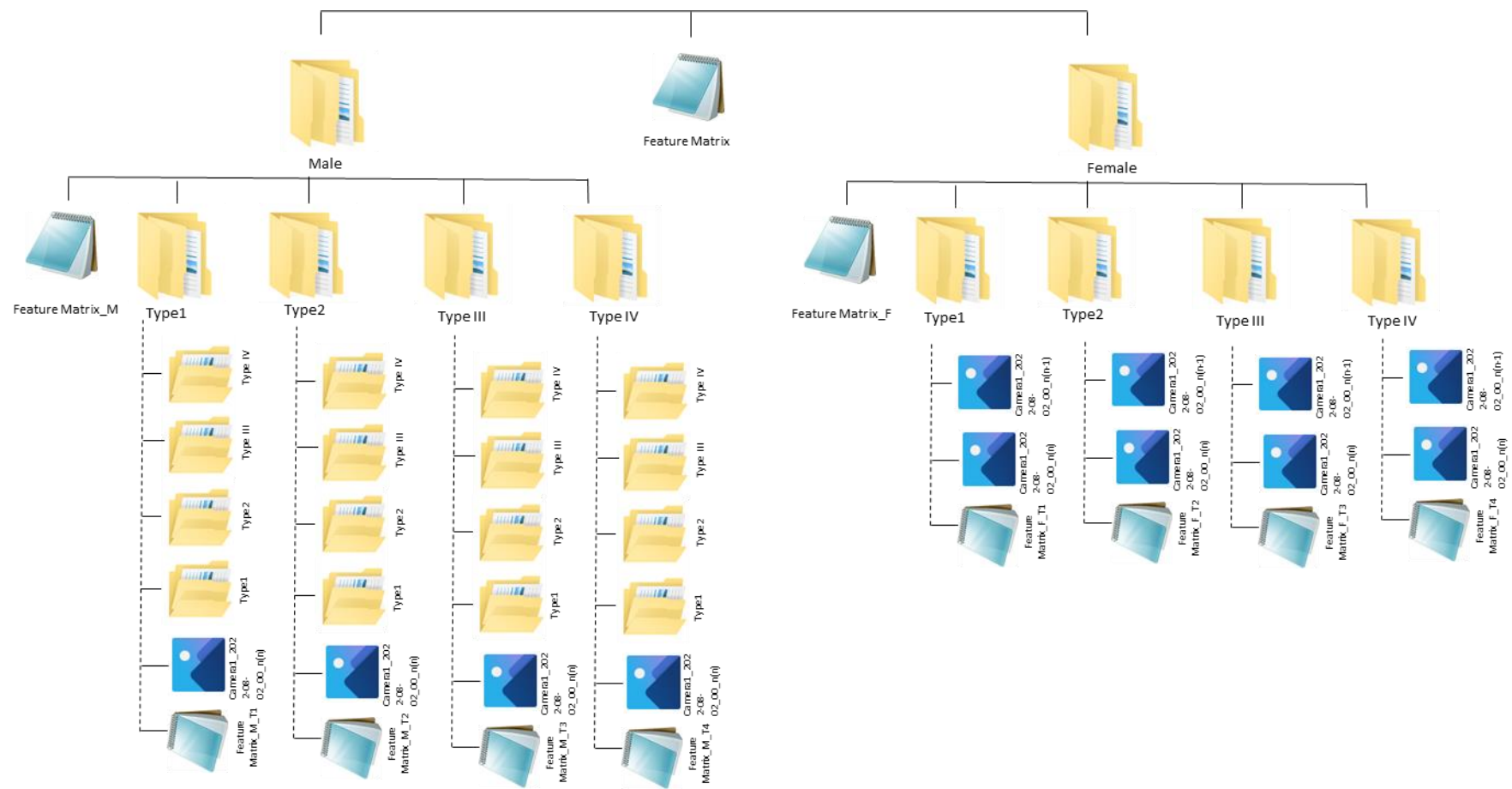


Figure 96 Classification Folders Scheme till Face Type Classification

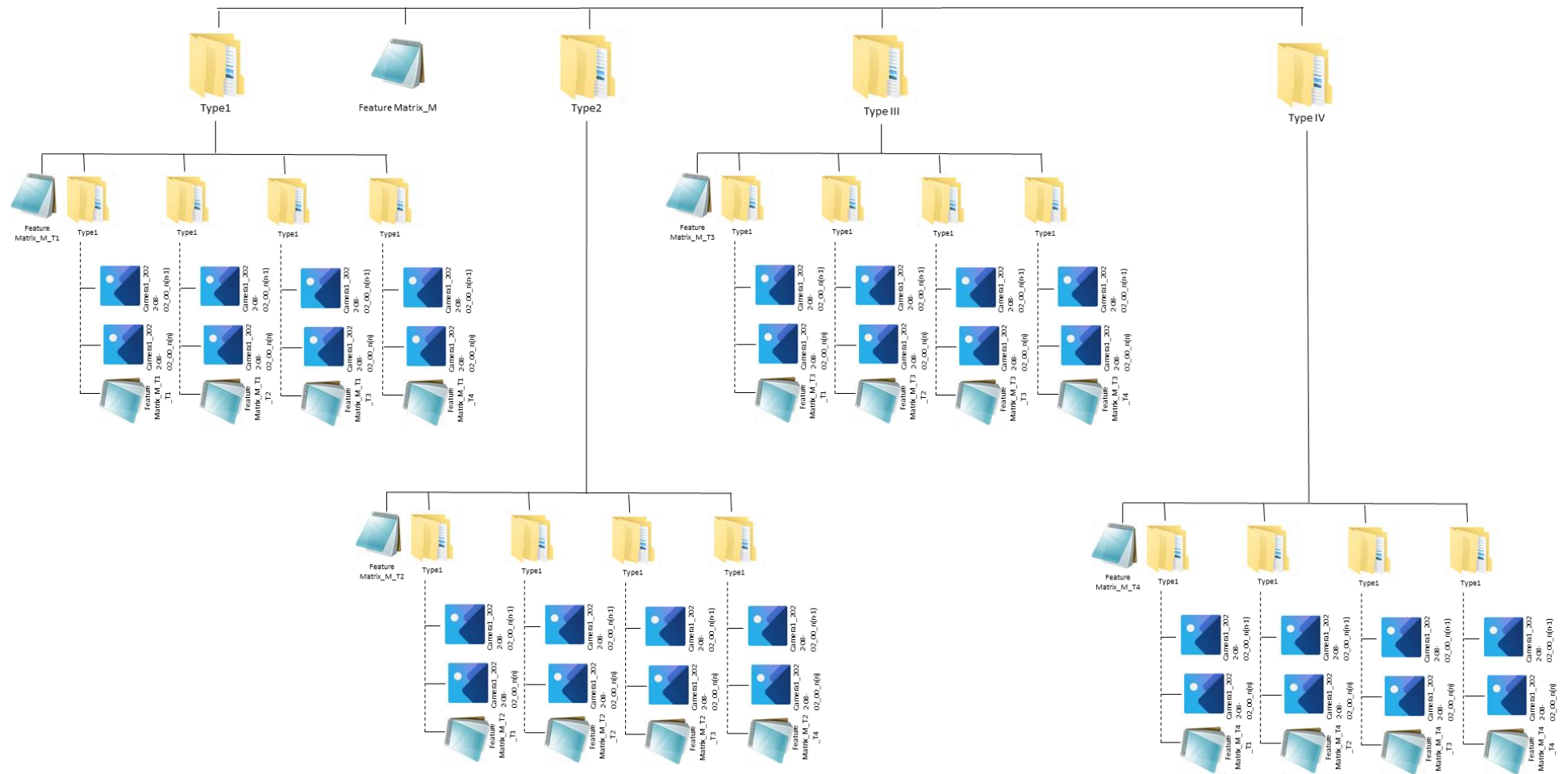


Figure 97 Classification Folders Scheme till Facial Hair Type Classification

Each category of the face needs to be stored in a designated folder. This work is done after the processing stage. As discussed in the earlier section this work is a prototype and includes only one main computing device. The proposed system proposes parallel devices working together and dealing with their designated areas. They can efficiently handle the images of their designated regions. Each image is needed to pass from the cascaded stages of the system. Figure 93 shows the flow of the system.

After passing through stage I, images having dimensions of 224 x 365 are available in grayscale. These images contain two types of data, one is the upper matrix having dimensions of 224 x 224 pixels containing the face details and the other part of 224 x 263 represents facial hairs from mid nose point to additional 63% pixels downwards.

First step is to extract the upper matrix and resize to the VGG network input size that is 224 x 224. Till this point the images are in grayscale, to meet the criteria for the input, image is converted to 224 x 224 x 3.

Then the image is ready for feature extraction. At this stage total 4096 features are extracted and stored in the matrix against the name of the image.

Extracted features are stored in the time folder for face matrix. This file contains the file name and the 4096 columns for the features. Now all the cropped faces are present in the respective time folder of each camera. Multiple computing devices can access the folder and work simultaneously on the files. Figure 94 shows the images placement till the feature extraction step.

III. Gender Classification

After this step the models are applied on the images one by one. Firstly, the gender model uses the features matrix and classify the images into male and female. These images are moved from the parent folder leaving it with only gender folders and feature file. A new feature matrix file is created in each gender folder copying only the relevant features of the class. Figure 95 shows the image placement folder organization after completion of the gender classification. One thing is to be noted here that as multiple processing devices can work together so the images are moved to next tier so only one

device would be working on one image. Features are moved from one file to the other file in the sub class.

IV. Face Type Classification

Till the above step, images are already in their respective gender folders along with the features matrices. The next step is to classify each gender image into its respective face type location. A new feature matrix file is created and that copies the features of the relevant type only. Face type folder management is shown in Figure 96.

The second layer works on the images in both locations separately. Face types are now detected using the same extracted features. It should be noticed here that each face has confidence level for each type. Face type having highest percentage of matching is selected. If a face has multiple face types with $\pm 15\%$ matching value of maximum is copied to the multiple face type folders. As per the experimental results it is a rare condition but needs to be taken care of. As if the query image may not have the same results. That is why this threshold has been set for the confidence level. This process is shown in the Figure 98.

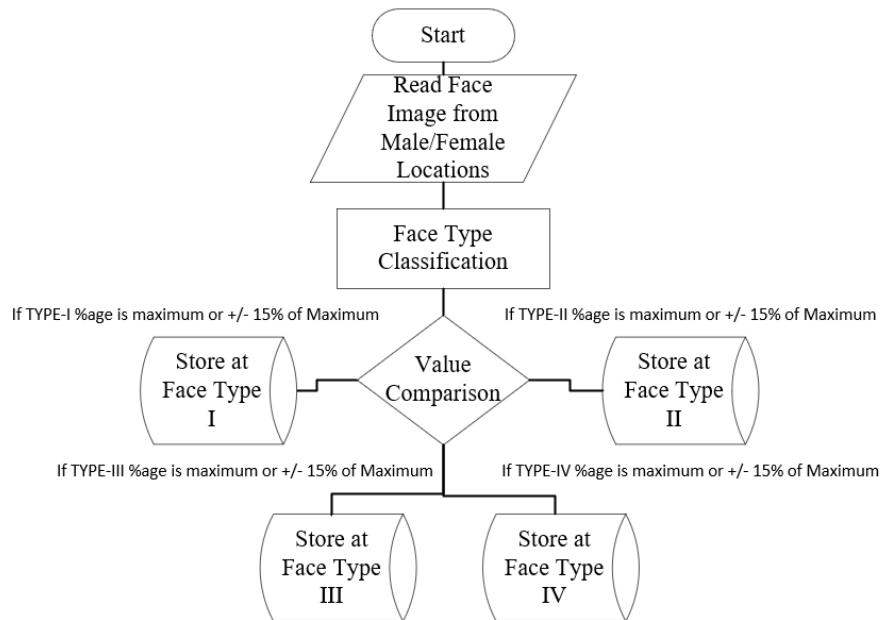


Figure 98 Face Type Classification

V. Facial Hair Type Classification

All the categorization in the female category is completed till this point. Whereas for the male face types the next layer to be evaluated is the facial hair styles. Facial hair matrix of the male classes is cropped. It crops the lower 224 x 263 pixels of the stored image. This part is then rescaled to ResNet-50 input layer criteria. Then the relevant category is predicted. Facial hair type folder management (only for male) is shown in Figure 97.

As discussed earlier four facial types are trained in four categories. Similarly, the facial hair length is also a tricky part. Same person can be placed in two categories of facial hair length as per the angle and illumination conditions. That is why face having +/- 20% margin to the neighbouring category is placed in two categories. Figure 99 shows the classification of the images in the categories.

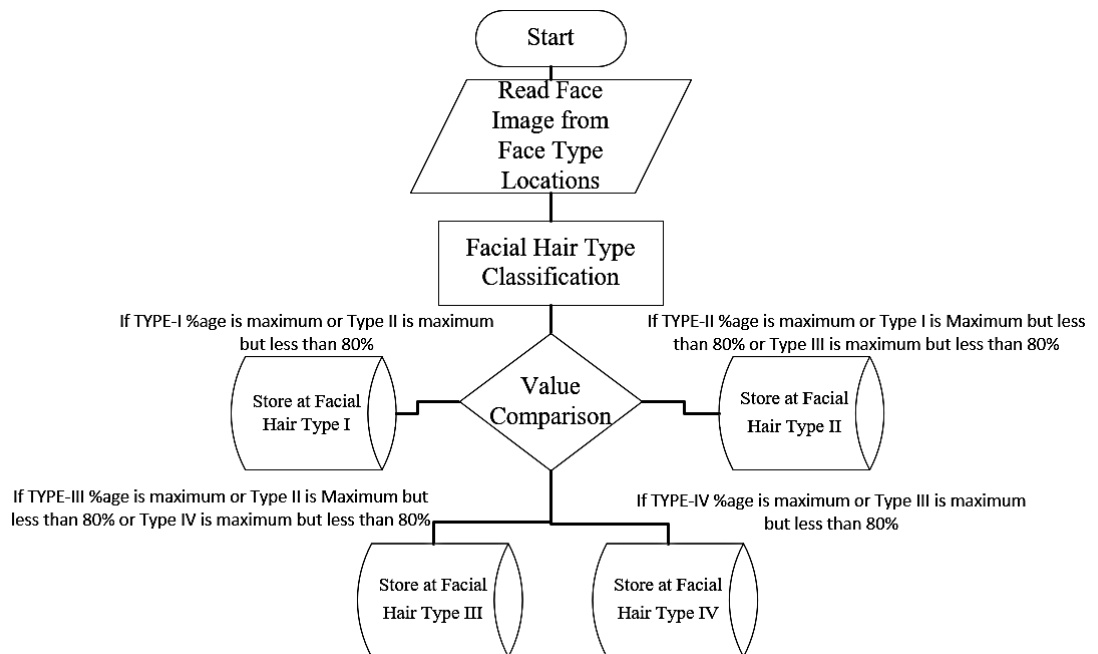


Figure 99 Facial Hair Classification

4.06 Target Search System

This stage gives user the access to the database to find the target from the cloud as shown in Figure 30. This system offers multiple options to user as earlier discussed in the Figure 31, Software Architecture Diagram. This is the final face recognition and tracking system as shown in Figure 100. Here the face image for tracking purposes is given to the system. Images can be of different types available with the law enforcement agencies. Basically, there are two types of the inputs taken care of. One is the colour or grayscale camera image, and the other can be sketch drawn by the forensic artists.

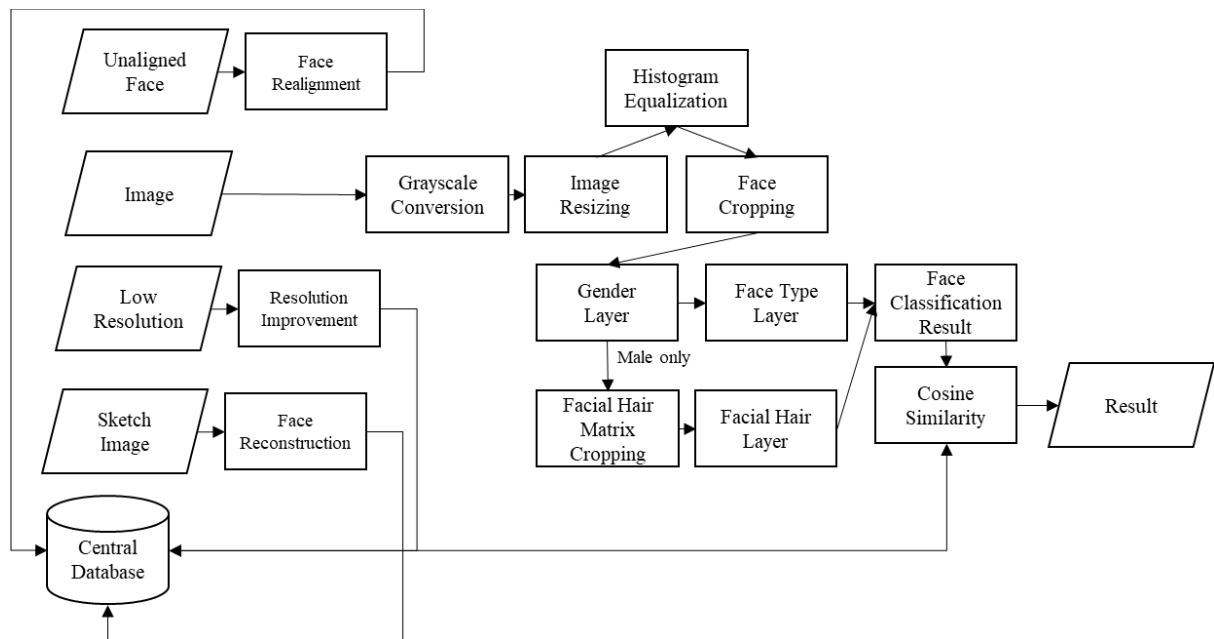


Figure 100 Prediction System

I. Database Image Search

Being an intelligent assistance system, this system helps the user to find the target in the database. As discussed earlier the faces those are being cropped from the frames sent by the network of cameras are already stored in their respective locations. As it is shown in the Figure 100, following steps are involved while finding the target from the database.

Cropped face that is available with the user, or it can be cropped from any of the frame or footage available is passed to the system or from any other database. This image is transformed into grayscale and dimensional compatibility is made for recognition.

This is further processed to increase the efficiency of the system. This processing includes sharpening of the image and evaluation of histogram. At this stage image may be too dark or too bright. This is checked by shift of the histogram. If it is darker then the distribution of the grey levels will be at lower side, and they will be distributed at upper side for a brighter image. It is equalized as per our sample image histogram. Histogram equalization is performed to attain better results.

It can be difficult and time consuming to find ID among all the databases. To avoid such conditions, database is divided into categories, Location, Date, Time, Gender, Face Type and Facial hair. Main folder is represented by the Camera ID in similar way as Digital Video Recorder (DVR)/ Network Video Recorder (NVR) does. In this folder daily data of the month is separated, then comes the hourly data and in this hourly data each hour folder contains the face classes. Data from each camera is placed in a specific location categorized by the camera ID/ location, Time of the event in hourly manner, Date on daily basis and face number of each frame, gender, face type and facial hair style in each category.

At this step, image has dimension of 224 x 365 and feature extraction has already been done while passing through the gender and face type categories layers. Whereas for the facial hair type the dimension is 224 x 263. Now, the input image is passed through the stages to categorize it. Images are tested for each layer and assigned a weight for each match result.

Its features are extracted and stored in an array. Feature extraction is a costly process that runs only once for the face here. Similarly, the facial hairs are also evaluated but only for the categorizing, it's not used for matching the database.

User selects the initial search by selecting the initial point (Camera ID), date and time of the event. Further categories are identified by the system already by passing the input image through the categorization layers.

At last, the face is further matched with the IDs with discovered characteristics. Features extraction, cosine similarity is used for matching the input image features

with the features of the images present in the respective category having the highest matching score (confidence level).

Matching two images using features extracted by the VGG Face model is a broadly used method in the field of facial recognition. The VGG Face based model trained on a large dataset of facial images and extracts 4096 features from facial images. The features extracted by the VGG Face model are used to compare and match two facial images using cosine similarity. It calculates the similarity between two non-zero vectors. In facial recognition, the cosine similarity of two feature vectors is already extracted by the VGG Face model. The cosine similarity score ranges from zero to one, where zero indicates no similarity, and one shows perfect similarity. Cosine similarity between the two feature vectors is calculated taking dot product of the two feature vectors and dividing it by the product of their magnitudes as given in Eqn. 4-139.

$$\text{Similarity}(I, D) = \frac{I \cdot D}{||I|| \times ||D||}$$

Eqn. 4-139

Or using Eqn. 4-140.

$$\text{Similarity}(I, D) = \frac{\sum_{i=1}^n I_i \times D_i}{\sqrt{\sum_{i=1}^n (I_i)^2} \times \sqrt{\sum_{i=1}^n (D_i)^2}}$$

Eqn. 4-140

Where I , is the input image features and D represents database image features.

Compare the cosine similarity score to a threshold. If score is above the threshold, the two facial images are considered a match. Facial characteristics and cosine similarity already have significant uses, such as recognizing faces in surveillance video or matching faces in a database of known people. This method can also be used to recognise people in a variety of photos, such as those in social media profiles, or to spot forgeries and fraudulent behaviour.

It is crucial to keep in mind that this method has its drawbacks, including poor performance in low-quality photographs, fluctuations in stance and lighting, and variances in face emotions. Facial recognition has also come under fire for having the capacity to violate privacy and maintain bias and discrimination. Thus, it is crucial to

apply this technology with caution and to put in place the necessary safeguards to allay any potential moral and legal issues.

II. Sketch Image Search¹

In real world scenarios, it happens that crimes are reported in such a way that culprit is unknown to the victim and there is no video evidence to get the image of the criminal. In such cases, sketches are made by the investigation team with the help of victim(s) or witnesses. It is difficult for the humans to match the sketch face with the real face. It becomes almost impossible for the computer vision systems to recognize such targets. In this regard a system is created using MS-Cycle GAN to estimate a colour image of person from his sketch.

Multi-stage Cycle generative adversarial network is employed as a pre-processing phase to transform drawing into real-time face. Multi-stage Cycle networks are more precise than Cycle generative networks and highly dependable when transforming image data from one source to another, but they are more demanding in terms of data volume. Contrary to Cycle-GAN, multi-stage Cycle networks require numerous generators and discriminators, which contributes to their high processing time. The goal is to transform a drawn image into a hyper-real image using the GANs model. This led to the creation of a new generative adversarial network that can outperform Cycle-GANs and simple GANs in terms of matching rate. The proposed MS-CGAN network consists of numerous layers that each contain a different generating and adversarial network. One discriminator and two generators make up each layer. One generator, GA2B, is tasked with creating an image from the training dataset, and the second generator, GB2A, converts the created picture back to the training dataset image so that we can calculate the cyclic consistency loss as described in Cycle-GANs. The picture generated by generator GA2B will proceed to the second layer and become the input of the second layer generator GB2C, and the process will repeat itself, if the discriminator produces output greater than the threshold (let's assume $t > 0.5$). However, if the first layer discriminator's output is below the threshold, layer one's creation process will continue until GA2B creates a picture that

¹ "This work is in publication process as "High-Definition Image Formation using Multi-Stage Cycle Generative Adversarial Network with Applications in Image Forensic" in Arabian Journal for Science and Engineering, Ref.: Ms. No. AJSE-D-22-06413, cleared the reviewers after the revisions."

can deceive the discriminator. The procedure of the suggested MS-GANs is thoroughly explained in the figure. The output of the generator is supplied to the second layer of each layer, which has two generators and one discriminator. A two-layer GAN model is shown in the image below. It can be raised to different levels. Although training will cost more, the results will be better.

Details of the process is described in Figure 101. This system has only two layers of GAN model that can be expanded to multiple levels. This expansion will increase the training and execution cost. But results can be improved.

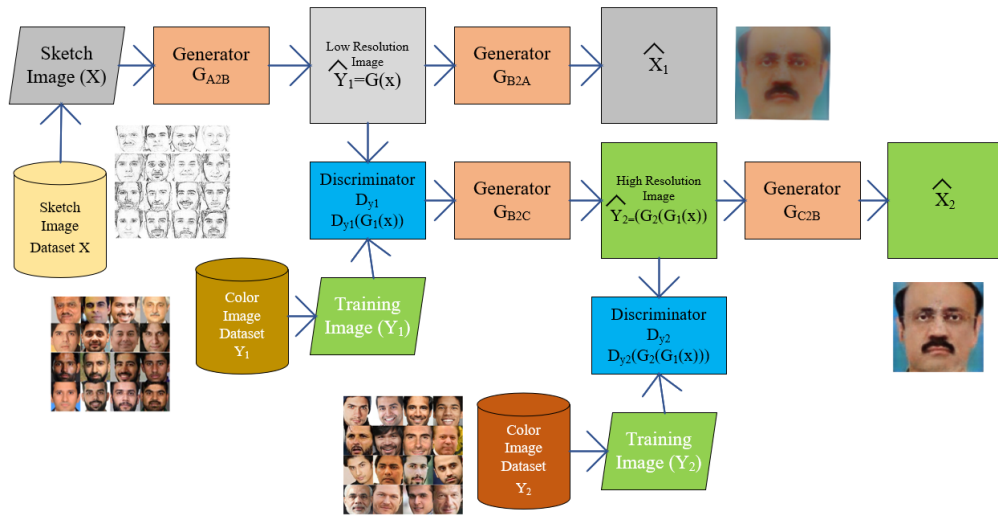


Figure 101 Process of Proposed MS-CGAN

(A) Mathematical Modelling of Network

In the MS-Cycle GAN architecture, there are two discriminators and four generators. With the aid of GB2A and GC2B generators, generated images are changed back to the original image. Finding the loss function that refers to the cyclic consistency loss is helpful. Below is a calculation of the estimated loss function. Loss function for a multi-layer GANs network can be computed independently. By submitting both losses at layers, the total loss is calculated.

Layer 1

Layer 1 can be written as Eqn. 4-141.

$$L_{AL1}(G_{A2B}, D_{Y1}, X, Y_1) = E_{Y_1 \sim p_{data}(Y_1)} [\log D_{Y1}(Y_1)] + E_{x \sim p_{data}(x)} [\log (1 - D_{Y1}(G_{A2B}(X)))]$$

Eqn. 4-141

Adversarial loss at layer 1 is shown as L_{AL1} whereas training data is represented by X , Layer 1 discriminator is represented by D_{Y1}

$$L_{CL1}(G_{A2B}, G_{B2A}) = E_{x \sim p(data)} x [||G_{B2A}(G_{A2B}(x) - x)||] \quad \text{Eqn. 4-142}$$

L_{CL1} shows cyclic consistency loss and calculated using Eqn. 4-142.

Layer 2:

Discriminator D_{Y1} needs a value of threshold to be fixed. This value is a hyper parameter and can be adjusted as per the generator and training data. Here value of threshold is set to 0.5. Real and adversarial loss function G_{A2B} at layer 2 is given by Eqn. 4-143.

$$\begin{aligned} L_{AL2}(G_{B2L1}, D_{Y2}, G_{A2B}(x), Y_2) \\ = E_{y2 \sim p(data)(y2)} [\log D_{Y2}(Y_2)] \\ + E_{G_{A2B} \sim Pdata(G_{A2B})} [\log (1 - D_{Y2}(G_{B2C}(G_{A2B})))] \end{aligned} \quad \text{Eqn. 4-143}$$

Like first layer adversarial loss is given by L_{AL2} at layer 2 and GANs loss is same. Cycle consistency loss is the second loss given in Eqn. 4-144.

$$L_{CL2}(G_{B2C}, G_{B2C}) = E_{G_{A2B}(x) \sim pdata} G_{A2B}(x) [||G_{C2B}(G_{C2B}(x)) - G_{A2B}(x)||] \quad \text{Eqn. 4-144}$$

Total adversarial loss in these layers is given by Eqn. 4-145.

$$L_{AL1+AL2} = L_{AL1}(G_{A2B}, D_{Y1}, X, Y_1) + L_{AL2}(G_{B2C}, D_{Y2}, G_{A2B}(x), Y_2) \quad \text{Eqn. 4-145}$$

Total consistency loss is given by Eqn. 4-146.

$$L_{CL1+CL2} = L_{CL1}(G_{A2B}, G_{B2A}) + L_{CL2}(G_{B2C} + G_{C2B}) \quad \text{Eqn. 4-146}$$

(B) Training Dataset

The dataset and network training are the most crucial steps in the process for any kind of neural network. The output of a neural network depends on the dataset's usefulness, the hyperparameters that are set, and the architecture that is employed. The dataset for this block is prepared with the same level of care and accuracy as the other phases. This dataset contrasts computer-generated sketches with real colour images rather than actual hand-drawn sketches. Due to financial limitations, it can only use

photographs that were generated by the system. The accuracy was tested using a dataset comparing real face photos to Asian sketches. But as was mentioned in the earlier section, it needs to be distinct for each sort of face. The method was tested on a dataset of 1400 images.

Results, Layer settings, and training data are displayed in the results section. Data from sources other than the dataset were used for training in accordance with the algorithm's testing rules. The technique was run on a set of sketch faces to produce the expected results. The grayscale sketch pictures were used to create RGB images.

III. Low Resolution to High Resolution Image Estimation

In real world scenarios, images acquired from CCTV cameras at various sites does contain many details. That can lead to a problem; of not getting enough details that can help identifying the target face. For designed system, human face image is required that have enough features to get recognized. This system gives one additional feature, that a low-resolution image can be used to estimate a high-resolution image.

Being an artificial assistance system, which can be used by monitoring team as well as for the automated classification system. It helps to increase the PPF of the input image (low resolution). MS Cycle GAN has already been used for photo-enhancement in many applications. Low resolution and the images with blur due to bad focus or motion can be handled with this technique. In the motion pictures blur happens and can cause discomfort for the monitor to identify the person of interest. It can be implemented to many other applications which involves the moving images. Reconstruction of images can be used in forensic to predict face from a low-resolution image. Results are shown in the results section.

IV. Face Alignment

This system is an *Intelligent Assistance System* that helps the user to track the target more accurately. An input image can be off-camera image that may lead to error. This system gives an option to user to perform face alignment. As human face can have lots of various poses and expressions. Sometimes two images of same person with different face expressions and pose are very different from each other. So, it becomes very tough for model to apprehend accurately. In a CCTV image same person is being tracked by

multiple cameras. But still the face can be non-frontal in each of the camera image. As showed in the cropping section 20% of the total face size has been added on both sides for the allowance to deal with the pose variation. Cameras are being installed on the road so the axis rotation in the pose cannot be varied too much as people be detected while standing, walking or sitting. It can be dealt with our designed cropping method.

In facial recognition, the main challenge in recognition is face pose. Different pose of face of same person are quite different from each other and may lose the important feature. For an example, the face of side pose face image and frontal face image of same person are quite different from each other and challenging in face recognition step. For accurate results in facial recognition, face alignment was performed – frontalization on image. The auto encoder structure provides an amazing transformation from side pose to face frontalization. Face image that is captured from CCTV camera is normally in front face. Therefore, the need of facial alignment is used to regenerate an image for the non-frontal face image of the target present in the library.

An iconic localization approach called Supervised Descent Method (SDM) is employed in the detection of facial features. It works especially well for jobs like face alignment, where the aim is to detect certain facial landmarks in a picture, including the corners of the mouth, nose, and eyes. SDM begins with a preliminary approximation of the facial landmark positions. An easier, less precise approach or hand annotation of a training dataset with ground truth markers are two ways to accomplish this. [306] The method of "Frontalization," developed by T. Hassner et. al. [306] which creates frontal-facing perspectives of faces from unrestricted photos, is the main emphasis of the study. This method uses one unmodified 3D surface as an approximation for all input faces, making the procedure simpler than prior approaches that attempted to estimate 3D facial forms for each query image. This technique enhances face recognition and gender estimation performance by generating frontal views in a simple, efficient, and highly effective manner.

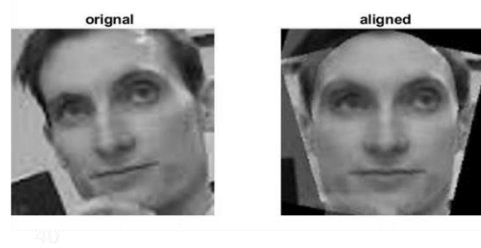


Figure 102 Face Without Facial Hairs

After passing the detected face to the alignment layer face is recreated and is kept inside the recognition matrix. First image is shown in the Figure 102 is the original image that is tilted on the axis and camera angle is also unable to show the frontal face. Second image is the output face after performing the alignment by the algorithm. If the first and second image of the Figure 102 is observed, it can be noticed here that the key features required for the recognition are still present in central facial matrix that represents the face. Similar happens to the facial hair matrix. Figure 103 shows an image that is not facing the camera and have facial hairs. This image is cropped from a frame where it was located at the bottom left corner. Second image from Figure 103 is perfectly reconstructed from the misaligned image and facial hairs are reconstructed well. The blurred areas around the face matrix will be dealt as background when passed to the recognition layer.



Figure 103 Face with Facial Hairs

Face is cropped and resized to the 50 x 50 pixels hence it does not take longer for alignment. To clear up this problem face alignment algorithm is carried out which align all face images and identical algorithm is also applied on test image also. Landmarks of face are projected from every image and from those landmarks, face is aligned. A collection of regression methods can be used to estimate the face's landmark coordinates without delay from subset of pixel intensities, reaching exquisite-actual time overall performance with excessive first-class predictions.

V. Graphical User Interface

Being an intelligent assistance system, a front end was required to communicate with the user and give the ability to achieve the desired results. Frontend/GUI is made using Tkinter. Frames uploaded on the cloud are accessed by the first detection layer that crops out the human faces. This task is done at the backend. Similarly cropped faces are passed through the recognition layers and placed in the relevant folders. Details already given in earlier sections.

The main screen offers all the options available for the assistance. First option is for the image search. Then there is an option for low resolution/ blurred image to high resolution image. Sketch to colour image conversion is also available in the main menu. Face alignment is available to generate frontal face image from side pose. Finally, the last feature is to add a camera. Main menu screen is shown in Figure 104.

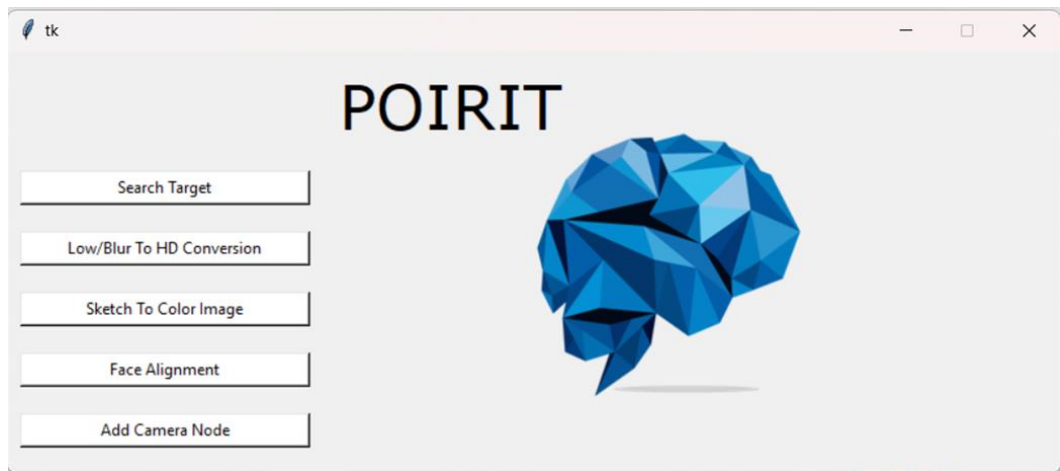


Figure 104 Main Page View

Figure 105 shows the view of the target search page. User is given multiple options to start the search. Firstly, user need to select the image from the database. Initially face is cropped along with the lower facial hair matrix and then converted to grayscale.

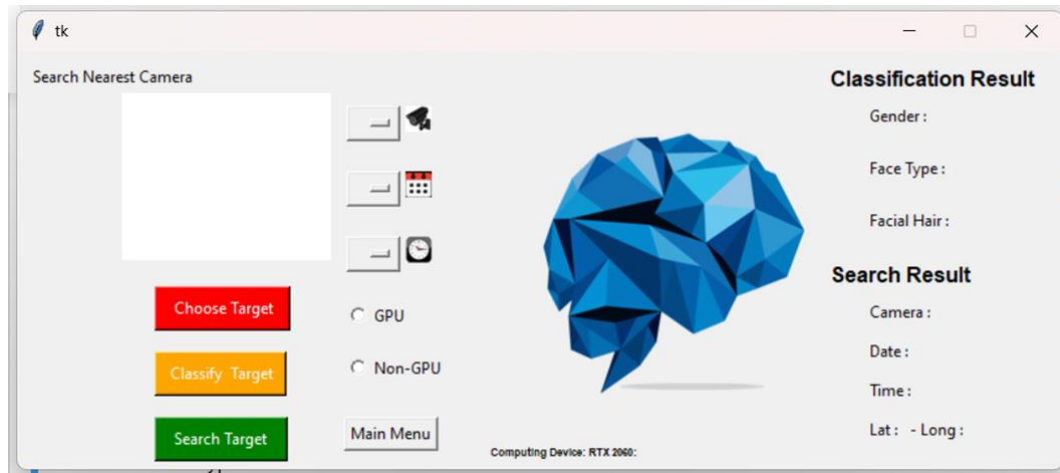


Figure 105 Face Search Page

User is asked to select the camera node to start the search shown in Figure 106.

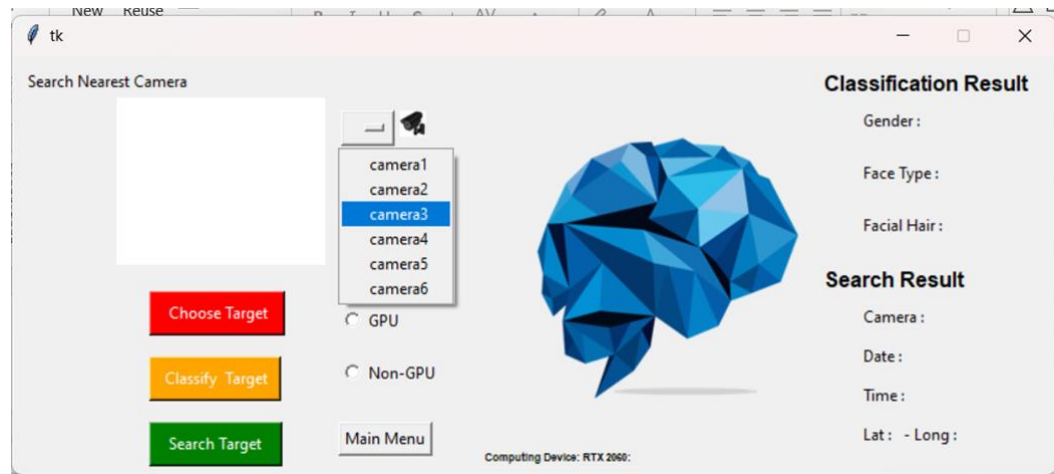


Figure 106 Camera Selection

Similarly, date and time of the event is required to select shown in Figure 107 and Figure 108 respectively.

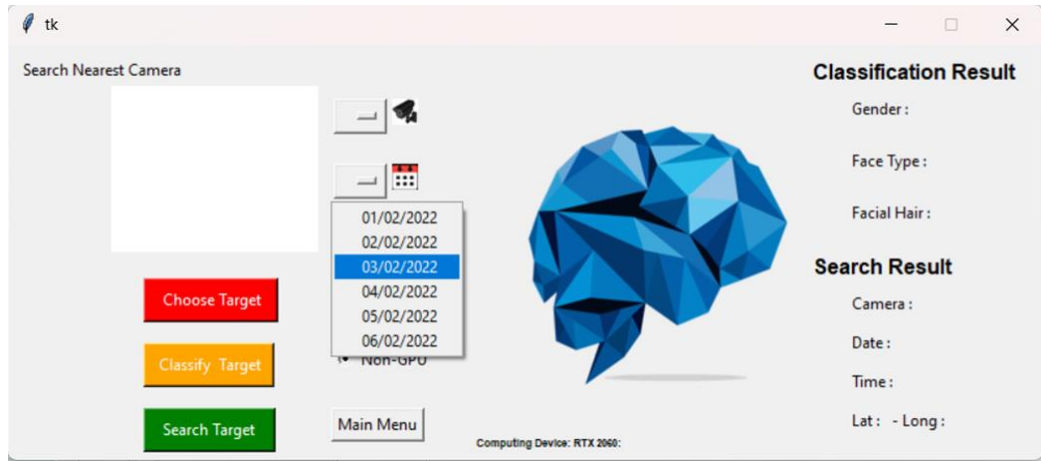


Figure 107 Date Selection

Classification result is displayed at the top right corner and search result is displayed at the bottom right corner. Currently the single folder search is implemented.

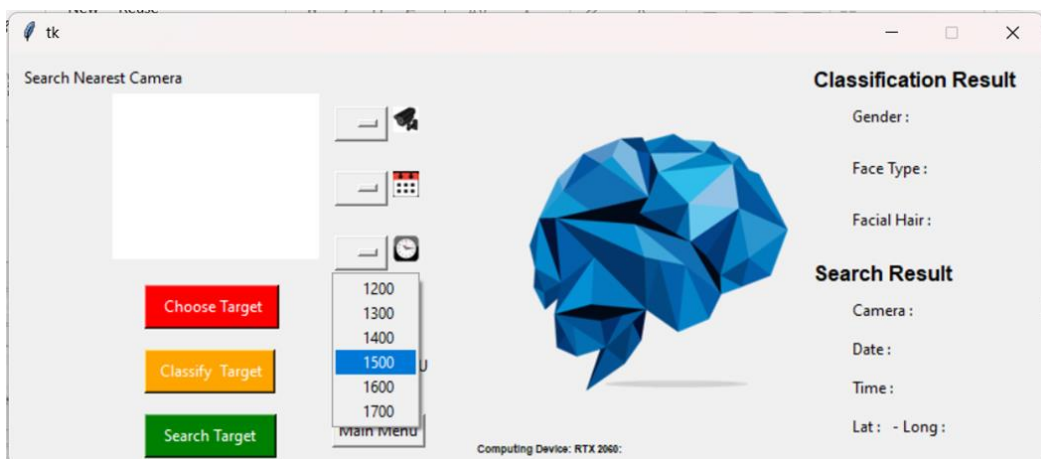


Figure 108 Time Selection

Another way of search can be performed by the nearest neighbor search algorithm. X and Y coordinates of the search system are the longitude and latitude of each CCTV camera node. Multiple cameras at the same locations are searched one by one. This GUI does not offer the option yet but is offered at later stages to track the occurrence of the target by radial search and draw the path for the person of interest.

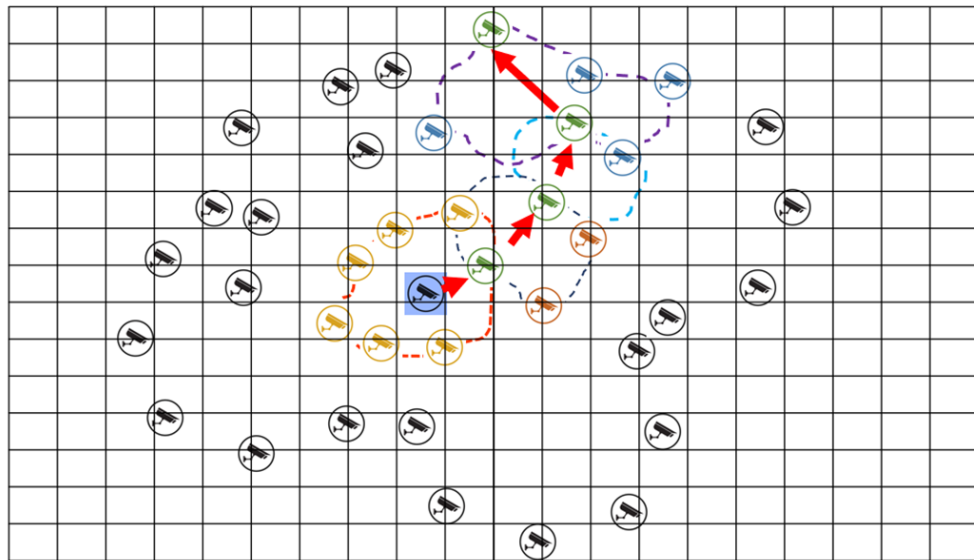


Figure 109 Radial Nearest Neighbor Search

Figure 109 shows the scheme where the blue node is the starting point, and the red dotted path shows the nearest neighbors. In this loop the green node shows the successful match. In the next step the other loop with dark blue dotted lines is covered and a similar approach is followed in upcoming loops. It must be noted here time folder must be updated/extended if the existence is found in the last minutes of the hour.

Sketch to colour image converter screen is shown in Figure 110. Users need to select the sketch first and then the colour image is generated on the click and saved at desktop of the system.

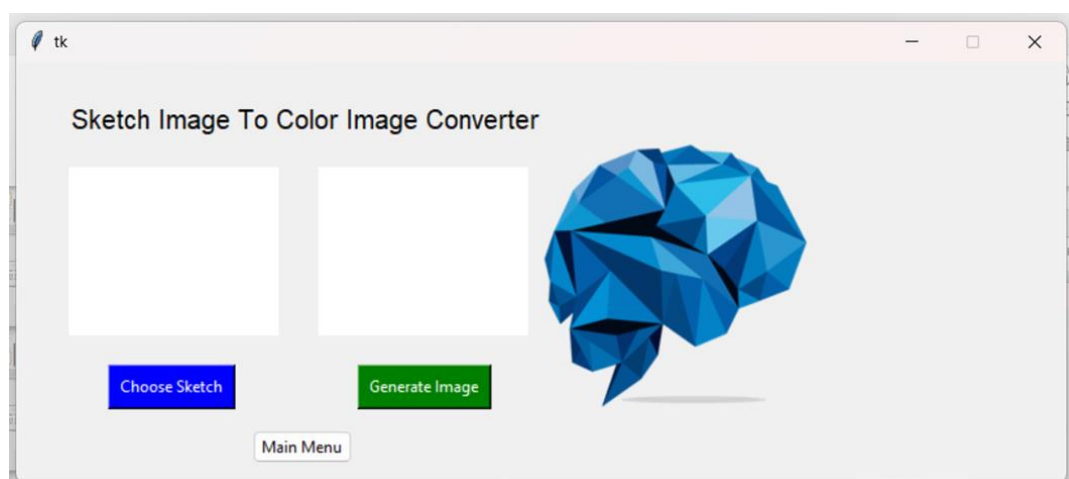


Figure 110 Sketch to Color Image Converter

Low/blur image to high resolution image converter screen is shown in Figure 111. Users need to select the low resolution/blur image and then the high-resolution image is generated on the click and saved at desktop of the system.

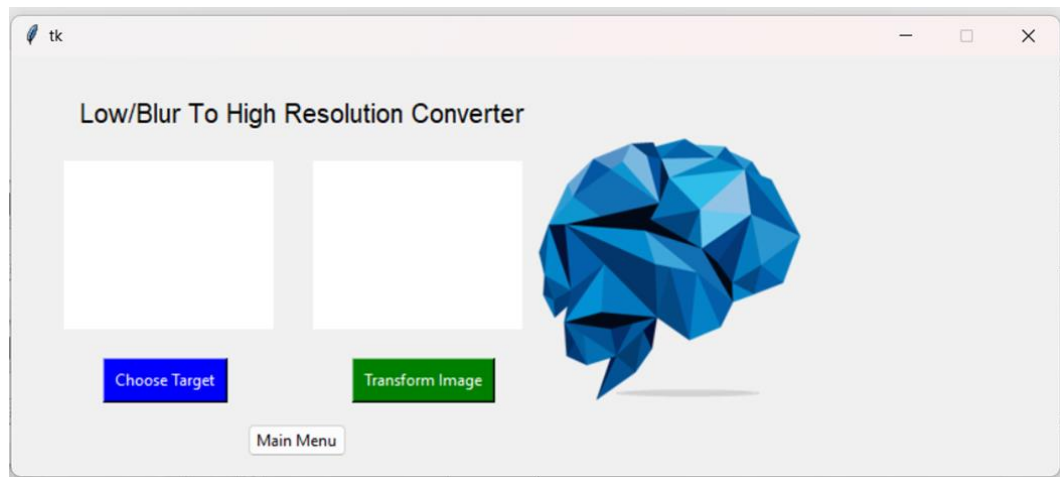


Figure 111 Low/ Blur to High Resolution Converter

Off camera/ unaligned or side pose images can be transformed using this feature. Screen is shown in Figure 112, selected image is aligned and stored at the desktop.

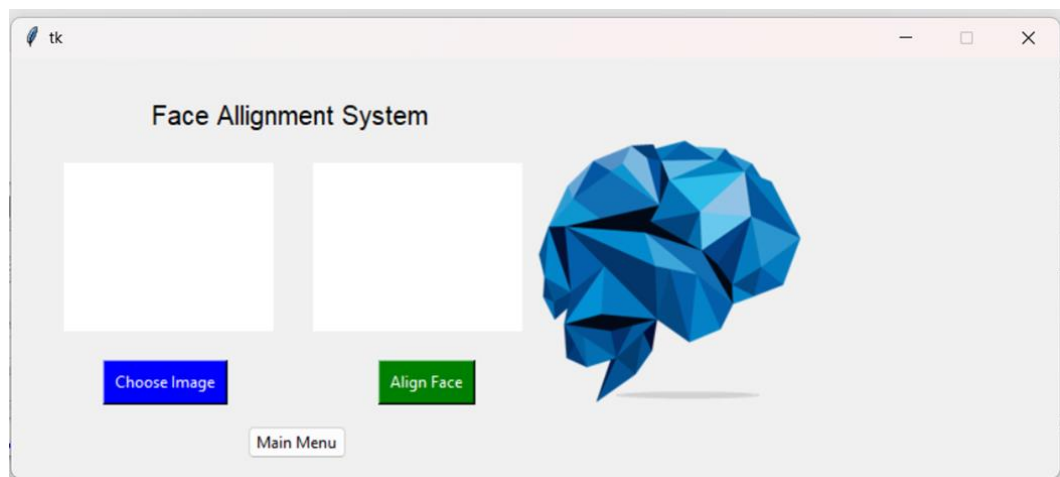


Figure 112 Face Alignment System

Last feature is shown in Figure 113, where cameras are added to system. Camera name is the primary key and other variables include the longitude, latitude, and IP address of the camera. This function accesses the Camera List.csv available in Frames and Faces folders shown in Face Categorization System. New row is added on the user request.

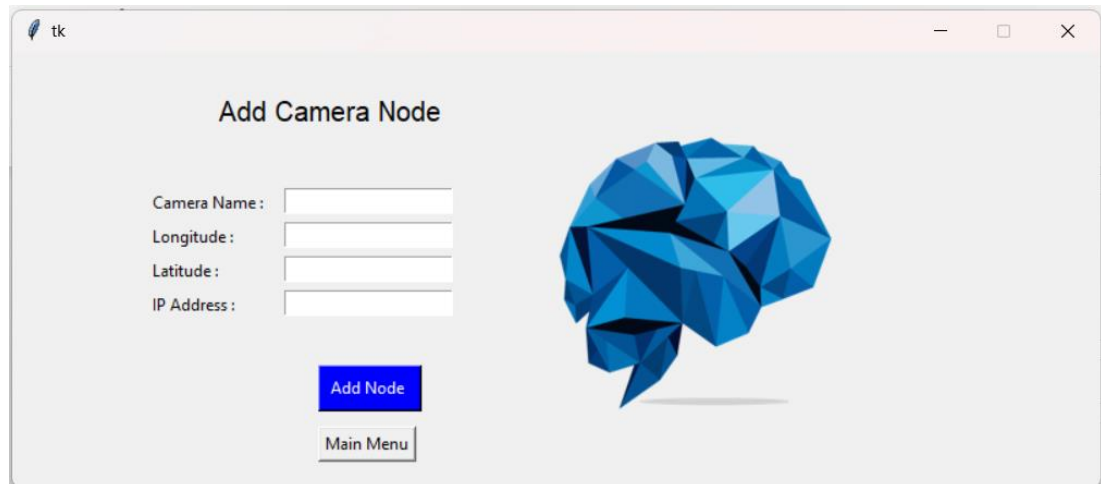


Figure 113 Add Camera Node

4.07 Summary

The complete system is divided into three major stages. The first stage is for image acquisition second stage is for face detection and the third layer is for classification of the targets into gender, ethnicity, and facial hair styles. The first stage hardware includes the CCTV cameras, PIR sensors, network switch and Raspberry Pi 4 as shown in Figure 114.

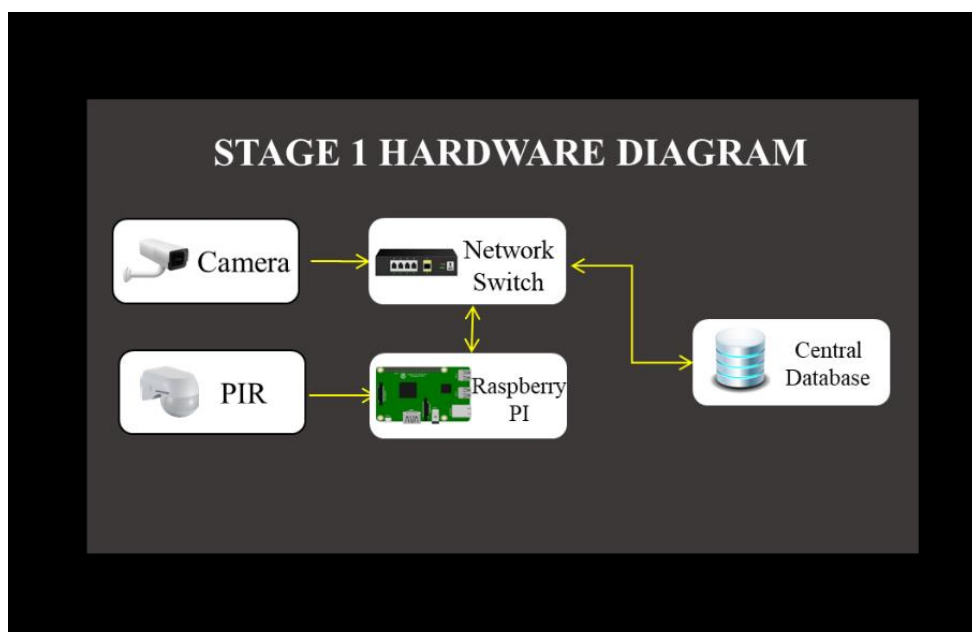


Figure 114 Stage I Hardware Diagram

In this research network CCTV camera is used and proposed with 8 MP resolution. Frame rate is 15 fps @ 50 Hz with 3480 x 2160 resolution. It has CMOS based 1/2.5"

image sensor with 6mm lens of 50 degrees of HFOV. IR range for night vision is 30m. Raspberry Pi 4 Model B powered with Quad CoreCortex-A71 (ARMv8).5GHz and 4GB RAM is used as processing unit. Long Range PIR Detector, OPTEX CX-702 is proposed for the final system.

Working of Stage I is shown in Figure 115. CCTV network cameras are connected to cloud through the network switch. Video quality can be altered by the administrators by logging into the network camera. Shutter rate of 1/1000s is to be set for avoiding trails in the images. Hue, saturation, and brightness are automatically adjusted by the modern-day cameras. These values of the captured frames can be changed by the computing device but consumes much larger time. Frame rate can be adjusted in the settings pane of the network camera for the video stream, it has no effect on the performance of the system unless it is higher than 2 fps. Network devices are selected based on the bandwidth required i.e., 7.68 Mb/s per camera in addition to the bandwidth requirement of video stream from camera. If change in infrared signature is detected by the PIR sensor, a pulse of less than 1 second is generated to give an interrupt to Raspberry Pi 4. After getting the interrupt Raspberry Pi starts capturing frames @ 2 fps from the video sequence available at network from the relevant camera.

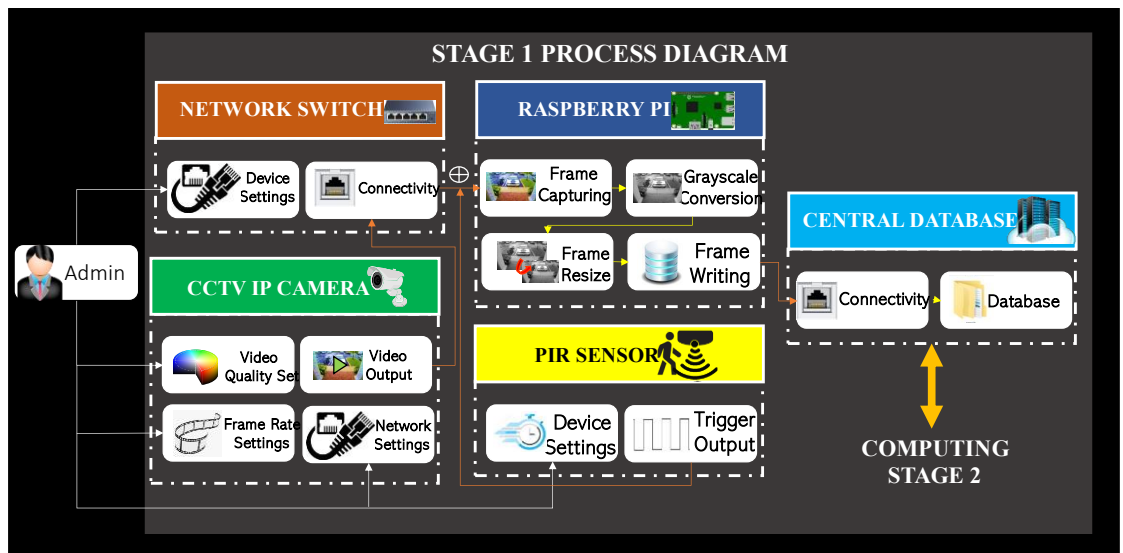


Figure 115 Process Diagram of Stage I

These images are converted to grayscale makes the processing time 3.33 times faster for upcoming operations. Frame is resized to attain 60 PPF at the maximum range of 50 feet with 8 MP. The original 3840 x 2160 is resized to 2792 x 1571. The

new image is reduced to 52.88% of the original image without effecting the performance. It increases the efficiency of the system by decreasing the bandwidth and memory requirements. Grayscale frame in *JPEG* format having dimensions of 2792×1571 occupies around 480KB whereas 3840×2160 in grayscale is around 741 KB large. The smaller image reduces the memory requirement to 64.7%. Equations are established to re-calculate the values of final resolution with the change of the parameters like field of view, working distance, camera resolution etc. Nomenclature for the frame is designed in such a way that it represents the camera ID that includes *zone, location, camera number* of that location followed by *date, time* and *frame number*. Each image is uploaded to the cloud server and placed in the relevant camera folder having subfolder of date and time in hourly manner.

Time consumption for each process for Core i7-9750H and Quad core Cortex A72 (Raspberry Pi 4) is shown in Figure 116. Total time consumed for all the operations for one frame was found to be 95.26 ms for Intel i7 and 148.98 ms for Cortex A72.

						
CPU	Read GPIO 1	Capture Frame	Grayscale	Resize	Name	Upload
Core (TM) i7-9750H	0.06 ms	1 ms	4 ms	20 ms	0.2 ms	70 ms
	Total = 95.26 ms					
						
CPU	Read GPIO 1	Capture Frame	Grayscale	Resize	Name	Upload
Quad core Cortex-A72	0.1 ms	2 ms	6.4 ms	32 ms	0.32 ms	110 ms
	Total = 148.98 ms					

Figure 116 Time Consumption at Processing Stage I

Figure 117 shows the hardware structure of processing stage 2 and 3.

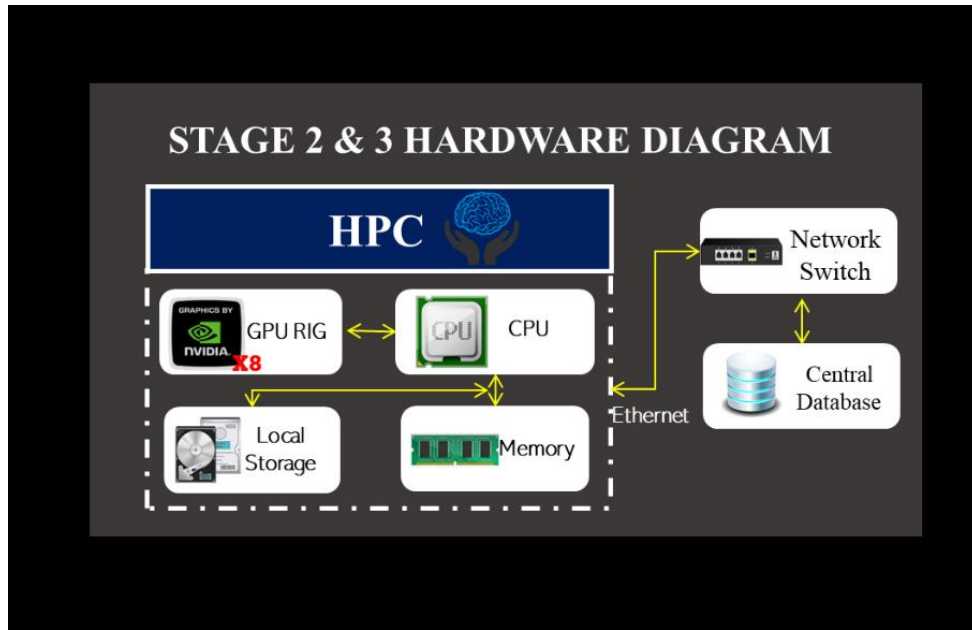


Figure 117 Stage II & III Hardware Diagram

Smart cities can be divided into zones, using an example of Lahore city with 10000 cameras, it is divided into 500 zones based on population concentration. Proposed High performance computers have *Intel Core i9-9980 HK* CPU with base frequency of 2.4 GHz, *C.B250A-BTC PLUS V20* motherboard that can support DDR4 memory up to 128GB. These computing systems contains eight *Nvidia RTX 2060* GPUs having 1920 CUDA cores with speed of 1050 MHz and 6 GB GDDR 6 memory. For fast data transfers *Intel Optane SSD 905P Series - 1.5TB* storage devices are proposed.

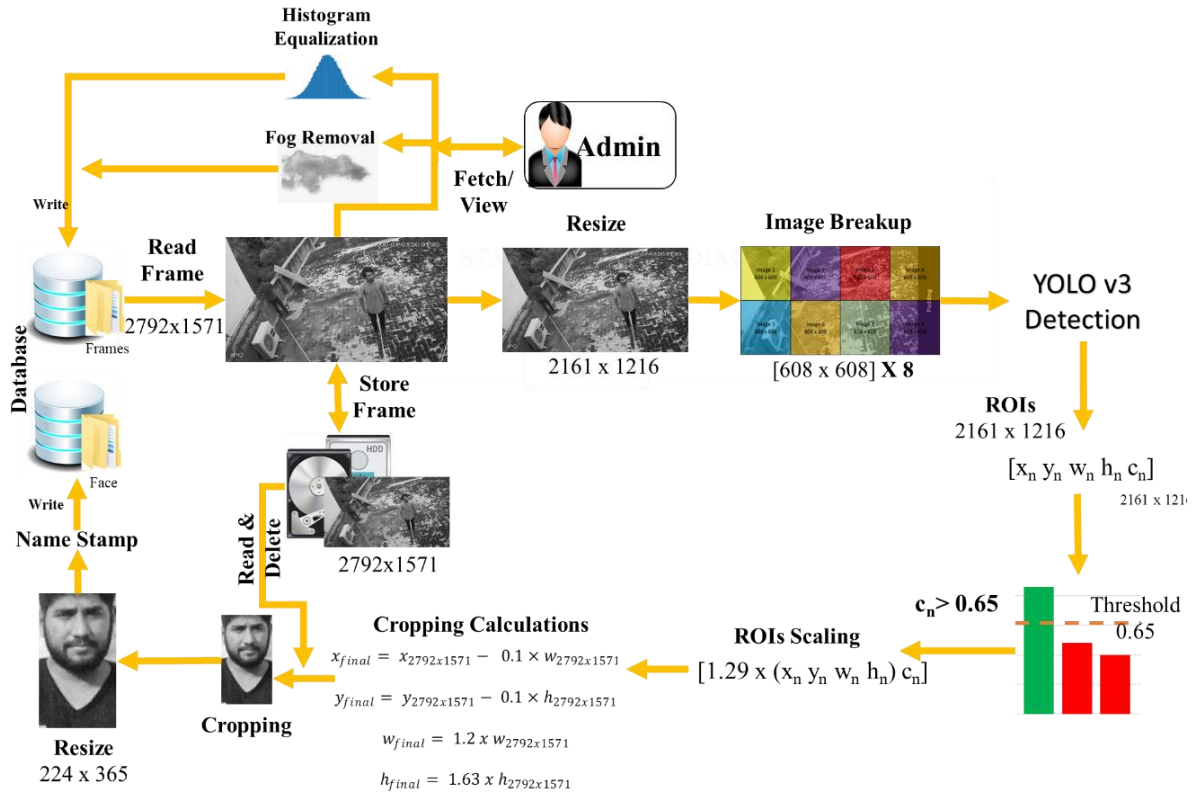


Figure 118 Process Diagram of Stage II

Processes involved in stage II are shown in Figure 118. HPC reads the image from the central database. This image has resolution of 2792×1571 and is stored in the local storage temporarily. It was calculated that resizing the image to 2161×1216 can increase the computing speed up to 30 fps using RTX 2060 Super. This image is then divided into 8 images of 608×608 with padding added to right side of sub image 4 and sub image 8. ROIs are detected using YOLO v3 model. Faces with confidence higher than 0.65 are kept. These ROIs are then scaled to 1.29 to get ROIs for 2792×1571 image. Final values are calculated so that additional regions at sides can be estimated. Using these new coordinates faces along with the facial hair regions are cropped from the stored 2792×1571 image. Then the images are resized to 224×365 . Calculations with justifications are available in frame resizing, face cropping and resizing sections. This image is stored to cloud with a new filename for face that is generated by $filename_{frame} U face_{number}$ of that frame. An additional option is available for this intelligent assistance system that can be used by the administrator manually. It can be used to remove Fog/Smog and histogram equalization. Retinex Algorithm with

Wavelet transform is used for fog/smog removal whereas *Rayleigh Distribution* is used for the histogram equalization.

For the processing stage III, same hardware specifications are proposed as of processing stage II.

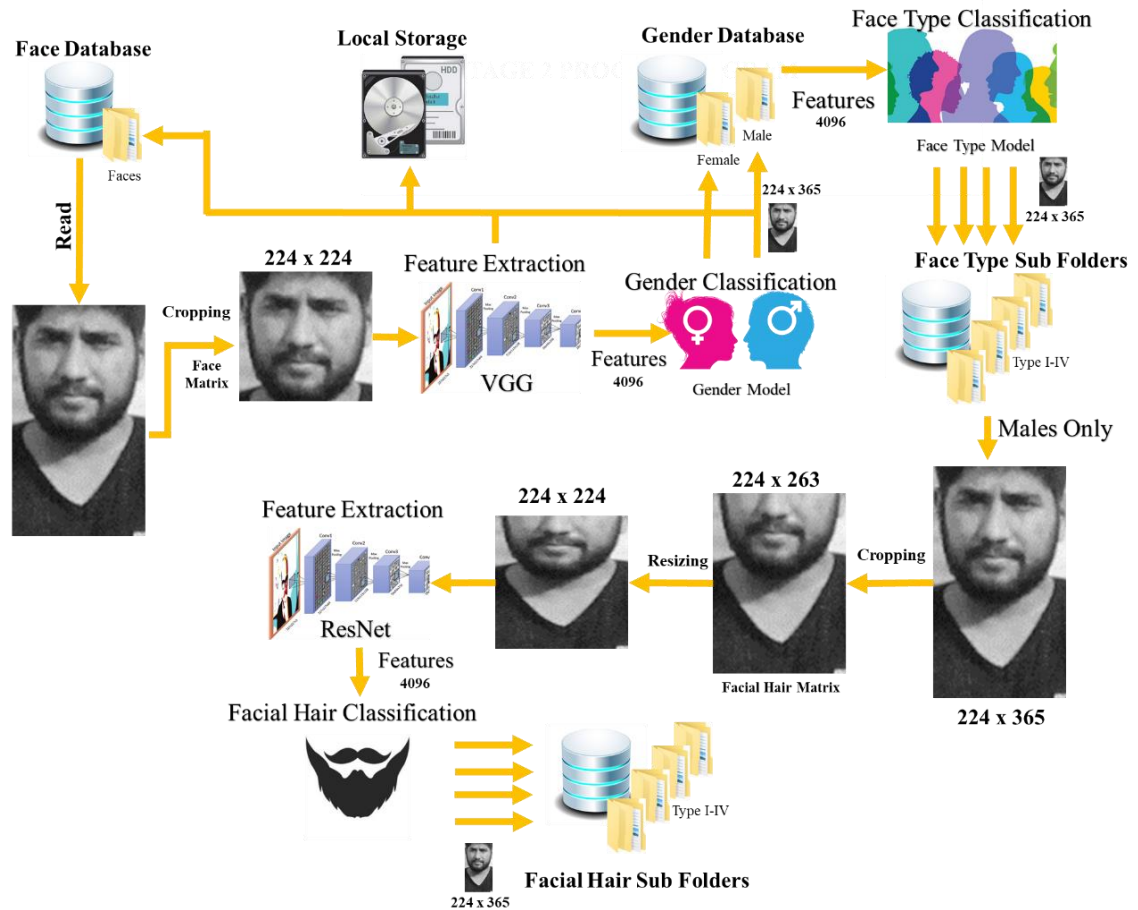


Figure 119 Process Diagram of Stage III

Figure 119 represents the process diagram of the processing stage III. It starts with the reading the image from the main database where extracted faces are placed in the relevant camera folders by the processing stage II. The first classification layer is to determine gender. *VGG Face* is used for the categorization. Upper matrix having dimensions of 224×224 is cropped and features are extracted by the model. These features are stored in the lookup table in .csv format in the face folder. Using these features, gender is determined and the face image of 224×365 is moved to the subfolder of the relevant gender. Extracted features of the faces those are already available are used in the next stage where face type is determined and the face image

of 224×365 is moved to the relevant face type folder after classification. In the final stage, image is read from the face type folders and lower matrix of 224×263 is cropped and then resized to 224×224 that is the input layer size of the network. ResNet-50 based model is used to predict relevant facial hair style. Input image of 224×365 is finally moved to the relevant sub folder of facial hair type.

Being an intelligent assistance system, as shown in Figure 120, this system offers multiple options to user and administrator. Using the frontend, the user can search the selected target by setting the camera node, date and time as starting reference. Target is classified to the relevant gender, face type (ethnicity) and facial hair style. Then it is searched in the relevant category folder sorted by stage III, from the starting node selected by the user.

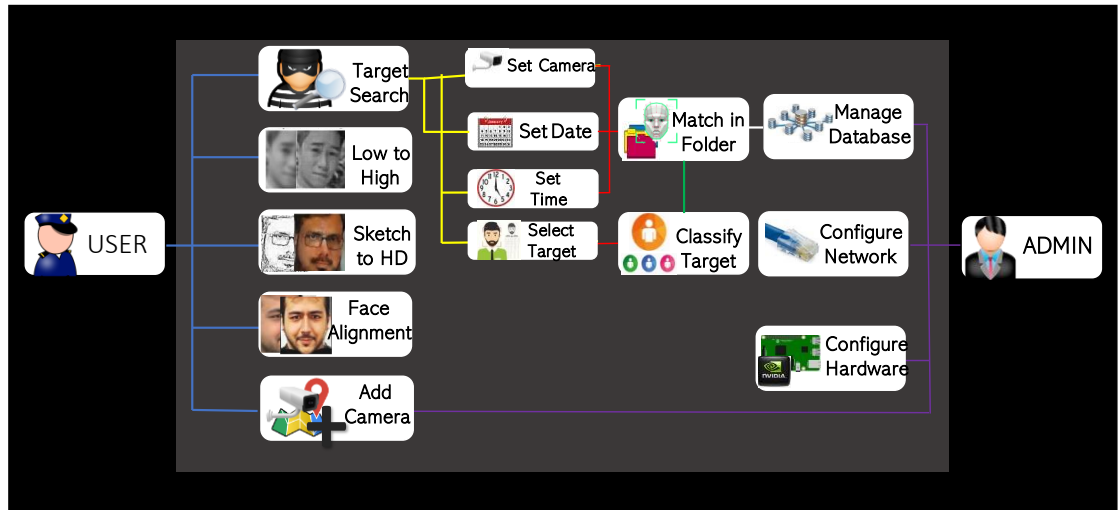


Figure 120 Use Case Diagram of POIRIT

Second option available is to increase the resolution of the input image using *MS CGAN*. Third feature is to estimate colour image from the sketch produced by the forensic team. This feature also uses *MS CGAN* algorithm. Face alignment can also be performed at this platform to increase the efficiency of the search system. Finally, the cameras can be added by the user. Whereas administrator can configure the network and hardware.

Chapter 5

Experimentation

5.01 Overview

The chapter elaborates on the strategic deployment of the design outlined in Chapter 4 across selected platforms and infrastructure, explaining the practical implementation approach undertaken. It starts with the discussing the computing devices required for the experimentation and the final prototype. Then it explains the need of the programming platforms for this research. Experimental setup describes the physical arrangements like placement of the imaging devices, varying lighting conditions, pose variations of the subjects and occlusions to generate different scenarios. Variance of speed of targets to adjust shutter and frame rates. Steps involved in training the face detection along with the learning rate and loss curves are elaborated in later section. After face detection algorithm, training details of all the models used for classification system, including *gender*, *face type (ethnicity)* and *facial hair styles* are discussed. Training parameters for *AlexNet*, *GoogLeNet*, *VGG Face*, *ResNet-50* and customized neural network are stated in each section. Moreover, training *accuracy* and *loss* curves are also displayed in each section for each training. Lastly, training process along with the *loss curves* are described for *low to high resolution* and *sketch to colour* image transformations.

5.02 Materials and Apparatus

The creation of intelligent systems for many applications has been facilitated by the improvement of deep learning algorithms and image processing techniques. The processing capability of the computers used for experimentation and model training is crucial for deep learning' models' effectiveness.

I. High Performance Computers

Initially research involved development of algorithms and blocks for image pre-processing, face detection and face recognition. As disused earlier that MATLAB

2021 with GPU coder was installed on this system, and it was found that software requires Nvidia GPU with compute capability of 3.2 or higher where our GPU does not meet this requirement. TensorFlow library with python were also used to perform training and to develop final models. The initial base system was Dell Alienware with core i7 3610QM processor. This processor has four cores and includes hyperthreading technology with giving this processor an ability to deal with multiple tasks those are no more to be scheduled saving the idle time. Cache is also important for our work as high computation speed is required and 6MB of cache memory provides more computation speed. This processor has clock speed of 2.3 GHz.

For better performance Asus Nitro with Intel Core i7-9750H processor having 2.60GHz speed, 6 hardware cores and 12 logical processors and 32 GB of RAM was used. The processor's base frequency is 2.60 GHz; however, it may be turbo-boosted to 5 GHz. The speedy and effective completion of the computations is guaranteed by the high-speed CPU. An NVIDIA RTX 2060 graphics card with 6 GB of GDDR6 memory was included in the machine. Solid state disk was used for the faster memory transfer from the memory to the processor. Having no mechanical parts involved and providing access to every cell simultaneously. It is more robust against ambient influences and making it more dependable. For the testing purposes this system was used. The graphics card is a great option for the experimentation setup because it is made to handle complicated image processing and deep learning computations. The graphics card offers enormous parallel processing capability for deep learning and image processing applications with its 1920 CUDA cores. The 32 GB of DDR4 RAM on the PC makes sure that calculations were completed quickly and smoothly. This system managed big datasets and models because to the enormous memory capacity. The 32 GB RAM ensured that training and data processing are not slowed down.

Impressive performance was achieved and saved huge amount of time using the most this high-end hardware configuration. For image classification, object detection, and segmentation, trained a variety of deep learning models, such as convolutional neural networks (CNN) and GANs. Experiments were performed with various architectures and hyperparameters because of the substantially shorter training period made possible by the fast processor and graphics card. Additionally, successfully used

this high-performance computer to several image-processing tasks, including denoising, deblurring, and image restoration etc.

II. Low Powered Computing Devices

Initially Raspberry Pi 3 Model B was used but was replaced by the Raspberry Pi 4 Model B in 2019 to attain more speed. As newer model has BCM2711 Broadcom, Quad Core Cortex-A71 (ARMv8) having 64 bits of SoC with an amazing speed of 1.5GHz. 4GB RAM variant was selected as it gives much faster speed for processing images with little compromise on price [307]. Ethernet was used where Wi-Fi is available but not required by the system. It has 40 GPIO pins but only one required per camera. Justification of selection is given in Chapter 4.

III. Triggering Module

HC-SR501 PIR sensor module was added to each camera shown in Figure 121 as triggering device. This sensor gives a trigger signal to Raspberry Pi GPIO pin when a change in infrared signature is observed. Now issue that was addressed before using the sensor module was to calculate the sensitivity of the sensor and the delay. This model has maximum range of 7 m and was used in prototyping. Its angle of view is 120 degrees more than that of camera view. Delay time has a range of 5 seconds to 5 mins. Both quantities in the module are adjustable with the help of variable resistors shown in Figure 122. But this module has a limitation while using with the 6mm 8 MP camera, as the camera FOV is 50 degrees that is far less than that of the PIR and the till the output remains high after detecting the change is also large as compared to the requirement. It should be placed in such a way that both camera and PIR array must be looking at the same region and time. Otherwise processing unit will be triggered to capture the frame when there is a change in the neighbouring camera view. It is only used for the evaluation purposes due to the cost constraints of the professional PIR module.

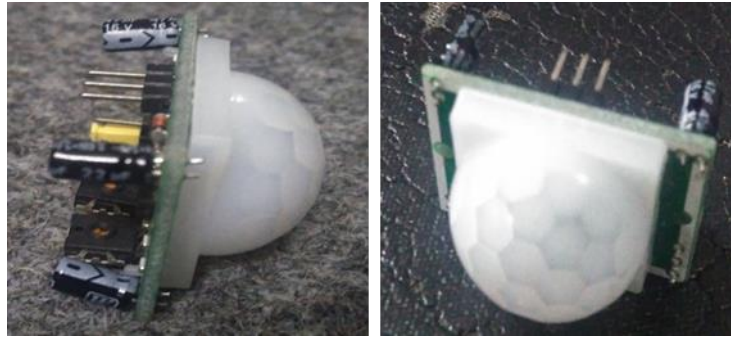


Figure 121 PIR Sensor Module Side and Front View

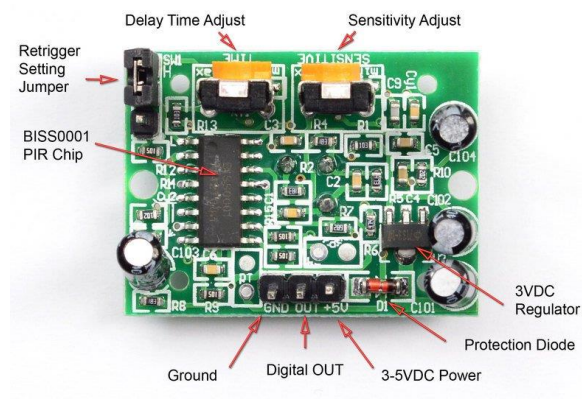


Figure 122 Circuit Board of PIR Module

Another variant of PIR Netelsan Ekson shown in Figure 123 with 360 degrees of coverage was used for experimental purposes. This was used to make system more sensitive at longer ranges.



Figure 123 Netelsan Ekson PIR

It can be replaced by professional Long Range PIR Detector of OPTEX CX-702 Series for the final system. It has a capability to detect movement at a maximum distance of 70 feet [308]. It only triggers when it detects radiation emitted by human body. Its processor has a capability to eliminate the smaller changes in the infrared pattern due to moving of tree leaves, flying birds and movement of small sized animals. Only abrupt high change is detected by this sensor. Range and FOV can be changed as per the requirement.

IV. Network Devices

Initially a normal network switch was used to perform the networking tasks with ethernet only support. For each node generally 2-4 cameras are required. Five ports (one for server connectivity, 3 for cameras and one for Raspberry Pi). Therefore, the TPLINK 10/100/1000 Mbps-TL-SG105 Ethernet Switch is used. Switch with more ports can be replaced at nodes where more than three cameras are needed.

For optical fibre connection, Multi-Mode Fibre Ethernet Media Converter was required. Initially, prototype was working with Ethernet and does not require media converter. On the later stage Cisco switch can be introduced to increase the efficiency in terms of communication speed. It must be noted here that the proposed system is developed in such a way that it can be integrated with the existing system. The current system requires a huge amount of bandwidth to transfer the video stream to command-and-control centre. Though it's a threat for the privacy and already been discussed in the introduction. Focusing on the networking, real performance of the system could only be judged by keeping all the parameters same.

V. Imaging Devices

Firstly, it should be noted that imaging devices are being used for the security and surveillance system. So only such type of cameras was used to make system more practical and affordable. Otherwise, high-performance cameras are also available, but they cannot be installed on mass level. That is why only initial tests were performed on professional photography camera Canon 50D.

System has been designed to work with any actual safe city cameras so similar night vision capability cameras is used with IR correction to control the exposure. Network IP based cameras are used in the system. It includes both wired and wireless cameras.

Wired cameras with ethernet are preferred because of the speed. Details for connectivity is discussed in the network devices section.



Figure 124 Dahua Network Camera

For this research three types of cameras were used for experimentation purposes. Camera with multiple level of resolutions was required to get various working conditions. All the above-mentioned parameters were kept under consideration. Experimental scenarios were created with their capabilities to replicate the real scenarios. It will be discussed in detail in the experimental setup chapter. Restricting to the hardware setup now, CMOS based Dahua (DH-IPC-HFW1420S) network camera as selected shown in Figure 124. It provides 4MP Full HD with 3.6mm lens with video in MPEG format. Used frame rate is 20 fps giving 2688×1520 resolution. It has Day/ Night modes.

The second camera is enormously powerful, and it replicates the actual node setup by the Punjab Safe Cities Authority in Lahore. However, the model of Dahua is cost effective with equivalent results as of expensive cameras for experimental purposes. System is designed in such a way that more input formats can be supported. Dahua (DH-IPC-HFW2831S-S-S2) network camera was used. It gives 8MP Full HD video available in MJPEG format. Original frame rate is 15 fps @ 50 Hz giving 3480 x 2160 resolution. It has 6mm lens with 50 degrees of HFOV. It has CMOS based 1/2.5" image sensor. It can be operated in wide range of temperatures and outdoor weather conditions. It has IR range of 30m for night vision option.

Third camera is Dahua wireless IP camera with 1/3" CMOS image sensor. It is also bullet-type camera. It has maximum resolution of 2688 x 1520 pixels. It has 2.7mm

lens with 100 degrees of view. It also supports MJPEG video codec as required by our system.

IP cameras are used in video capturing of various scenes. However, analysing these images on a remote computer can be difficult and computation intensive. Real-time picture processing and capture are possible with the Raspberry Pi 4. In this thesis, describes how to process images and capture IP camera frames using the Raspberry Pi 4. Details are given in the *System Design* chapter.

VI. Programming Platforms

Face detection and facial recognition research has made extensive use of MATLAB, a potent tool for deep learning and image processing. The upgrades and new features made in MATLAB 2017 to MATLAB 2022 have increased its usefulness for various applications. The advent of the Deep Learning Toolbox has been one of the most significant developments in deep learning using MATLAB in this research. For image classification, object recognition, and semantic segmentation, this toolbox was used that offers pre-trained models those were customized for purposes. Additionally, it incorporates numerous novel neural network topologies, such as transfer learning and generative adversarial networks (GANs). The Image - processing Toolbox in MATLAB was used with a variety of tools and functions for image pre-processing, segmentation, feature extraction, and classification for face detection and recognition. These resources were combined with the Deep Learning Toolbox to create models that were more reliable and precise. The Viola-Jones technique was also used in initial trials and for dataset preparations, which recognizes faces using boosted classifiers and characteristics like those in the Haar algorithm. The Image - processing Toolbox have seen a major revamp in MATLAB 2020a, new capabilities like automatic image enhancement and deep learning-based image segmentation feature was utilized in the research. The effectiveness and speed of face detection and identification algorithms were increased by these new techniques.

The most recent edition of MATLAB (2022a) also features performance and usability upgrades. Updates to deep learning frameworks like PyTorch and TensorFlow are included in the new release, and it offers better support for CUDA-enabled GPUs, which helped to speed up deep learning model inference and training.

The Live Editor of the latest version was used for modifying and executing code. For deep learning and image processing research on face detection and facial identification, MATLAB has been a key tool. Building and optimising deep learning models for these applications has become simpler with the inclusion of the Deep Learning Toolbox, Image Processing Toolbox, and several additional tools and functions in MATLAB 2017 to MATLAB 2022.

The usage of Python in deep learning and image processing research has made it one of the most well-liked programming languages in the world. The Jupyter Notebook, was used which offers an interactive environment for data analysis and visualisation, is one of the most widely used study tools in this area. With this web-based interactive computing environment, enables to write and run code, carry out data analysis, and produce visualizations. Jupyter Notebook has capability of organizing and presenting research work since it allows to integrate code, text, and graphics in one document. TensorFlow library was utilized for data flow and differentiable programming in a variety of activities, including deep learning. It includes several pre-built models for various applications, such as face detection and facial recognition, and it allows to design and train neural networks. TensorFlow, Keras is a high-level neural networks API that makes it simpler to create and train deep learning models. With just a few lines of code, Keras ability of having user-friendly interface for building complicated neural networks was utilized and its large number of pre-trained models for face detection and recognition made it more useful. Pandas, a Python data analysis and manipulation toolkit used for working with structured data, such as CSV, Excel, and SQL databases, it offers simple data structures and data analysis capabilities. Data pre-processing and data cleaning, which are key processes in getting data ready for deep learning and image processing tasks, was done with the help of this essential library. These libraries were combined to perform deep learning and image processing research for facial recognition and face detection. Furthermore, Python has a sizable and vibrant developer community that has produced numerous open-source libraries and tools for deep learning and image processing. Finding information, finding answers to issues, and collaborating with others are all made simple by this.

Octave GNU's Video Capture Package was used to capture frames from the IP Cameras. This software offers an easy-to-use interface for using the camera and taking

pictures. Additionally, Octave GNU's Image Package, can provides a wide range of image processing tools like filtering, segmentation, and feature extraction but only limited features like resizing and colour conversion is used. Real Time Streaming Protocol (RTSP) protocol was used to communicate with the IP camera. The Raspberry Pi 4 receives video from the IP camera through RTSP using the *GStreamer package*.

5.03 Experimental Setup

It was a difficult task to perform experiments on roads, to cope with this issue an open space available at Department of Electrical Engineering, University of the Punjab was selected to use install the setup for replicating the real scenario. As it is already discussed in the system design section, range of the camera must be selected to give at least 60 PPF. To replicate a street view using an IP camera mounted at a height of 13 feet, the experimental setup would involve several components, including the camera itself, a mounting bracket, a power source, and a means of transmitting data. The camera was selected based on the required specifications, such as resolution, frame rate, and field of view. Camera was chosen to capture high-quality images in varying lighting conditions. The mounting bracket was used to securely attach the camera to a pole or other structure at a height of 13 feet. It was important to ensure that the camera is level and stable to avoid any distortion or image shake.

The power source for the camera was an AC power outlet, depending on the specific requirements of the project. It was ensured that the power source is reliable and can provide sufficient power for the camera's operation to avoid disconnections. Additionally, the data transmission method would need to be selected based on the required range and reliability. Ethernet was used for data transmission, depending on the location and distance of the camera from the receiver. The next step has been to set up the camera's software and settings. That includes configuring the camera's image resolution, shutter rate, frame rate etc. It was crucial to optimize the camera's settings to ensure that it can capture high-quality images and transmit them reliably. Raspberry Pi was connected to the same network to capture frames from the incoming video stream.

I. Placement of Camera(s)

The camera used for the testing was Hikvision DS-2CD2085FWD-I, it has resolution of 3840×2160 and maximum frame rate of 20 frames per second. Frame rate was decreased in the software as 2 frames per second. As shown in Figure 125 the camera placement replicates the setup used at roads crossing in Lahore city. It has height of 13 feet from the ground. These cameras are arranged in such a way to provide 50 degrees of FOV. These cameras are adjusted in such a way to see at a range of 50 feet.



Figure 125 Camera with Light

Keeping in view all the above discussed constraints, a similar approach was adopted while setting up the cameras for experimentation. Movable channel of iron clamp was used to hold camera on pole. It has a provision to change the height of the camera and bottom frame has variable tilt so that the angle of depression can be varied. Marks for the ease of measurement were placed on the pole.

Distances of the people from the camera were required to verify the effectiveness of the capturing device. Field of view and distances were marked on the floor with the help of reflective tape. It helped to perform analysis while performing the experiments. It is shown in Figure 126.

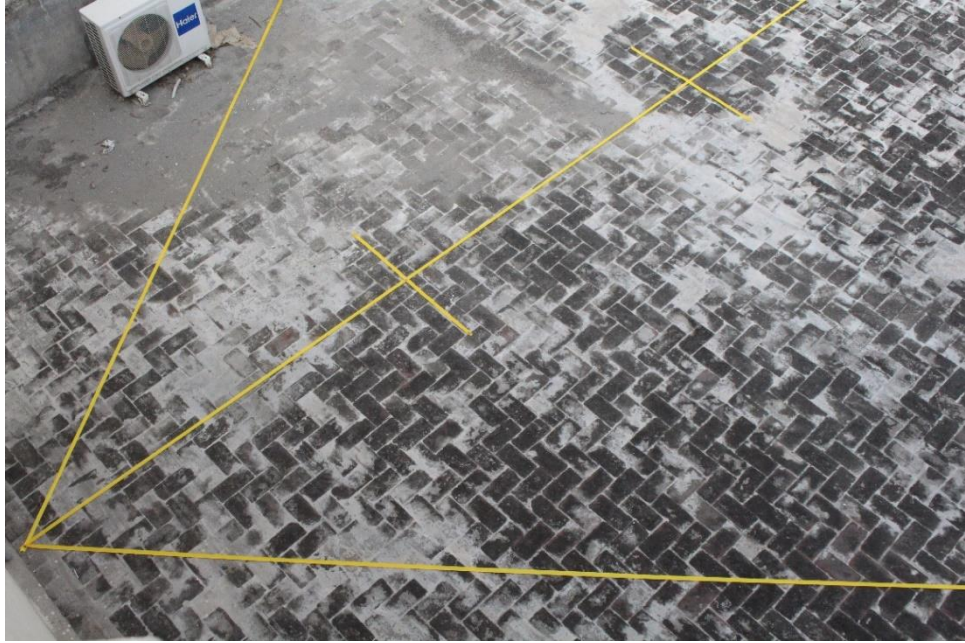


Figure 126 Distance Markings

As shown in the Figure 127, the majority of the background for the humans is of the similar shade that is grey. Face cropping is the first step of the process. For testing the robustness of the detection system background of the test people was altered with the help of flexes having various types of the prints shown in Figure 127. This setup gives more challenging environment for testing the system in addition to the training images and video sequences. It helped to check the robustness of the detection system.



Figure 127 Background Alterations

II. Lighting Effects

As discussed in the System Design chapter multiple parameters can affect the image. Lighting is one of the major constraints to be considered while designing a surveillance system. Latest cameras include the manual image quality settings as shown in Figure 128 as well as automatic brightness control with respect to ambient light.

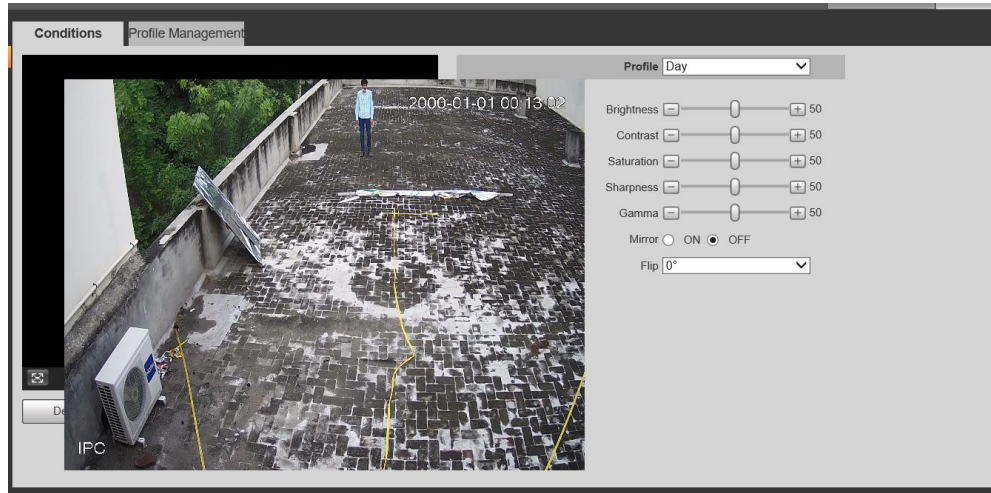


Figure 128 Camera Light Condition Settings

This system is designed in such a way that cameras should adjust the brightness and saturation automatically by using its own IR sensors. It was required by the testing environment to have the ability to replicate such variances. Cameras used in the system can effectively work all day. Infrared sensors are used in the cameras to adjust the lighting conditions, but in the surveillance area additional lights are used to monitor the activities of pedestrians. To create similar environment, 50 W LED with 10 mm lens were used to light up the targets at different distances. Lights were placed at all the four sides of the experimental area. As it can be seen in Figure 129 and Figure 130, Pure white lights were used to replicate the real projections light installed at the camera sites and lights from sideways.



Figure 129 Projection Light



Figure 130 Side Light

It was observed in the real time video sequences that sometimes light flares are caught in the camera. These are due to slower shutter rates and reflection from the shiny surfaces present at the site. This phenomenon is widely observed when light source is not static and passes through the frame of the camera. As discussed earlier, the experimental arrangement was made on the roof top and was not possible to get vehicles there. To create such scenario, instead of vehicles, sheets covered with aluminium foil were used. Few small mirrors were also used to spread the light. In Figure 131, preparation of the large aluminium coated reflective sheet is shown.



Figure 131 Aluminum Foil Sheet

These sheets were moved manually to replicate the movement of vehicle. Wheelchair was used to carry the sheet smoothly along the designated path. This setup was made with the help of staff members of College of Art & Design, University of the Punjab to get the expert opinion. This method is adopted in professional photography. System was tested in all time of the day and at night. Sun rises from the left and sets on the right of the camera point. Light was used with butter paper in front of the camera to replicate the sun rising from the front of the camera.

III. Pose Variations & Occlusions

As discussed in the system design, this system can deal with variety of poses that can be adopted by the pedestrians in the environment. It was observed by observing the CCTV footages that subjects do not change the orientation of the face more than 10 degrees in normal walk. Whereas targets may look off-camera while traveling from one place to other. This problem can be handled by covering one spot by multiple cameras focusing from different sides. That is why experimental setup was made in such a way that various angular variation around the axis were adopted by the people taking part in the experiments. Rotation of the face causes the cropped image to skip

complete face coordinates. In this regard the compensation rectangles were used to include the important part of the face. It was evaluated with the help of experiments shown in Figure 132 that applying 10 degrees of change to face image increases the dimension to 8% area on either side. The cropped image includes the sides as thoroughly discussed in system design section.



Figure 132 Face Rotation

Moreover, face directly facing to the camera is the best case when both the sides of the sides are exposed equally to the camera lens. As already stated in the system design, face alignment is to be done to every image. It can deal with the axis rotation as well as the side variation.



Figure 133 Off Camera Face

Multiple scenarios with the variance of angles of attack were created like Figure 133. It helped to test the level of robustness of the system against the face exposure to the cameras. Another test was to test the ability against the occlusion. Different levels of occlusions were added to faces. It helped to verify the system's ability to detect and recognize faces with a little occlusion. Moreover, occlusion is a category of occurrence that can be dealt with as alarm in specific scenarios. This category has not been added to this research but successfully conducted to detect masks. That can be integrated at later stages.

IV. Variance of Speed and Movement

In the system design section, shutter rate and frame rate were set by keeping in view the parameters of the real time environment. Fast moving objects leaves trails of light when lower shutter rates are used. For face detection, it can cause blurred faces and finally losing the features. On the other hand, slower frame rate can miss the targets in the frame. Speed of the person of interest travelling through the frame was one of the important parameters while setting the frame rate. In this regard scenarios for experiments were created by moving targets with different speeds. Measured distances were marked on the floor for the reference points to calculate the speed of the targets.

PIR sensors were used to trigger the Raspberry Pi to capture frame when there is change in the IR pattern due to movement of anything in the field of view. Scenarios were created with targets moving in numerous directions within the field of view with different speeds. This experimental setup also helped to adjust the sensitivity of the PIR sensors.

5.04 Face Detection

Training an object detection model like YOLO v3 with a dataset of 10,000 images requires a carefully planned setup to ensure optimal performance and accuracy. To train a YOLO v3 model, high-performance system with a powerful graphics card was used. NVIDIA GPU, RTX 2060 was used, to accelerate the training process. The system should also have a minimum of 16 GB of RAM and enough storage space to store the dataset. The dataset of 10,000 images were carefully prepared to ensure that it covers a diverse range of objects and scenes that the model is expected to detect. The

dataset was labelled using MATLAB image labeller tools was used to annotate objects with bounding boxes and class labels. The annotations were saved in the format that YOLO v3 can understand. The YOLO v3 model was trained using the Darknet framework, which is a free and open-source neural network framework. It was required to configure the training parameters such as learning rate, batch size, and number of iterations. Learning rate of 0.001 and a batch size of 8 was used for the training. Number of classes in our dataset was set to 1. Once the training configuration was set up, the training process was started. The YOLO v3 model was trained using backpropagation with stochastic gradient descent (SGD) optimization. The model was updated iteratively by computing the gradients of the loss function with respect to the model parameters and adjusting the parameters to minimize the loss. The training process took several hours, due to the complexity of the dataset and the performance of the system. Once the training was complete, the performance of the model was evaluated on the validation set. Mean average precision (mAP) was used to measure the accuracy of the model. Where the performance was found not satisfactory, model was fine-tuned by adjusting the training parameters and adding more images data. Details of the dataset and the model are given in the Face Detection Section in System Design chapter. For training the model, batches were set to 64 with 16 subdivisions. Total number of epochs were set to 60, making total 157 number of batches per epoch. There is total of 37680 iterations and 628 iterations per epoch.

Two types of the training curves were observed during the training process, Training Loss vs Iteration Curve and Learning Rate vs Iteration Curve. The loss training curve is an essential tool for monitoring the training process and evaluating the performance of the model. If the loss does not decrease over time, it may indicate that the model is not learning correctly, and the training process may need to be adjusted. Additionally, if the loss decreases too quickly or too slowly, it may indicate that the model is overfitting or underfitting the data, respectively.

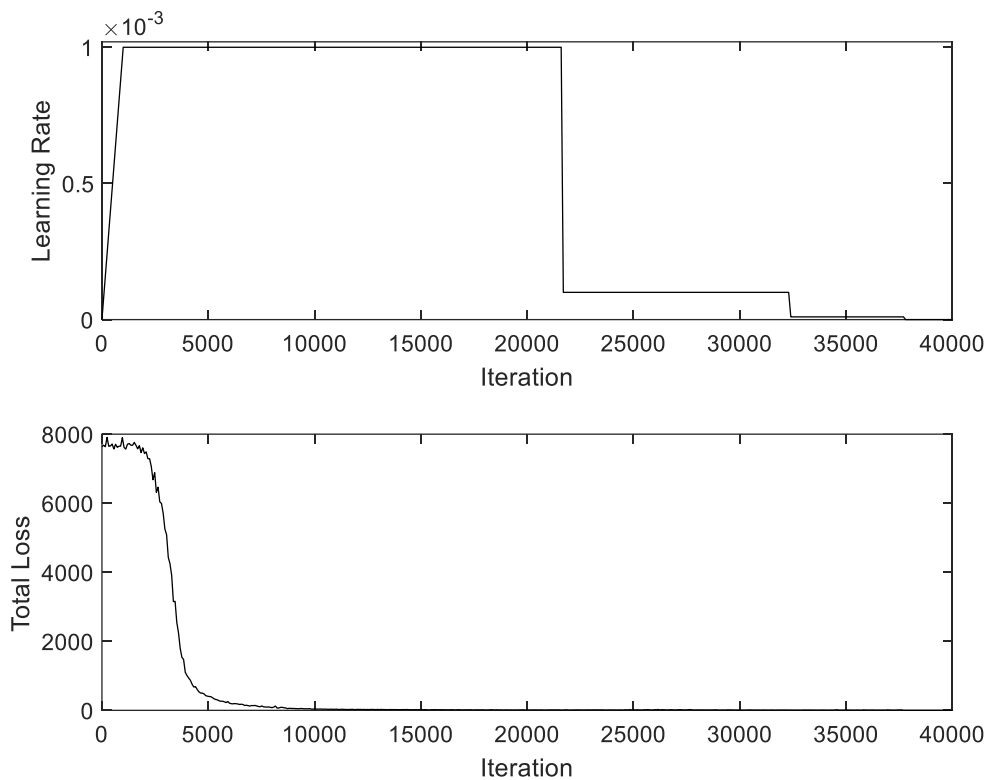


Figure 134 YOLO v3 Model Training curves

The loss training curve of YOLO v3 for the dataset is shown in Figure 134, graph shows that the training loss of the model changes over time during the training process. The loss function is used to measure the difference between the predicted bounding boxes and the ground truth bounding boxes. The YOLO v3 model is trained using the Darknet framework, which uses stochastic gradient descent (SGD) optimization to adjust the model parameters to minimize the loss. The loss training curve starts at a high value of 7635 and gradually decreases with the ripples in the start from around iteration number 3500. It shows that the model becomes better at predicting the correct bounding boxes after that point. As the model learnt from the training data, the loss decreases to a negligible point at iteration number 8000. It can be interpreted that at this point, the model has learned as much as it can from the training data, and further training may lead to overfitting. There are several types of loss functions used in object detection models, including the mean squared error, binary cross-entropy, and focal loss. The YOLO v3 model uses a combination of these loss functions to improve its accuracy and reduce false positives.

The learning rate curve is also represented in Figure 134 for YOLOv3 model training. It shows visualization of the learning rate during the training of the YOLO v3 object detection model. The learning-rate diagram shows the learning rate as a function of the number of iterations during training. Fixed learning rates are simple to construct but are not the best option for all training epochs. As a result, it is usual to adopt a schedule that gradually modifies the learning pace. The step decay and lambda decay learning rate schedules are two common varieties.

Step decay strategy is used for learning rate that involves reducing learning rate by a certain amount at predetermined intervals or epochs. A learning rate of **0.001** was the starting point. Warm-up phase is first few hundred iterations, the learning rate is gradually increased from a small value to the initial learning rate. This helps the model to start learning more quickly and stabilizes the training process. For models with a lot of iterations, like YOLOv3, step decay can be useful.

5.05 Models Training for Classification System

As discussed in the earlier section, total 10000 images per gender were collected from which 50% data was used for training and 50% for testing. Images of both genders were collected for training the models. Datasets includes the open-source images and images of the participants. Due to the ethical constraints in Pakistan, majority were the male participants from Department of Electrical Engineering participated in conducting the experiments. Only pictures of male participants are included in the text with their consent. The dataset is used to train three types of the networks, AlexNet, GoogLeNet, VGG-Face, Custom Neural Network and ResNet-50. First of all, the training images are downgraded to 50x50 for the intentional pixel loss and then resized to 224 x 224 to match the input layer size of the network. As there are only two categories, that is why binary classification is used.

Pakistan is the country with an extra ordinary important geopolitical importance. It is located at the junction of different civilizations. People living in the surroundings have different races with prominent distinguishable features. These races overlap in the country of Pakistan making it different from all over the world [289]. Detailed discussion has been done in the System Design section. The images of four types of ethnicities or face types were collected. It was a hard job to find open-source images

for Pakistan region. Major local ethnicities are descendants of ethnicities like Persians, Asians, Europeans, South Indians, North Indians Africans etc. Images of the people belonging from the root of the face types are also added to provide the characteristic features of the type to the respective model. For example, Central Asian face images are added to define the Face Type I, European faces are utilized in addition to Pathans for Face Type II, Face III includes typical North Indian faces those live in Punjab region and East African and South Indian face images are used to define Face Type IV. These images are also downgraded as of the gender classification dataset. This recognition layer was also tested with the real test participants. Luckily, University of the Punjab is the largest public sector university of Pakistan and students from all over the country come here to study. Students from various departments were invited to participate in the experiments after filling the consent form.

The third classification layer was the facial hair style layer. Dataset preparation for the facial hair types was a crucial step in developing the model that can accurately identify individuals based on their facial hair styles. In this context, a dataset of different facial hair styles was prepared to train deep-learning models to identify individuals with different levels of facial hair having 4000 images per class. In addition to open-source images, system uses the images from the video sequences captured from the experimental setup. There were fewer participants available for long beard hence used artificial beards to replicate the category. These beards were transformed in various shapes without changing the overall length to test the adaptability of the system. Dataset is divided into four facial hair styles, Type I is for clean shave individuals, samples with moustache only are categorized as Type II, long beard people are labelled as Type III and finally short, bearded people are categorized as Type IV. To ensure that the dataset is representative of the population, the images are collected from diverse group of people of different ages, and ethnicities. The images collected were under different lighting conditions to ensure that the model can handle variations in lighting. It was important to ensure that the labels are accurate, and there are no mislabelled images in the dataset. Once the images were labelled, the next step was to pre-process them. This involves cropping the images to focus on the face, rescaling the images from 224 x 263 to network input size that is 224 x 224, and converting the images to a standardized format, which is JPEG. As discussed in the

System Design, the lower matrix from mid-nasal point to 63% additional pixels below the face were cropped for the training the system. Face detection algorithm is used to prepare dataset for training. Calculations are shown in the Face Cropping section of System Design chapter. Images were normalized, to achieve consistent brightness, contrast, and colour balance.

After pre-processing the images, the next step was to split the dataset into training and testing sets. The training set was used to train the deep learning model, while the testing set is used to evaluate the accuracy of the model. The dataset was splinted randomly, with 50% images in each category in both the training and testing sets. Finally, it was ensured that the datasets are balanced. This means that there should be an equal number of images in each category to ensure that the model does not become biased towards one category.

I. Models Training using AlexNet

AlexNet model was trained in MATLAB by utilizing the Deep Learning Toolbox. As discussed in the System Design Chapter, dataset was carefully prepared for the genders, face types and facial hair types. The dataset is divided into training, validation, and testing sets. The AlexNet model is defined in MATLAB. The model has eight layers, including three fully connected layers, three max-pooling layers, and five convolutional layers. There are 96 filters in the first convolutional layer, 256 filters in the second and fourth layers, and 384 filters in the third and fifth layers. Each of the completely connected layers contains 4096 neurons.

Data augmentation plays a vital role in model optimization. Before passing data in training process, data augmentation was performed in which images processed by Scaling, Cropping, Flipping, Padding, Rotation and Affine transformation. Moreover, augmentation like Brightness. Contrast. Saturation. Hue also performed on images to enlarged dataset for training on AlexNet.

(A) Gender Classification through AlexNet

For Gender Classification, there are total 5000 images per gender for training and validation. To obtain best results stochastic gradient decent with momentum – SGDM optimizer with *learning rate of 0.01* was selected. To reduce overfitting *L2*

regularization is used with parameter of *0.001*. Moreover, *L2norm* is selected for gradient threshold method, *learning rate drop factor* is *0.1* and *rate* is *10*. *Momentum* is kept *0.1*. Data is divided in 10 mini batches and for maximum 6 *epochs* with shuffling at each epoch, after 6 epochs accuracy showed constant behaviour. 50% percent of dataset is for training and validation and remaining is for testing.

The training and validation curves in Figure 135 show the model's development as it is being trained. The progression of the training loss and accuracy is depicted by the training curve. The validation curve with black line displays the validation loss and accuracy changed over the training process. Training process took 721 minutes.

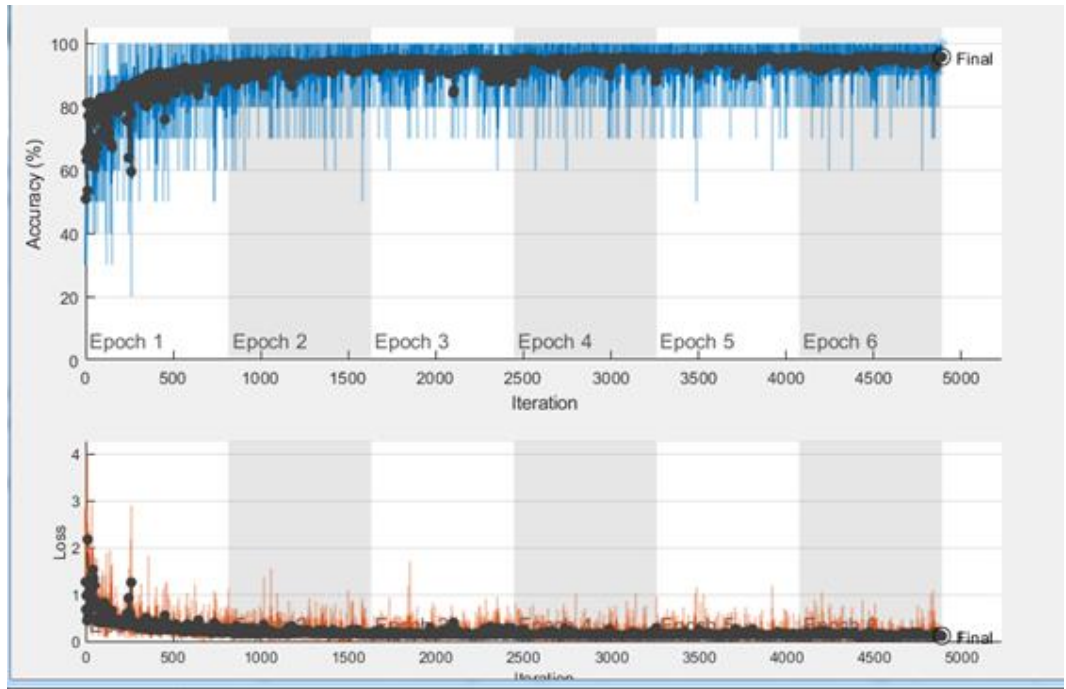


Figure 135 AlexNet Gender Classification training

Figure 135 represents the training scenario of AlexNet for gender classification. In 4890 iterations of 6 epoch training reached its maximum possible accuracy which is **95.79%** on validation dataset.

(B) Face Type Classification through AlexNet

For Face Type Classification, there are total 2000 images per face type for training and validation. For Face Type Classification, *learning rate* of *0.01* was selected and *L2 regularization* is used with parameter of *0.001*. *L2norm* is used for gradient threshold method, *learning rate drop factor* is *0.1* and *rate* is *10*. *Momentum*

is kept 0.1 . Data is divided in 10 mini batches and after, accuracy became constant after 6 *epochs*. Complete training took total 412 minutes.

The Figure 136 displays the training results of face type classification using AlexNet in MATLAB as a pre-trained network for transfer learning. The validation accuracy approach to **89%**.

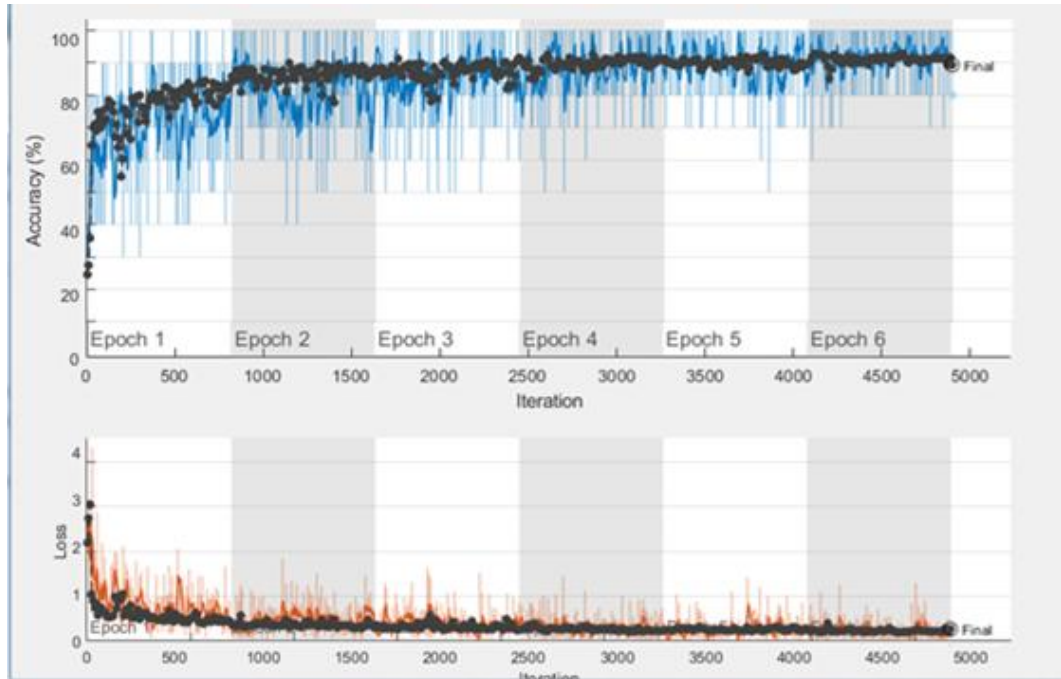


Figure 136 AlexNet Face Type Classification

(C) Facial Hair Type Classification through AlexNet

2000 images per Facial Hair Type were used for training and validation for classification model. For Face Type Classification, *learning rate of 0.01* was selected and *L2 regularization* is used with parameter of 0.001 . *L2norm* is used for gradient threshold method, *learning rate drop factor is 0.1* and *rate is 10*. *Momentum* is kept 0.1 . Data is divided in 10 mini batches and number of *epochs* were set to 6.

From Figure 137 , It can be inferred that no valuable features information was learned by the network. Hence this model was discarded, and no more calculations are made.

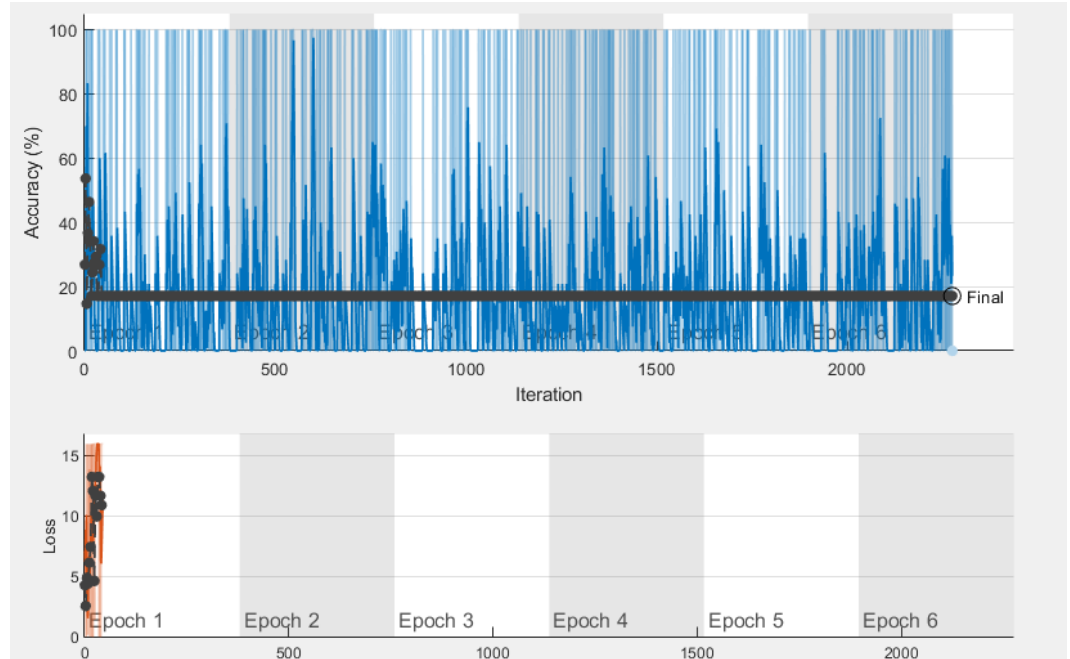


Figure 137 AlexNet Training Results for Facial Hair Classification

II. Models Training using GoogLeNet

To gain accurate results on classification of gender and face type, another deep neural network model GoogLeNet was used. GoogLeNet is quite complex than AlexNet in term of number of layers and diversity of activation functions. After the training on same data and hyperparameter tuning, it is concluded that GoogLeNet outperforms AlexNet in both classification in term of accuracy, however the training time of GoogLeNet is quite higher than AlexNet even on GPU due to complex architecture which can be a disadvantage of GoogLeNet over AlexNet. GoogLeNet model was also trained in MATLAB by utilizing the Deep Learning Toolbox. As discussed in the System Design Chapter, dataset was carefully prepared for the genders. The dataset is divided into training, validation, and testing sets in this training as well.

As transfer learning is performed using GoogLeNet on the dataset so the perimeter of convolution base is set to non-trainable, and a feed-forward network is attached on the convolution base of GoogLeNet to perform classification. The number of neurons in output layer depends upon the number of classes which are two in case of gender classification and four in case of face type classification. The same data augmentation

steps were considered for both GoogLeNet and AlexNet, which includes image rotation with different degrees, reflection, and basic image processing tasks.

GoogleNet has total 27 convolutional layers which are spread across the Inception modules and other layers. Every Inception module has multiple parallel convolutional layers with diverse filter sizes. For dimensionality reduction, modules have pooling layers and 1x1 convolution layers. The first convolutional layer in the GoogLeNet architecture has 64 filters, second convolutional layer has 64 filters, and the third layer has 192 filters. The model has 22 layers, including two fully connected layers, 27 pooling layers, three convolutional layers and 9 inception modules. There are 2 fully connected layers with 4096 neurons.

(A) Gender Classification through GoogLeNet

Just like AlexNet, same 5000 images per class were used for training and validation. *Mini batches* were set to 10, optimizer is stochastic gradient decent momentum. *L2 regularization* is applied with value of 0.0001 to decrease overfitting. Momentum was set to 0.9 and learning rate is 0.01. To validate model augmented validation is used with *frequency of 3* and *validation split of 0.7*.

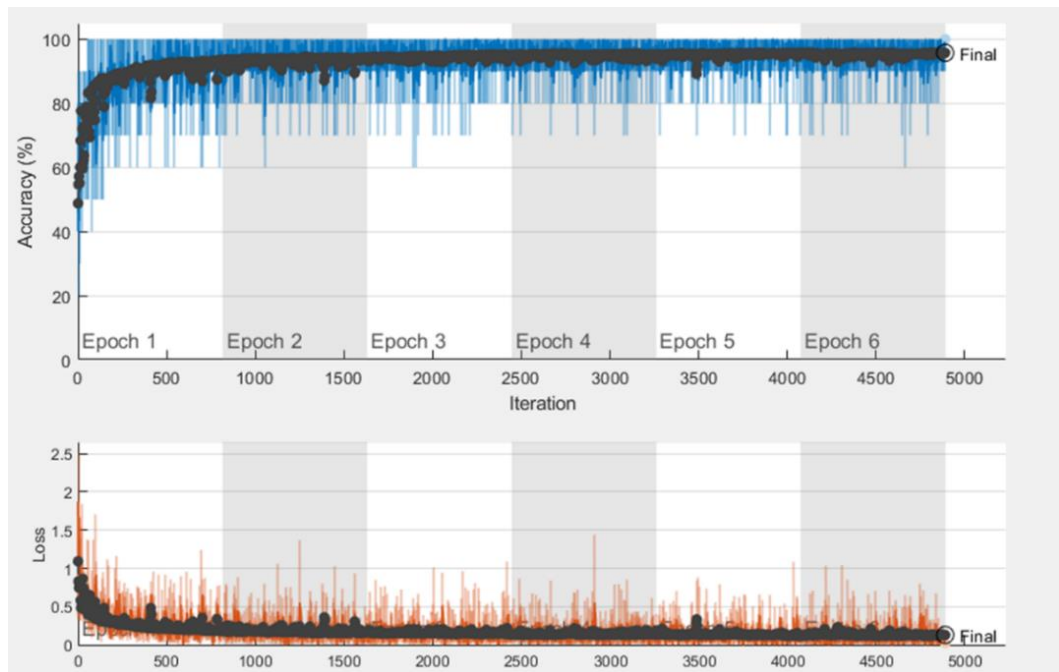


Figure 138 GoogLeNet Gender Classification Training Process

Figure 138 represents the training scenario of GoogLeNet for gender classification using MATLAB. **95.76%** validation accuracy was achieved.

(B) Face Type Classification GoogLeNet

Same 2000 images per class were utilised for training and validation, like AlexNet. Optimizer is stochastic gradient decent momentum. *Mini batches* were set at 10. To reduce overfitting, *L2 regularisation* is used with a value of *0.0001*. The learning rate was set to *0.01*, and the momentum to *0.9*. With a frequency of 3 and a validation split of 0.7, enhanced validation is employed to validate the model.

The Figure 139 shows the training results of face type classification problem using GoogLeNet in MATLAB as a pre-trained network for transfer learning. The validation accuracy approach to **91.18%**.

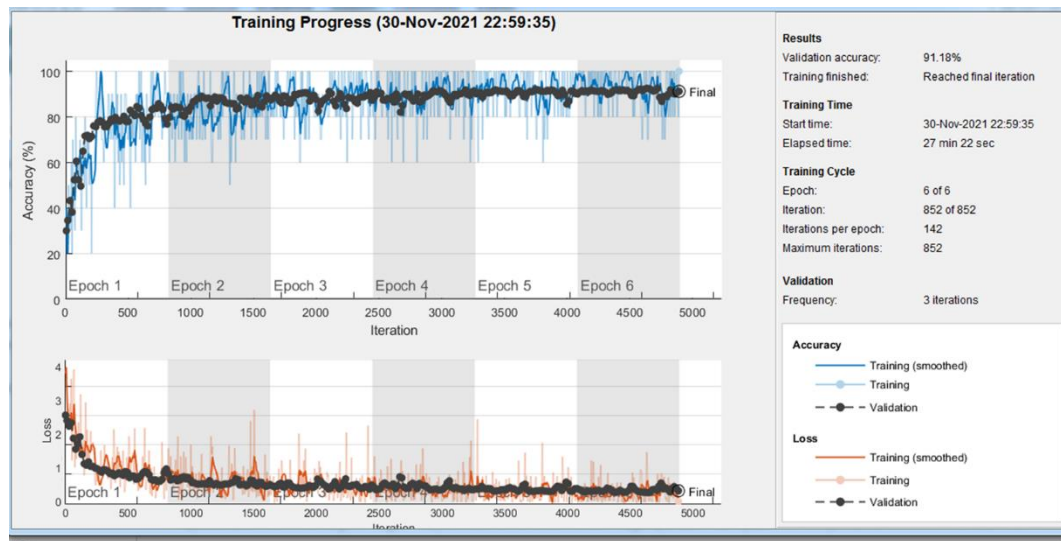


Figure 139 GoogLeNet Face Type Classification

(C) Facial Hair Type Classification GoogLeNet

With 2000 images per class were utilised for training and validation. Optimizer is *stochastic gradient decent momentum*. *Mini batches* were set at 16 and number of epochs were set to 10. To reduce overfitting, *L2 regularisation* is used with a value of *0.0001*. The learning rate was set to *0.01*, and the momentum to *0.9*.

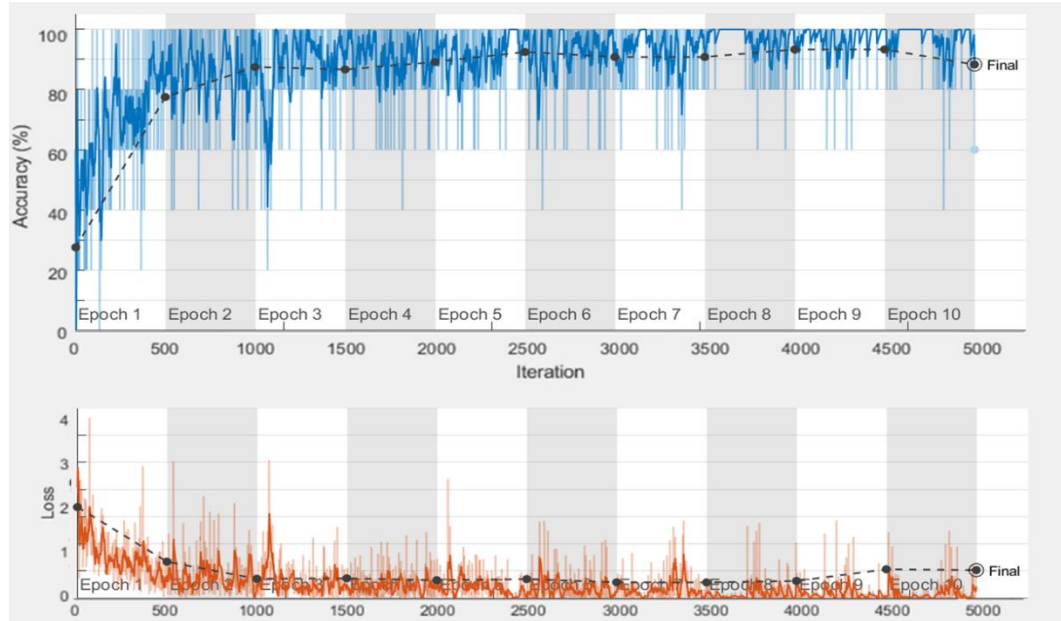


Figure 140 GoogLeNet Training Results for Facial Hair Classification

The Figure 140 shows the training results of facial hair classification problem using GoogLeNet in MATLAB as a pre-trained network for transfer learning. After epoch 9 the validation shows the decreasing trend and the final validation accuracy achieved is **87.7%**.

III. Models Training using VGG-Face

A well-liked deep learning model for facial recognition applications is the VGG Face model. MATLAB does not yet have the VGG Face model but does have the VGG 16 model, which was a problem. The VGG Face model can still be utilised in MATLAB, though. Converting the pre-trained VGG Face model's weights to a MATLAB-compatible format is one technique to use the VGG Face model in MATLAB. A tool like the Keras to MATLAB Converter can be used for this. The pre-trained weights of the VGG Face model can be converted using this tool and then imported into MATLAB. It was performed in the experimentation phase. The VGG Face model weights were loaded into MATLAB and used them to carry out face recognition tasks after converting them to the feature matrix format. The pre-trained weights of the VGG Face model were used to construct a neural network model using MATLAB's deep learning framework. It was done by loading the VGG Face model's pre-trained weights. The VGG Face model's pre-trained weights are then applied to the top 30 layers of a new neural network those were defined. To create the final

output, fully connected layers and then dropout layers were added to the network. Finally, the classification layer was added. It was a lengthy and slow process. The simple solution was to train model Python and a deep learning framework, TensorFlow and Keras, the VGG-Face model was used for facial recognition tasks. The pre-trained VGG-Face model weights and architecture were downloaded. The Keras library, with pre-trained version of the VGG-Face model, was used to accomplish this.

The output of convolution base is a feature vector of length 4096. Which are the extracted features of each face. These features vectors are pushed to feed forward network for classification task. In this research, the case is, the feed forward network is composed on one flatten layers and three hidden layers of 512, 256 and 128 neurons. The last layer is classification layer in which number of neurons depends upon the number of classification categories that is two for gender and four for face type. Moreover, the activation functions which is used in the layers are leaky-ReLU and in last layer the activation function is SoftMax function, which provide probability distribution for classification task. After training the same data which was trained on AlexNet and GoogLeNet with same parameter of validation and data augmentation, it has been found that VGG-Face outperforms all the other pre-trained DNN-models. Results are shown in the results section.

(A) Gender Classification through VGG-Face

To train a VGG Face model for gender classification using 8000 images per gender, the *batch size* is set to 32 adjusted as per the hardware. Learning rate is set to 0.001. Number of epochs are 12. *Binary cross-entropy* is used as Loss function. Optimizer used for the training was *Stochastic Gradient Descent*. Augmentation with *random cropping*, *flipping* and *rotation* were utilized. *L2 regularization* and *dropout layer* with 0.2 value is used.

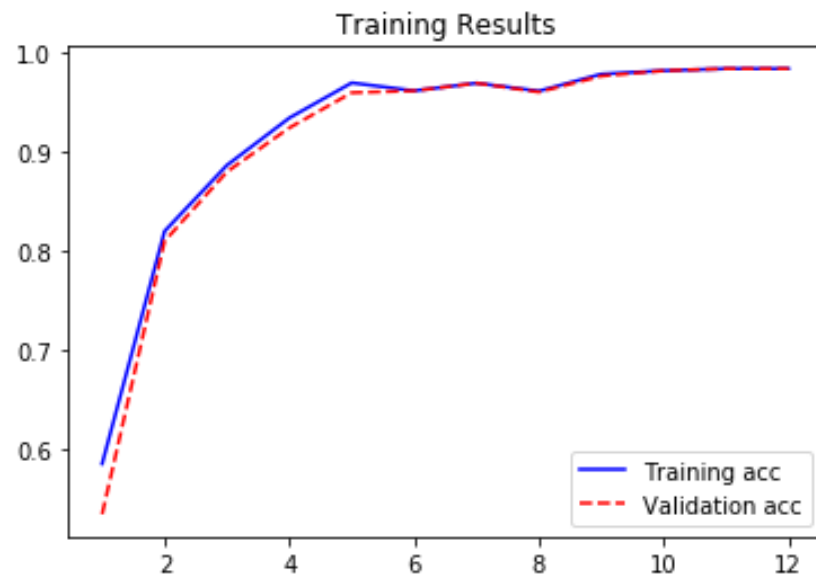


Figure 141 VGG-Face – Gender Classification Training Accuracy & Validation Loss

Figure 141 shows the training and validation accuracy curves. After 10 epochs the training and validation accuracy shows constant behaviour and reached to **98.5%**.

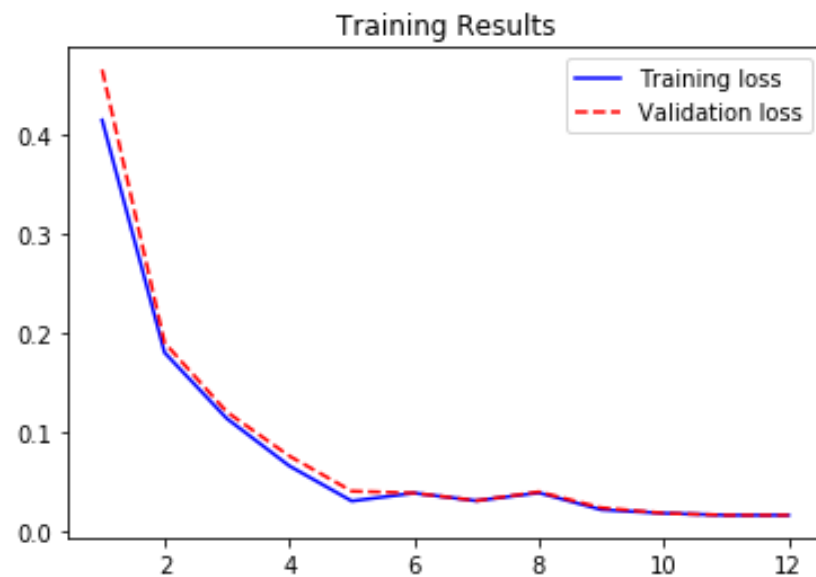


Figure 142 Gender classification Training & Validation loss

Training and validation loss is represented in Figure 142. Like accuracy, Training and Validation Loss decreases and reaches minimum after epoch 10.

(B) Face Type Classification through VGG-Face

VGG Face was also used for face type classification of four ethnicities. Model was trained using 2000 images per face type for training and validation, the *batch size* is set to 32. Number of epochs are 12. Learning rate was set to 0.001. *Categorical cross-entropy* is used as Loss function. Optimizer used for the training was *Stochastic Gradient Descent*. Augmentation with *random cropping*, *flipping* and *rotation* were utilized. *L2 regularization* and *dropout layer* with 0.2 value is used.

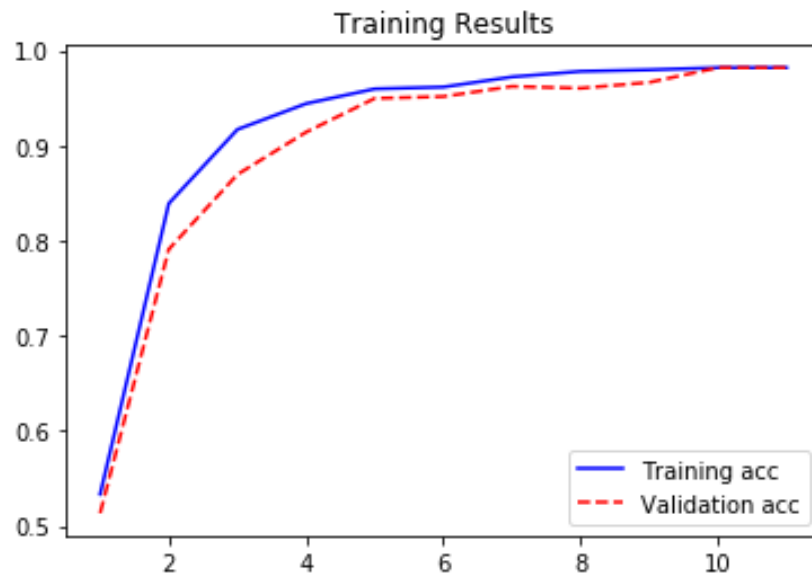


Figure 143 Face Type Classification Training and Validation Accuracy using VGG-Face

Figure 143 represent the training and validation accuracy. Training was terminated after epoch 11 as it shows constant behaviour after reaching to **98.2 %**. Whereas Figure 144 represents the training and validation loss. One thing can be noted that after 10 epochs the training accuracy matched with validation accuracy showing no overfitting in training. This is due to use of regularization and dropout layer with 0.2 value in classifier network.

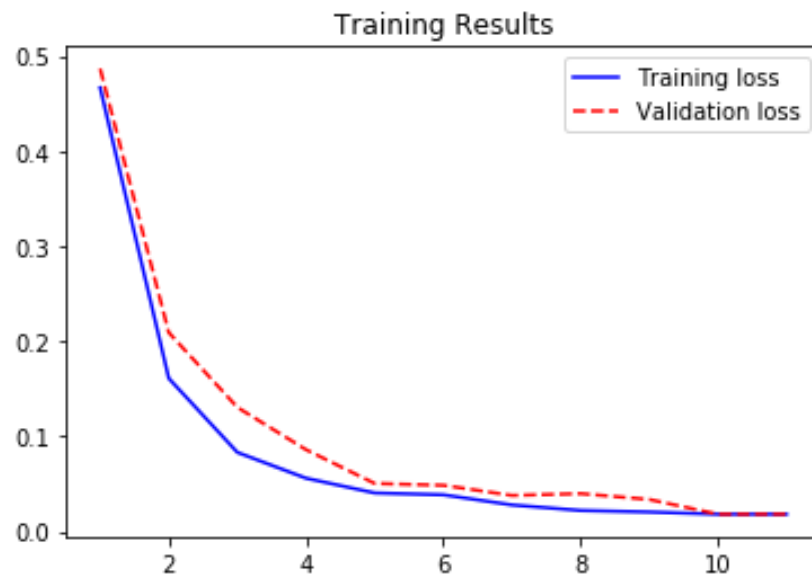


Figure 144 VGG-Face Face Type Classification Training and Validation Loss

IV. Model Training using Custom Neural Network

As for facial hair classification, transfer learning through Google-net and Alex-net didn't give dependable results. Therefore, convolution network from scratch was designed. The network has 33 layers with 3,500,004 trainable parameters. Five convolution layers have been used followed by batch and dropout layers. The last four layers are used for classification task SoftMax function in output layer. Detailed network architecture is given in Figure 145.

Layer (type)	Output Shape	Param #
conv2d_12 (Conv2D)	(None, 224, 224, 32)	896
max_pooling2d_8 (MaxPooling2D)	(None, 112, 112, 32)	0
conv2d_13 (Conv2D)	(None, 112, 112, 32)	9248
batch_normalization_12 (Batch Normalization)	(None, 112, 112, 32)	128
max_pooling2d_9 (MaxPooling2D)	(None, 56, 56, 32)	0
dropout_8 (Dropout)	(None, 56, 56, 32)	0
conv2d_14 (Conv2D)	(None, 56, 56, 64)	18496
batch_normalization_13 (Batch Normalization)	(None, 56, 56, 64)	256
conv2d_15 (Conv2D)	(None, 56, 56, 64)	36928
batch_normalization_14 (Batch Normalization)	(None, 56, 56, 64)	256
max_pooling2d_10 (MaxPooling2D)	(None, 28, 28, 64)	0
dropout_9 (Dropout)	(None, 28, 28, 64)	0
conv2d_16 (Conv2D)	(None, 28, 28, 128)	73856
batch_normalization_15 (Batch Normalization)	(None, 28, 28, 128)	512
conv2d_17 (Conv2D)	(None, 28, 28, 128)	147584
batch_normalization_16 (Batch Normalization)	(None, 28, 28, 128)	512
max_pooling2d_11 (MaxPooling2D)	(None, 14, 14, 128)	0
dropout_10 (Dropout)	(None, 14, 14, 128)	0
flatten_2 (Flatten)	(None, 25088)	0
dense_4 (Dense)	(None, 128)	3211392
batch_normalization_17 (Batch Normalization)	(None, 128)	512
dropout_11 (Dropout)	(None, 128)	0
dense_5 (Dense)	(None, 4)	516
=====		
Total params: 3,581,092		
Trainable params: 3,580,084		

Figure 145 Custom Network Architecture for Facial Hair Type Classification

Convolutional, pooling, and dense layers are among the many layers that make up a CNN. These layers learn from the input images and extract key features. It's relatively simple to implement a unique CNN in Python using the Keras API than MATLAB. Required libraries first, such as Keras and the layers module from TensorFlow were imported. Then, using the Sequential model, which enabled to design a linear stack of layers, defines the CNN architecture. Conv2D layer with the proper number of filters and kernel size were in the first layer in the architectural definition. To add non-

linearity, a ReLU activation function was included. Utilizing the MaxPooling2D layer with the proper pool size, max pooling layer was added after the convolutional layer.

After the convolutional and pooling layers, flatten layer was used to smooth out the output to apply dense layers. Fully interconnected layers that use the ReLU activation function make up the dense layers. The last dense layer utilizes the softmax activation function to display probabilities for each class. Loss function, optimizer, and evaluation metric were set before compiling the model. The model is then trained using the fit technique after being loaded, preprocessed, and the number of epochs, batch size, and validation split being specified. CNN's architecture and hyperparameters were carefully selected based on the current situation. The performance of the model could be significantly impacted by factors like the number of filters, kernel size, and pool size.

(A) Facial Hair Type Classification

For training the model to classify four types of facial hair types, total of 2000 images per facial hair style were used. These images are divided into 80%-20% portions for training and validation respectively. Images are fed to the network in 32 *batches*. *Categorical cross-entropy*, Loss function was used in the training. Learning rate was *0.001*. Optimizer applied for the training was *Stochastic Gradient Descent*. Augmentation with *random cropping*, *flipping* and *rotation* were employed. *Dropout regularization* was applied to prevent overfitting. Number of epochs were 12,

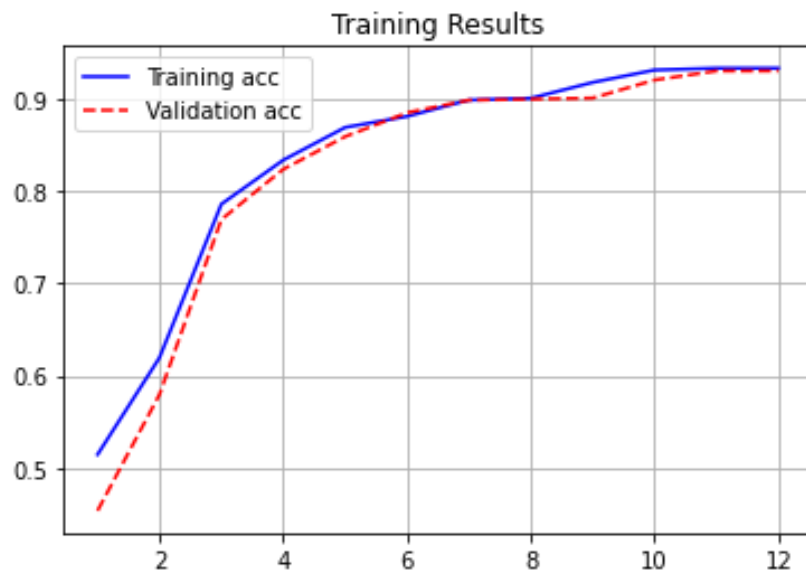


Figure 146 Training & Validation Accuracy Graph using Custom-Network for Facial Hair Type Classification

Training accuracy and validation accuracy results are given in Figure 146, shows that there is no overfitting. The reason behind is the use of dropout layers and L2 regularizations in layers which proved to be useful. Validation accuracy of **96.1%** was achieved after 12 epochs. Training and validation loss curve is represented in Figure 147.

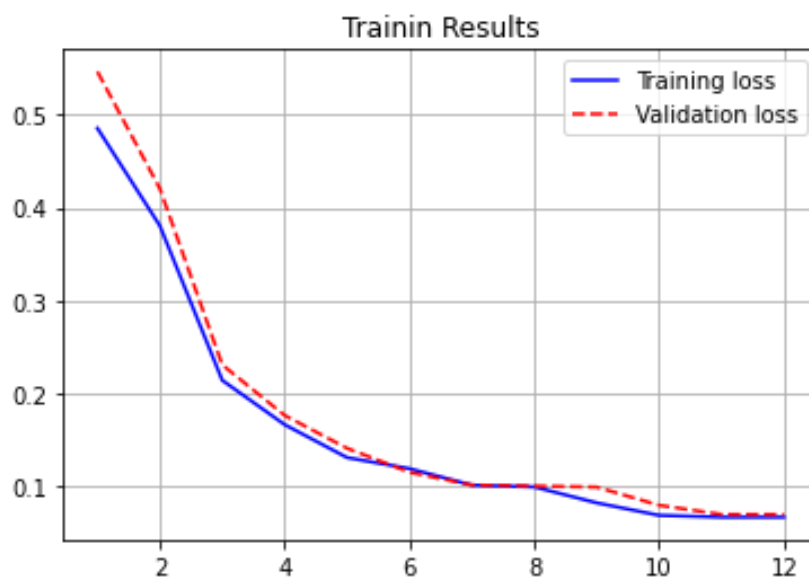


Figure 147 Training & Validation Loss Graph using Custom-Network for Facial Hair Type Classification

V. Model Training using ResNet-50

A deep neural network ResNet50 has demonstrated excellent performance in image classification tests. It is an adaptation of the ResNet design, which is renowned for its capacity to train extremely deep neural networks. The ResNet-50 architecture is a deep and intricate model because it has 50 layers. When using a unique dataset to train the model, pre-trained weights are accessible. For the classification of facial hair styles, Python and Keras are used to train a ResNet-50 classification model that is easier method for accurately classifying images. Data loading and preparation were the first steps in training the ResNet50 model for classification. Keras *ImageDataGenerator* class was used to load images from the folders and perform adjustments like resizing and data augmentation. ResNet50 model was trained after loading and preparing the data. The Keras applications module, which offers a pre-built ResNet50 model, was also used. The model training metrics, optimizer, and loss function were specified. The model was trained using the fit approach. The number of epochs were set, each with several steps, during training. The percentage of training and validation dataset was set on which model makes predictions at each step and modify its weights in response to the discrepancy between the predicted and actual labels. To balance accuracy and training duration, batch size and epoch count were tuned. Lastly, trained model's performance was verified on a test dataset. The evaluation method, which returns the model's accuracy and loss on the validation set was displayed to look at the model performance during training. Pre-trained weights of ResNet50 were used, which were available for download on the Keras website. Using pre-trained weights allows to drastically cut down training time while maintaining accuracy.

(A) Facial Hair Type Classification through ResNet-50

The feedforward network is attached on convolution base of ResNet with 4 dense layers with neurons of 512, 256, 128 and 4 at output layer with *softmax activation* function. *ReLU activation function* is in each dense layer except the last one. Furthermore, for training 2000 images per class, 80% images were kept for training and 20% for validation. Optimizer used was *stochastic gradient decent momentum*. *Mini batches* were set at 32 and number of epochs were set to 20. To reduce overfitting

effect *L2 regularization* and dropout layer is used with value of 0.2. The learning rate was set to 0.01, and the momentum to 0.9.

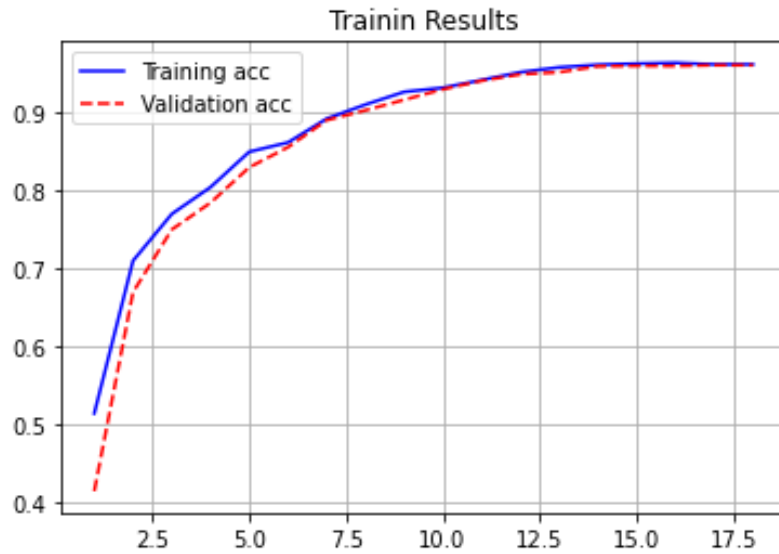


Figure 148 Training and Validation Curve for Facial Hair Classification using ResNet-50

The Figure 148 shows the training accuracy and validation accuracy for facial hair classification problem using ResNet50 as a pre-trained network for transfer learning. The accuracy gets stabilizes after the 15th epoch and validation accuracy approached to **97.3%** with the well-tuned hyper-parameter settings. Whereas Figure 149 represents the training and validation loss per epoch. Loss values reaches to minimum value after 15th Epoch.

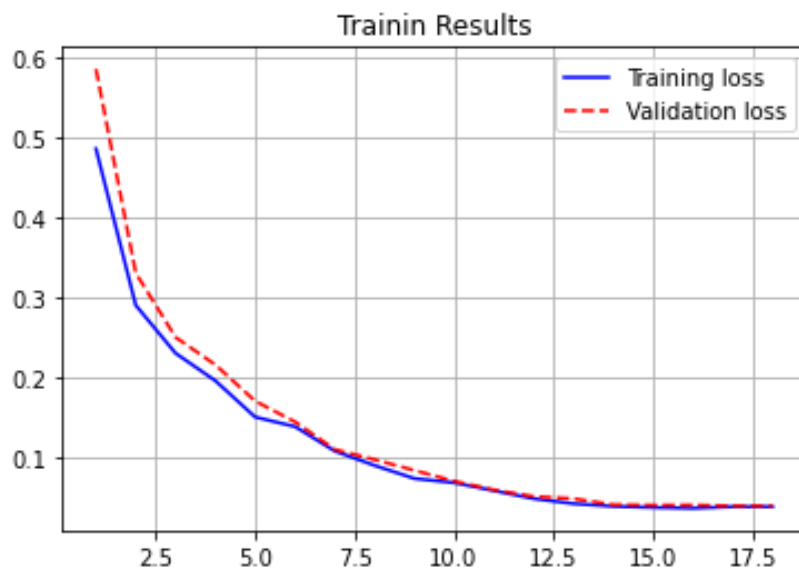


Figure 149 Training Loss and Validation Loss Curve for Facial Hair Classification using ResNet-50

5.06 Models Training for Image Transformation

A popular deep learning method called Multistage Cycle GAN was utilised for resolution enhancement and colorization/ image-to-image translation problems. In order to learn the mapping between two domains, two adversarial networks that cooperate must first be trained. In this research, Multistage Cycle GAN model is used to train models in Python using the TensorFlow framework. Task was to translate blur/low resolution images to high resolution images and sketch images to colour images. The most important part was to prepare training datasets for both the problems. Face images dataset used for the face classification was used for this purpose. Images were resized to 256 x 256 and same images were converted to sketch like images and blur images using Python. To convert a colour image to a sketch-like image and blur in Python, image processing techniques such as edge detection, thresholding, and blurring were used with the help of OpenCV library.

Images were converted to grayscale using the *cv2.cvtColor* function, then Gaussian blur was applied to the grayscale images using the *cv2.GaussianBlur* function to reduce noise. Adaptive thresholding was applied to the blurred images using the *cv2.adaptiveThreshold* function to create a binary image. Finally, binary images were inverted using the *cv2.bitwise_not* function to create the sketch-like image. For blur image, last two steps are omitted. For training, pixel values normalized by setting between -1 and 1.

Two generators, G1 and G2, and two discriminators, D1 and D2, make up the Multistage Cycle GAN model. While the discriminators make the distinction between authentic and false images, the generators translate images from one domain to another. In this instance, G1 will convert sketch images into colour images, and G2 will convert colour images to sketch images. And for resolution enhancement same is done for low resolution/blur images and high-resolution images. For training the model, generators and discriminators were initialized. Loss functions for the generators and discriminators were defined. Model is trained in multiple stages, with each stage consisting of several epochs of training. During each stage, the generators and discriminators are updated using backpropagation and gradient descent. Discriminator is responsible for getting the input image. Size of the input image is 256x256 whereas the 30x30 is the size of

the output tensor. Classification result of the 70x70 portion of the input image is held at every neuron. The value indicated by the discriminator is actually the classification result. Details of the layers and generator architecture is given in Figure 150.

Layer	Activation size
Input	3x256x256
64x4x4 conv, stride 2, pad 1	64x128x128
128x4x4 conv, stride 2, pad 1	128x64x64
256x4x4 conv, stride 2, pad 1	256x32x32
512x4x4 conv, stride 1, pad 1	512x31x31
1x4x4 conv, stride 1, pad 1	1x30x30

Generator Architecture	
Layer Activation	Size
Input	3x256x256
64x7x7 conv, stride 1, pad 3	64x256x256
128x3x3 conv, stride 2, pad 1	128x128x128
256x3x3 conv, stride 2, pad 1	256x64x64
256x3x3 conv, stride 2, pad 1	256x64x64
9 consecutive Residual Blocks	256x64x64
128x3x3 cony Transpose, stride 2, pad 1, out_pad 1	128x128x128
64x3x3 cony Transpose, stride 2, pad 1, out_pad 1	64x256x256
3x7x7 conv, stride 1, pad 3	3x256x256

Figure 150 MS CGAN Layer and Generator Architecture

Training of model is most important part of neural network. Output of any neural network depends upon these factors like architecture, hyper parameters, and correct dataset. Therefore, to train the model firstly very precise and well-ordered dataset was gathered because one of the major problems with deep learning is the requirement of large dataset and pre-processing is also important part of this training process. Being most user friendly and power language Python was used for developing the model.

I. Sketch to Colour Image Transformation

There are various phases involved in training an MS CGAN (Multi-Stage Cycle Generative Adversarial Network) model in Python. First, the 1400 sketch images in the collection were prepared and pre-processed, it includes scaling and normalisation, along with their matching colour images. The MS CGAN model was developed using a deep learning framework, TensorFlow. A generator network and a discriminator network make up the model architecture.

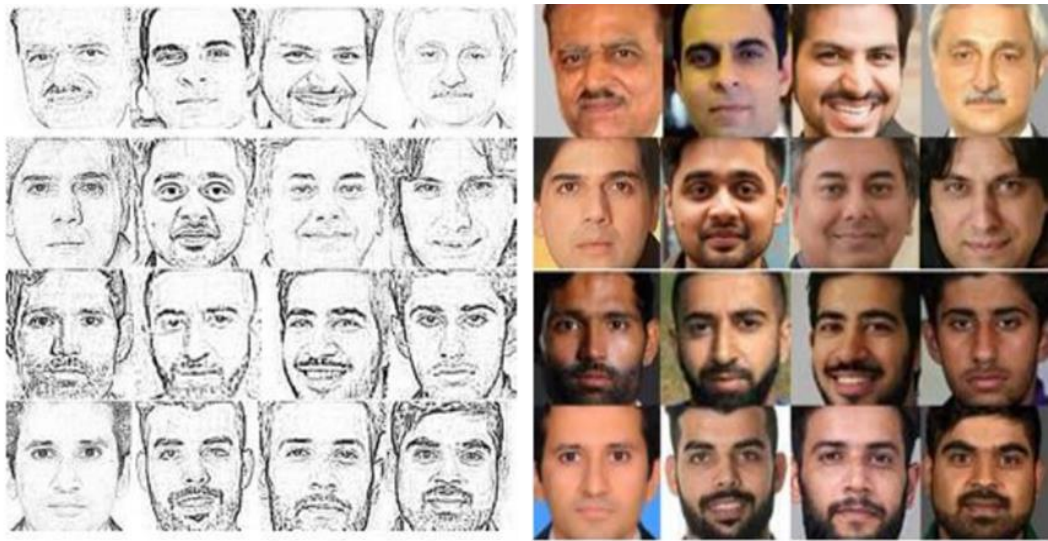


Figure 151 Input images X1 & X2 for MS- Cycle GAN

The generator uses the sketch picture as input to produce a colour image, and the discriminator assesses the created image's quality in relation to the original colour image, sketch and colour images are shown in Figure 151. Using gradient descent and backpropagation, the loss function is optimised. In order to reduce loss and enhance the quality of the generated colour images, the model's parameters are adjusted during the training phase by repeatedly iterating over the dataset. Figure 152 represents the generator and discriminator loss curve. It shows that as reaches to 50th epoch, values

for loss for both generator and discriminator were decreased. The final value for the generator loss was found to be **0.59** and for discriminator the value for loss is **0.4**.

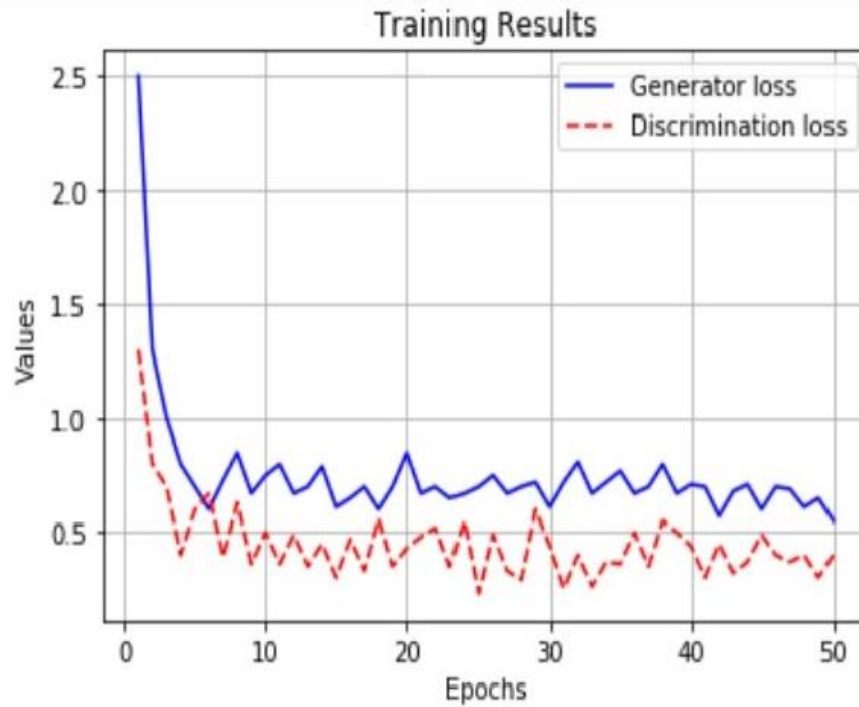


Figure 152 Generator and Discriminator Loss for MS Cycle GAN for Sketch to Color Image

II. Low to High Resolution Conversion

To get best optimal results in detection and recognition, an additional feature of low to high resolution of image feature has been added. Several traditional image pre-processing steps are normally used. However, here Cycle-generative adversarial network was used just like the sketch to hyper-real image method. The results of Cycle-GAN to achieve super resolution are much better than conventional image processing techniques. However, training time for this task was very high and need pair data of low- and high-resolution images. Furthermore, model size also has memory constraints, but prediction time is low. The training results and process has been shared Results section. The configuration of Generative network and discriminator is same as for above network which has been used for sketch to real image conversion. Sample training data is shown in Figure 153.

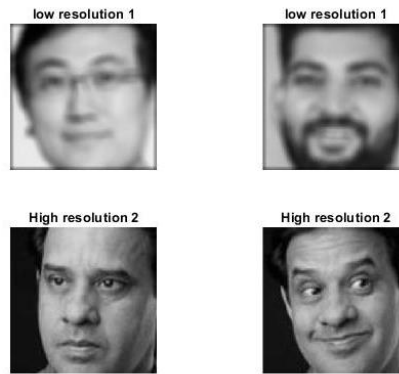


Figure 153 Unpaired Low to High Resolution Dataset

Figure 154 represents the generator and discriminator loss curve. It shows that as epoch reaches its maximum value, the loss for both generator and discriminator decreased. The final value for the generator loss was found to be **0.8** and for discriminator it was **0.1**.



Figure 154 Training Results for Low to High Resolution Image

III. Face Alignment

In facial recognition, the main challenge in recognition is face pose. Different pose of face of same person are quite different from each other. For an example, the face of side pose face image and frontal face image of same person are quite different from each other and challenging in face recognition step. For accurate results in facial recognition, face alignment was performed – frontalization on image. Face image that is captured from CCTV camera is normally in front face. Therefore, the need of facial alignment is used to regenerate an image for the non-frontal face image of the target present in the library.

A supervised learning system called the Supervised Descent Method (SDM) was used to recognise facial landmarks. It trained on a set of regressors using a labelled training dataset made up of pictures with labelled face landmarks. These images are further divided into thirteen poses and twenty lighting conditions. Every regressor is engineered to improve the precision of landmark predictions produced by the step that came before it. To be more precise, a specific regressor is trained for every face landmark, frequently with the use of decision trees model. These regressors acquire the ability to calculate the amount of adjustment required to move the current landmark position closer to the location of the ground truth.

The trained models are arranged in a sequential pattern similar to a classifier's steps in the cascade of regressors. Every regressor in the cascade improves on the placements of the facial landmarks, building on the work of the one before it. The key to SDM's efficacy is its iterative refining process, which enables the steady improvement of landmark accuracy. To make sure the algorithm converges to a final, extremely accurate set of landmark positions, the refinement process is carried out either for a certain number of stages or until a predefined stopping criterion is satisfied.

Model is evaluated on multi-Pie dataset that contains face images with varied poses, lighting, and facial expressions. This dataset has 0.75 million face images of more than 300 people. These images are further divided into thirteen poses and twenty lighting conditions. The training iterations and with loss graph is shown in Figure 155.

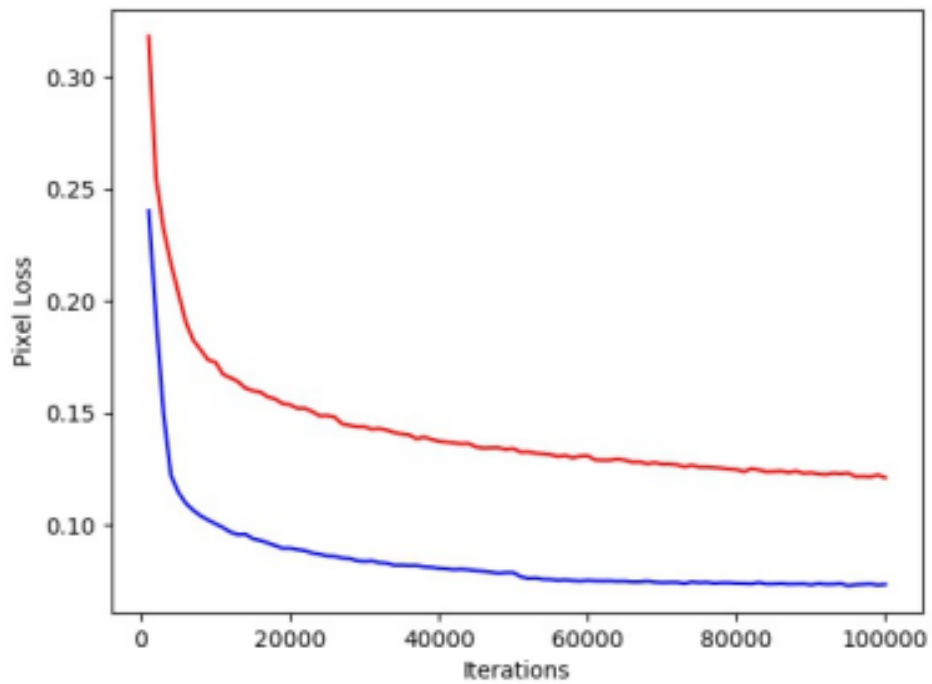


Figure 155 Pixel Loss vs Iteration Graph

5.07 Experimental Results

Following shows the results of the training sessions of each type of model.

I. Face Detection

YOLO v3 model was trained with a dataset of **10,000** training images by keeping learning rate of 0.001. For training, number of batches were 64 and 16 subdivisions. There was total 60 epochs, with 157 number of batches per epoch. Making total of 37680 iterations and 628 iterations per epoch. The YOLO v3 model was trained using the Darknet framework, with stochastic gradient descent optimization. Step decay strategy is used for learning rate.

II. Gender Classification

Total of **16000** training images were used for training model for gender classification, after 6 epochs, AlexNet reached maximum validation accuracy of **95.79%**, whereas GoogLeNet reaches to **95.76%** after same number of epochs. Loss and Accuracies became constant after 6 epochs. VGG Face achieved maximum validation accuracy of **98.5%** after 10 epochs as summarized in Table 10.

Table 10 Gender Classification Training Results

Model	Training Data	Validation Accuracy
AlexNet	16,000 images	95.79%
GoogLeNet	16,000 images	95.76%
VGG Face	16,000 images	98.50%

III. Face Type Classification

2000 images per class were used for training face type classification model. AlexNet achieved validation accuracy of **89%** for the classification of face types. GoogLeNet reaches **91.18%** after the same number of epochs, in contrast. After 6 epochs, loss and accuracy were consistent. After 10 epochs, VGG Face had a maximum validation accuracy of **98.2%**. Summary is given in Table 11.

Table 11 Face Type Classification Training Results

Model	Training Data	Validation Accuracy
AlexNet	2000 per class	89%
GoogLeNet	2000 per class	91.18%
VGG Face	2000 per class	98.20%

IV. Facial Hair Type Classification

As given in Table 12, for facial hair styles classification AlexNet was unable to give any considerable results. GoogLeNet achieved validation accuracy of **87.7%** for the classification of face types. VGG Face is specially trained for extracting facial features that is why it was not evaluated for this type of classification. Customized neural network was trained, and it achieved validation accuracy of **96.1%**. Finally, ResNet-50 was used to train the model and achieved validation accuracy of **97.3%** after 15 epochs.

Table 12 Facial Hair Type Classification Training Results

Model	Training Data	Validation Accuracy
AlexNet	2000 per class	Un-conclusive
GoogLeNet	2000 per class	87.7%.
Custom NN	2000 per class	96.10%
ResNet-50	2000 per class	97.30%

V. Image Transformation

In Table 13, experimentation results for two types of image transformations are displayed. For sketch to colour image transformation, **1400** images were used to train the model using MS-CGAN. The ultimate generator loss value was determined to be **0.59**, and the discriminator loss value is **0.4**. Similar approach was used to train model for resolution enhancement where generator loss was **0.8** and discriminator loss was **0.1**.

Table 13 Image Transformation Models Experimentation Results Summary

Transformation	Training Data	Generator Loss	Discriminator Loss
Sketch to Colour	1400	0.59	0.4
Resolution Enhancement	750	0.8	0.1

5.08 Summary

For the experimentation process computer with 9th generation Core i7 CPU with 6 cores and 12 threads was used in the research, it was installed with 32 GB of RAM and 6 GB RTX 2060 GPU. Raspberry Pi 4 was used as parallel computing platform. For development of models, *Deep Learning Toolbox* along with the *Image Processing and Computer Vision Toolbox* was used in MATLAB 2021. In addition to MATLAB, *Jupyter Notebook* was used to develop models in *Python*. Libraries such as *TensorFlow*, *Keras*, *Tkinter*, *Panda* and *PyTorch* were used in developing deep learning models. *Octave GNU* was used as programming platform for basic image processing tasks in Raspberry Pi. For experimentation Hikvision DS-2CD2085FWD-I, with resolution of 3840×2160 , 50 degrees of FOV, 20 frames per second and 1/1000s shutter rate was mounted on movable channel of iron clamp on pole. Frame rate was decreased to 2 frames per second by the in the software. Working distance was set to 46 feet with varying background with the help of flex sheet. Brightness and saturation were set automatically by the camera with the help of IR sensors. 50 W LED with 10 mm lens were used to vary the lighting conditions. Mirrors and sheets covered by aluminium foils were used for creating replica of reflections created by vehicles. Angular variations were made by the targets to validate the performance of image augmentation performed while training the system. Moving targets were also used to verify the significance of PIR sensor in reducing the workload on the computing devices. Training parameters and experimental results summary is given in Experimental Results Section.

Chapter 6

Results

6.01 Overview

This section shows and explains the results of the experiments performed. It includes the face detection, gender recognition, face type recognition, facial hair classification, low resolution to high resolution conversion results and transformation of sketch to coloured image. Various metrics are used to evaluate artificial intelligence models. Binary classification involves making a prediction for a single binary output class, such as "male" or "female". Multiclass classification makes prediction for one class among multiple classes, such as "face type I", "face type II", "face type III" and "face type IV" [309]. For a classification problem, the quantities True Positive (TP), False Positive (FP), False Negative (FN) and True Negative (TN) are calculated with a specific threshold. TP are the instances where the model predicted the positive class correctly. In FP, the negatives are predicted as true class by the model. FN is opposite of FP where positive instance is classified as negative. TN are the cases in which model predicts the negative class correctly [310]. The line separating a positive or negative classification for an instance is referred to as the threshold in a confusion matrix. It is employed to assess the trade-off between forecasts that are TP and FP. When classifying positive cases, a high threshold indicates that the model is more conservative, whereas a low threshold indicates that the model is more liberal. Using methods such as receiver operating characteristic (ROC) curve analysis, the ideal threshold value can be found [311].

A frequent metric in classification used to assess a model's performance is accuracy. It calculates the percentage of the model's predictions that were accurate and calculated using Eqn. 6-1.

$$Accuracy = \frac{TP + TN}{TP + FN + TN + FP}$$

Eqn. 6-1

In multiclass classification, accuracy is the ratio of correctly classified cases to the total number of cases and found using Eqn. 6-2.

$$Accuracy_{Class\ n} = \frac{TP_{Class\ n} + TN_{Class\ n}}{TP_{Class\ n} + FN_{Class\ n} + TN_{Class\ n} + FP_{Class\ n}} \quad \text{Eqn. 6-2}$$

Another classification metric called precision is employed to assess the precision of the model's positive predictions using Eqn. 6-3.

$$Precision = \frac{TP}{TP + FP} \quad \text{Eqn. 6-3}$$

The percentage of the model's positive predictions that come true is known as precision. For multiclass classification, precision can be calculated for each class separately as Eqn. 6-4.

$$Precision_{Class\ n} = \frac{TP_{Class\ n}}{TP_{Class\ n} + FP_{Class\ n}} \quad \text{Eqn. 6-4}$$

And then averaged over all classes to obtain macro-average precision as given in Eqn. 6-5.

$$Precision_{MACRO\ AVERAGE} = \frac{Precision_1 + Precision_2 \dots + Precision_n}{Number\ of\ Classes} \quad \text{Eqn. 6-5}$$

Recall is a classification metric that measures how well a model can detect positive examples. The proportion of positive events that the model properly identified as positive is measured by recall. Recall is defined as in Eqn. 6-6.

$$Recall = \frac{TP}{TP + FN} \quad \text{Eqn. 6-6}$$

In multiclass classification, recall is the ratio of correctly predicted positive observations to the total actual positive observations and calculated using Eqn. 6-7.

$$Recall_{Class\ n} = \frac{TP_{Class\ n}}{TP_{Class\ n} + FN_{Class\ n}} \quad \text{Eqn. 6-7}$$

And then averaged over all classes to obtain macro-average recall using Eqn. 6-8.

$$Recall_{MACRO\ AVERAGE} = \frac{Recall_1 + Recall_2 \dots + Recall_n}{Number\ of\ Classes}$$

Eqn. 6-8

F1 score is a precision and recall-balanced index of a classifier's accuracy. It is described as the harmonic mean of recall and precision as given in Eqn. 6-9.

$$F1\ Score = \frac{2 \times Precision \times Recall}{Precision + Recall}$$

Eqn. 6-9

The F1 score in multiclass classification is the harmonic mean of recall and precision for a given class. It is calculated using Eqn. 6-10. By accounting for both the quantity of true positive predictions as well as the quantity of false positive and false negative predictions, it strikes a compromise between precision and recall.

$$F1\ Score_{Class\ n} = \frac{2 \times Precision_{Class\ n} \times Recall_{Class\ n}}{Precision_{Class\ n} + Recall_{Class\ n}}$$

Eqn. 6-10

The macro-average of F1 score is calculated using Eqn. 6-11.

$$F1\ Score_{MACRO\ AVERAGE} = \frac{2 \times Precision_{MACRO\ AVERAGE} \times Recall_{MACRO\ AVERAGE}}{Precision_{MACRO\ AVERAGE} + Recall_{MACRO\ AVERAGE}}$$

Eqn. 6-11

In a binary classification problem, specificity measures a classifier's ability to properly identify negative samples. It is calculated by dividing the total of accurate negative predictions by the total of accurate negative and accurate positive forecasts as shown in Eqn. 6-12.

$$Specificity = \frac{TN}{TN + FP}$$

Eqn. 6-12

The average specificity across all classes is typically used to calculate specificity in multiclass classification problems. To accomplish this, the specificity for each class is computed, and the average is calculated as given in Eqn. 6-13.

$$Specificity_{Class\ n} = \frac{TN_{Class\ n}}{TN_{Class\ n} + FP_{Class\ n}}$$

Eqn. 6-13

As an alternative, the specificity scores for each class can be averaged across all binary classification problems to obtain a macro-average specificity [312].

The true positive rate, often referred to as recall, is shown against the false positive rate, which is calculated as one minus the specificity, on the ROC curve. The proportion of positive instances that are accurately classified as such is measured by the true positive rate, whereas the proportion of negative instances that are mistakenly classified as positive is measured by the false positive rate. The ROC curve offers a tool to compare the effectiveness of several classifiers or to choose an appropriate threshold for a classifier, allowing the trade-off between true positive rate and false positive rate to be shown. A one-versus-all strategy can be used to create ROC curves for each class in a multiclass classification problem, and then a macro-average ROC curve can be created by combining the individual class ROC curves. The classifier's total performance is summarised by the macro-average ROC curve by considering how well it performs across all classes [311].

Precision-recall (PR) curves are a means to visualise the trade-off between precision and recall in binary and multiclass classification. A single precision-recall curve that illustrates the relationship between precision and recall when the threshold for categorising an instance as positive is changed can be drawn for binary classification. The curve summarises the classifier's performance across all potential threshold values and makes it feasible to see how recall and accuracy trade off against one another. For multiclass classification, a one-versus-all method can be used to create precision-recall curves for each class, which can then be combined to create a macro-average PR curve. The classifier's overall performance is summarised by the macro-average precision-recall curve by considering how well it performs across all classes. The precision-recall curve offers a technique to evaluate the performance of several classifiers or choose an appropriate threshold for a classifier in both binary and multiclass classification. The area under the PR curve is a scalar summary of a classifier's performance and a gauge of its capacity for high precision and recall at the same time [313].

In the field of computer vision, mAP (mean Average Precision) is a frequently used evaluation metric, notably in object identification and picture classification tasks. It

calculates the average precision for a set of item detections across all feasible recall levels. The precision-recall curve is condensed into a single value by mAP, which represents the average precision over various recall levels. The scale goes from 0 to 1, with 0 reflecting the worst-case scenario and 1 showing flawless precision and recall. Because it offers a thorough assessment of a model's overall performance, taking into account both the precision and recall of the detections, mAP is a well-liked evaluation metric [314].

In computer vision, the IoU (Intersection over Union) assessment measure is frequently employed, especially in tasks involving object detection and image segmentation. It calculates the amount of overlap between the expected and actual regions. The intersection of the anticipated and ground truth regions is compared to the union of these regions to determine the IoU. IoU has a value between 0 and 1, where a score of 1 indicates a perfect match between the predicted and ground truth regions and a score of 0 indicates no overlap. The average IoU across all classes in a multiclass classification or segmentation issue is known as the mean IoU. It is frequently used to compare several models or to keep track of a model's performance while it is being trained. It provides a single value summary of the model's performance for all classes. By adding up the individual IoU values for each class and dividing by the total number of classes, mean IoU is determined. Like other assessment metrics, mean IoU has a scale from 0 to 1, with 0 being the worst-case scenario and 1 representing perfect class separation [315]. Intersection over Union is the relationship between overlap region and union area of the ground truth and predicted boundary box. It is calculated with the help of Eqn. 6-14.

$$IoU = \frac{\text{Intersection Area}}{\text{Union Area}}$$

Eqn. 6-14

As shown in Figure 156, union is the area of both ground truth and predicted boundary boxes and intersection area is the area of the overlap. For a good object detector intersection area should be higher. Maximum value for IoU can be 1 that is an ideal condition.

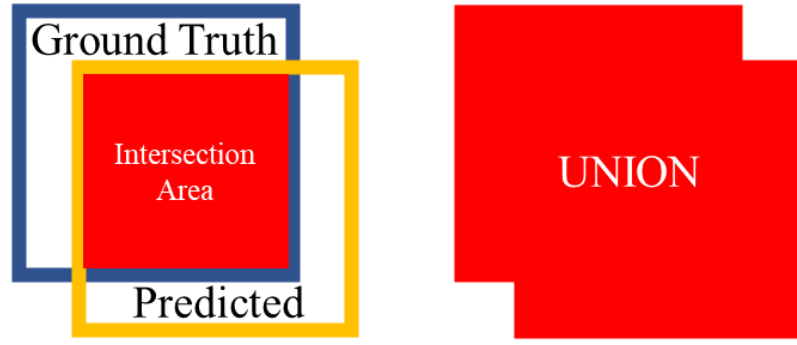


Figure 156 Intersection and Union

Additionally utilised as evaluation metrics in Generative Adversarial Networks are Mean Squared Error (MSE) and Root Mean Squared Error (RMSE). MSE and RMSE are used in the context of GANs to compare the generated data with the real data. The real data is a sample taken from the true data distribution, whereas the generated data is the output of the generator network.

MSE is given by Eqn. 6-15.

$$MSE = \frac{1}{N} \sum_{t=1}^N (Y_t - \hat{Y}_t)^2$$

Eqn. 6-15

where $T = Pixel$, $N = Number\ of\ Pixels$, $Y = Ground\ Truth\ Image$,
 $\hat{Y} = Generated\ Image$

Whereas RMSE is given by Eqn. 6-16.

$$RMSE = \sqrt{MSE}$$

Eqn. 6-16

The distance between the simulated and actual data distributions is calculated using MSE and RMSE. The goal is to reduce the difference in distributions of created and real data, which will improve the quality of the generated data. The quality of the generated data improves with a decreasing MSE or RMSE value. Both MSE and RMSE offer details on the accuracy of the data that are generated, but RMSE may be easier to understand because it offers a statistic in the same units as the data [316].

Cosine similarity method is used to check the similarity between two vectors. In our case images are the vectors of multi-dimensional space. In this method cosine of the angle between the vectors is calculated. It generates value between -1 and 1, where

1 indicates 100% similarity and -1 indicates 0% similarity. Eqn. 6-17 and Eqn. 6-18 are used to calculate the cosine similarity of two images.

$$\text{Cosine Similarity}_{xy} = \frac{x \cdot y}{\|x\| \|y\|} \quad \text{Eqn. 6-17}$$

or

$$\text{Cosine Similarity}_{xy} = \frac{\sum_{i=1}^n x_i y_i}{\sqrt{\sum_{i=1}^n x_i^2} \sqrt{\sum_{i=1}^n y_i^2}} \quad \text{Eqn. 6-18}$$

Training curves are graphical depictions of a machine learning model's performance over time, often as a function of iterations or epochs. These curves offer information on the model's behaviour during training and can be used to identify and avoid over- or underfitting. The training loss and validation loss are plotted as a function of the number of iterations or epochs in the loss curve. This curve is used to track the model's development throughout training and to identify over- or underfitting. Plots training accuracy and validation accuracy as a function of iterations or epochs, or the number of iterations. This curve is used to track the model's performance throughout training and to identify over- or underfitting [317].

6.02 Face Detection in CCTV images

With the help of literature, it was found that the YOLO v3 performs better than its predecessors YOLO and YOLO v2 in terms of accuracy and specially for detecting small images. RCNN, Fast RCNN and Faster RCNN also does not match the performance of YOLO v3 as per the literature. Results for the YOLO v3 based, trained model is displayed in this section. As discussed in earlier section the YOLO v3 model is trained on two types of datasets, the one that is generated by the experimental setup that was used to create replica of street cameras and the other one is *Large Scale Multi-Camera Detection Dataset Kaggle*. This classification involves only involves one class that is face and does not have *True Negative* Values for the dataset. 1000 test images are passed through the model and boundary boxes are extracted along with the

confidence score. These values are used to calculate IoU, Precision and Recall with different threshold. For the calculation of the mean IoU, images with multiple faces were included so that model can be evaluated on large test data. Total 1000 test frame images were used for evaluation of precision, recall and F1 score of the model. Each face in the image is 40 to 60 pixels wide so that the model can be tested in the worst scenarios.

I. Mean IoU

The values of IoU are found for the 4320 faces in 1000 images separately and then Mean IoU is found using Eqn. 6-19.

$$mIoU = \frac{\sum_1^n IoU_n}{n} \quad \text{Eqn. 6-19}$$

$$n = \text{Number of Faces} = 4320$$

Mean IoU for the model is found to be **0.67**.

II. Average Precision & Precision vs Recall Curve

YOLOv3 model was run the test dataset and obtained the predicted bounding boxes and their confidence scores. Confidence thresholds were from 0.1 to 0.9. For each threshold, precision and recall were calculated. *True positive* refers to detected boundary box having IoU greater than 0.5. *False Positive* is the boundary box having IoU less than 0.5 or non-Face is detected as face. *False Negative* is the case when boundary box not detected at all. *True Negative* is not applicable as there is only one class. Recall and Precision which are sensitive to each other and increasing one can cause decreasing to other. Figure 157 shows the Precision vs Recall Curve for the model.

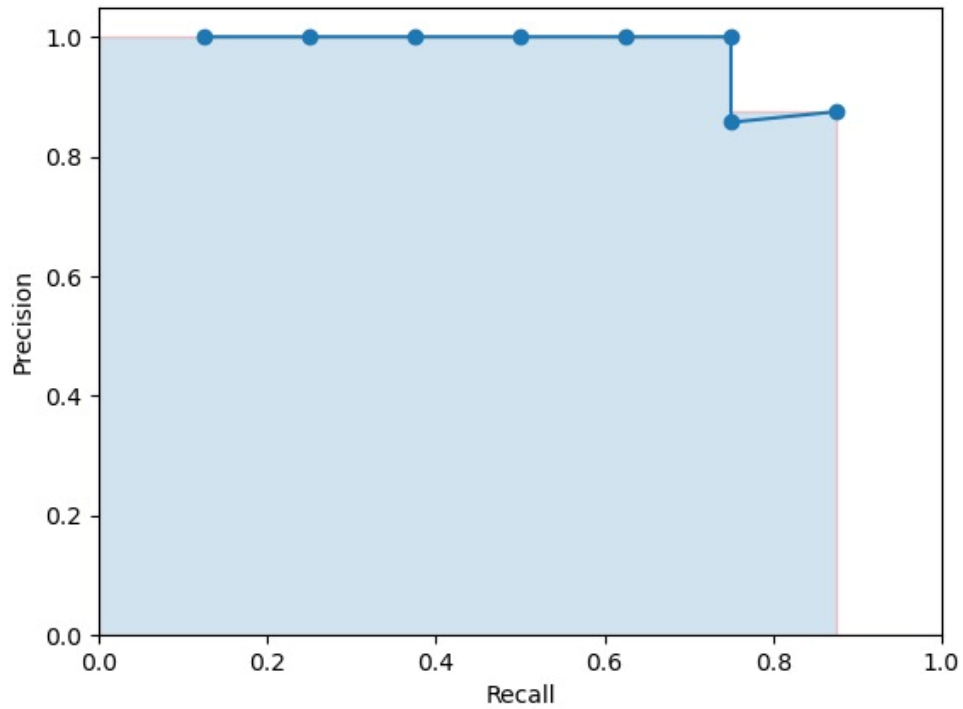


Figure 157 Recall vs Precision for YOLO v3

Average Precision is calculated by calculating area under the curve of precision vs recall curve using Eqn. 6-20. Before calculating the values, the Mean IoU value is set, to consider if the detection is valid or not.

$$AP = \sum_{k=0}^{k=n-1} [Recall(k) - Recall(k + 1)] \times Precision(k)$$

Eqn. 6-20

$Recall(n) = 0, Precision(n) = 1, n = \text{Number of threshold}$

Average Precision for the model is found to be **0.856**.

III. Precision, Recall, Accuracy and F1 Score

Values for precision, recall and F1 score are calculated by testing the model on the test images having 5120 faces with width of 40-60 pixels. Following values are found for considering 50% IoU as threshold.

Using the above values, the four important values are calculated, accuracy, precision, recall and F1 score.

$$TP = 4976$$

$$FN = 144$$

$$FP = 392$$

$$Accuracy = \frac{TP}{TP + FN + FP}$$

Eqn. 6-21

$$\textbf{Accuracy=0.9027}$$

With the above value accuracy calculated using Eqn. 6-21, is referring to a low value of 90.27% but the major contributor for achieving low value is the False Positive values. These values can affect the system by putting additional burden on the classification system but will not affect the final results because the final step for matching the features will eliminate the undesired image.

$$Precision = \frac{TP}{TP + FP}$$

Eqn. 6-22

$$\textbf{Precision=0.9269}$$

As discussed above the False Positives are high that leads to lower precision calculated using Eqn. 6-22.

$$Recall = \frac{TP}{TP + FN}$$

Eqn. 6-23

$$\textbf{Recall=0.9718}$$

Recall gives the value that refers how good model is in detecting the faces from the input images. Simply the results found using Eqn. 6-23, shows that 97.18% faces are detected. From the analysis it was found that model only misses faces less than 45 pixels wide. F1 score is calculated using Eqn. 6-24.

$$F1\ Score = \frac{2 \times Precision \times Recall}{Precision + Recall}$$

Eqn. 6-24

$$F1\ Score = 0.9488$$

IV. Results Summary for Face Detection

Table 10 presents a comprehensive summary of the face detection results achieved through the utilization of the YOLO v3 model on the dataset. The Mean Intersection over Union (IoU) was recorded at 0.67. The Average Precision was calculated to be 0.856. Among the detected faces, a total of 4976 were identified as True Positives (correctly detected faces), while 392 were False Positives (incorrectly identified faces). Additionally, there were 144 False Negatives (missed faces). The Precision, representing the ratio of correctly predicted faces to the total predicted faces, was found to be 0.926, and the Recall, indicating the ratio of correctly predicted faces to the total actual faces, was notably high at 0.971. The overall Accuracy of the face detection was determined to be 90.7%, signifying an impressive level of correctness in the predictions. The F1-Score, a metric that balances Precision and Recall, was calculated to be 0.948, underlining the effectiveness of the YOLO v3 model in accurately detecting faces in the dataset.

Table 14 Results summary for Face Detection

	Mean IoU	Average Precision	True Positive	False Positive	False Negative	Precision	Recall	Accuracy	F1-Score
FACE DETECTION RESULT	0.67	0.856	4976	392	144	0.926	0.971	90.7%	0.948

6.03 Gender Classification

Gender classification was performed by training the dataset on AlexNet, GoogLeNet and VGG-Face. Results are shown below. For training a model initially portion of a dataset was used for the comparison.

I. Gender Classification Results using AlexNet

Training is done on GPU due to high computation and large amount of dataset. NVidia GPU 970 is used to boost training and reduce training time.

(A) ROC and Precision vs Recall Curve

ROC is a plot between true positive rate and false positive rate. True positive rate is also known as recall and sensitivity while false positive rate is known as probability of false alarm. ROC curve for gender classification using AlexNet is given Figure 158 showing the performance of the model and trade-off between true positive and false positive rates with varying the threshold. Precision vs recall curve is represented in Figure 159 illustrate trade-off of the precision and recall on the basis of threshold value. These curves represent the diagnostic ability of the classifier.

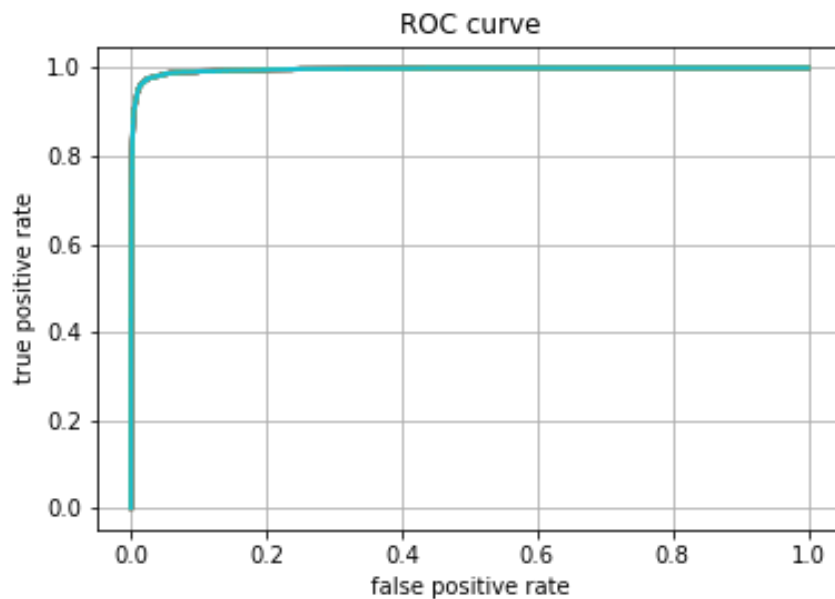


Figure 158 ROC curve AlexNet gender classification

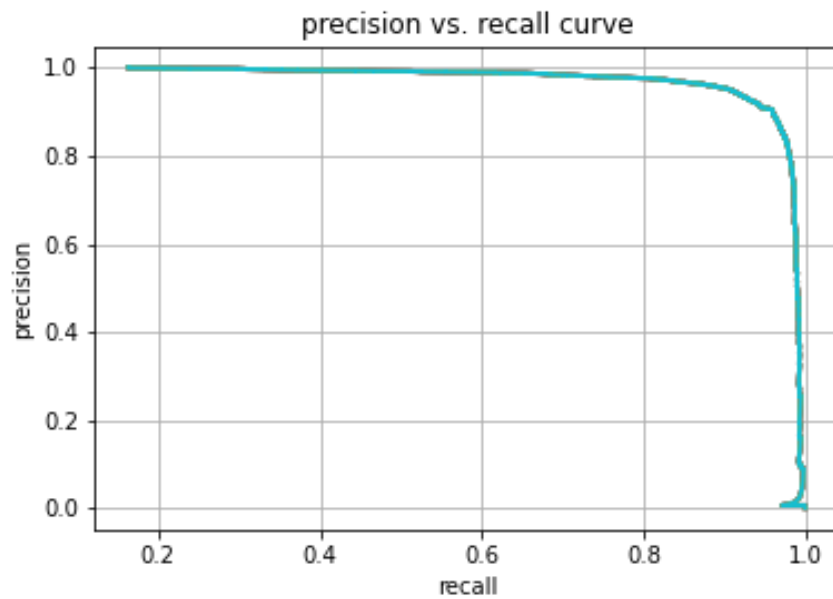


Figure 159 Precision vs Recall Curve

(B) Confusion Matrix

Figure 160 represents the confusion matrix on test data of AlexNet for gender classification problem. This model is tested on 5000 samples of each class.

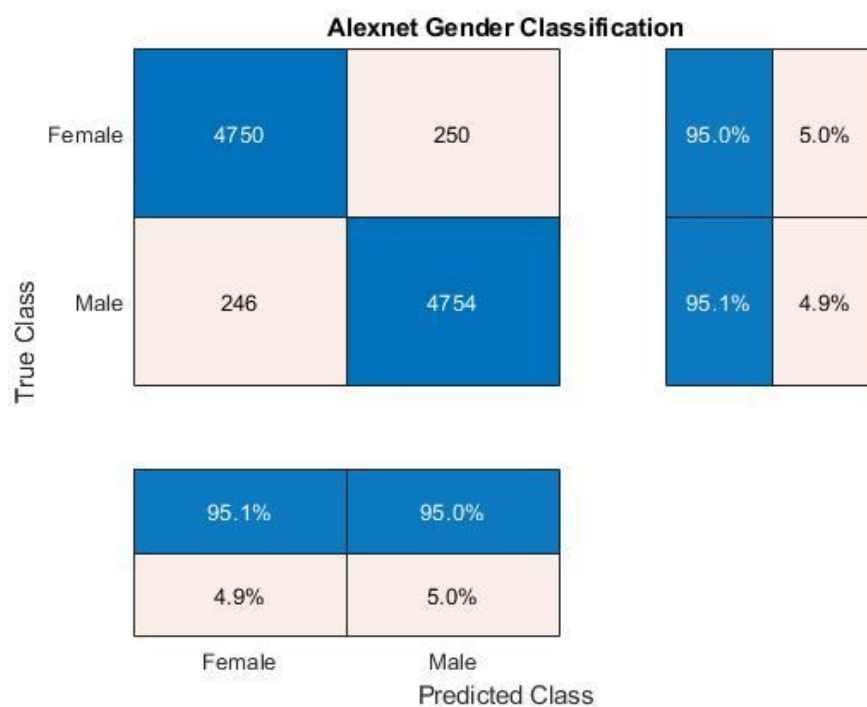


Figure 160 Confusion matrix Alex-net Gender Classification

This model has predicted following results.

$$TP = 4750$$

$$TN = 4754$$

$$FN = 250$$

$$FP = 246$$

(C) Precision, Recall, Accuracy and F1 Score

Values of accuracy, recall, precision, specificity, and F1 Score are follows.

$$Precision = 0.9508$$

$$Accuracy = 0.9504$$

$$Recall = 0.95$$

$$F1\ Score = 0.95$$

$$Specificity = 0.9508$$

II. Gender Classification Results using GoogLeNet

Training of GoogLeNet on gender classification produce almost similar results with AlexNet not in term of accuracy only but also almost same in precision and recall which leads to similar in ROC and F1 score also.

(A) ROC and Precision vs Recall Curve

. The performance of the model and trade-off between true positive and false positive rates with adjusting the threshold are shown in Figure 161 which uses GoogLeNet to classify the genders. Figure 162, precision vs. recall curve depicts the trade-off between precision and recall based on a threshold value. These curves show how well this classifier can diagnose problems. It can be noted that both of these curves depends upon the probability of predicted classes and from those threshold is calculated.

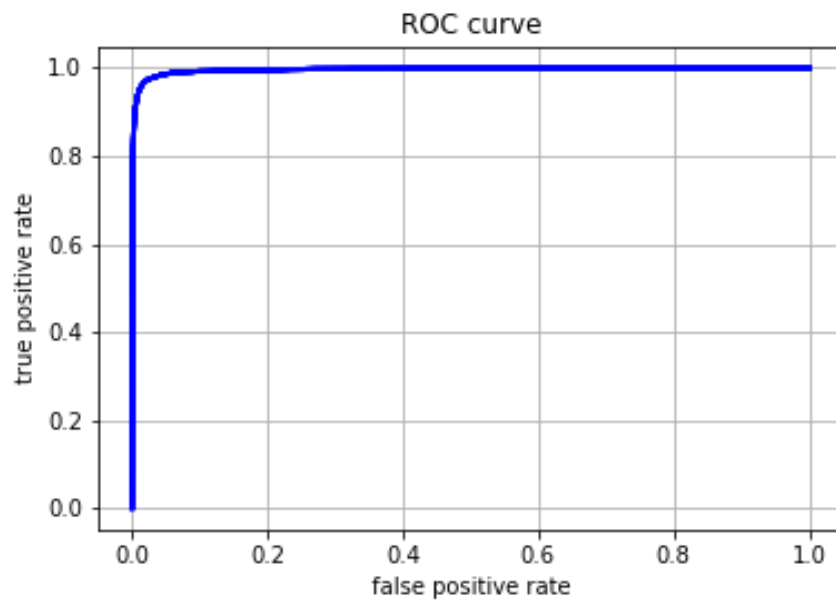


Figure 161 ROC Curve- GoogLeNet for Gender

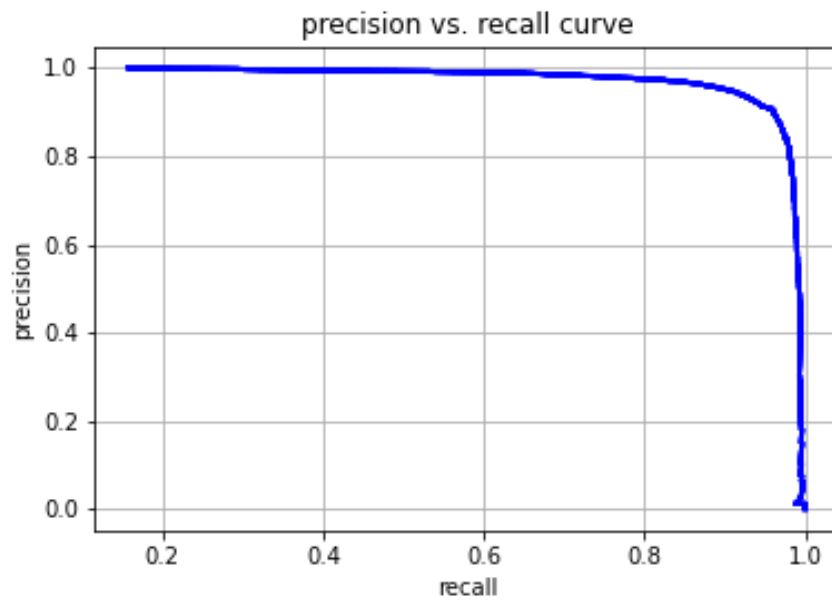


Figure 162 Precision vs Recall Gender Classification GoogLeNet

(B) Confusion Matrix

Figure 163 represents confusion matrix of GoogLeNet for gender classification. Same 5000 test images are used for evaluating the GoogLeNet based model.

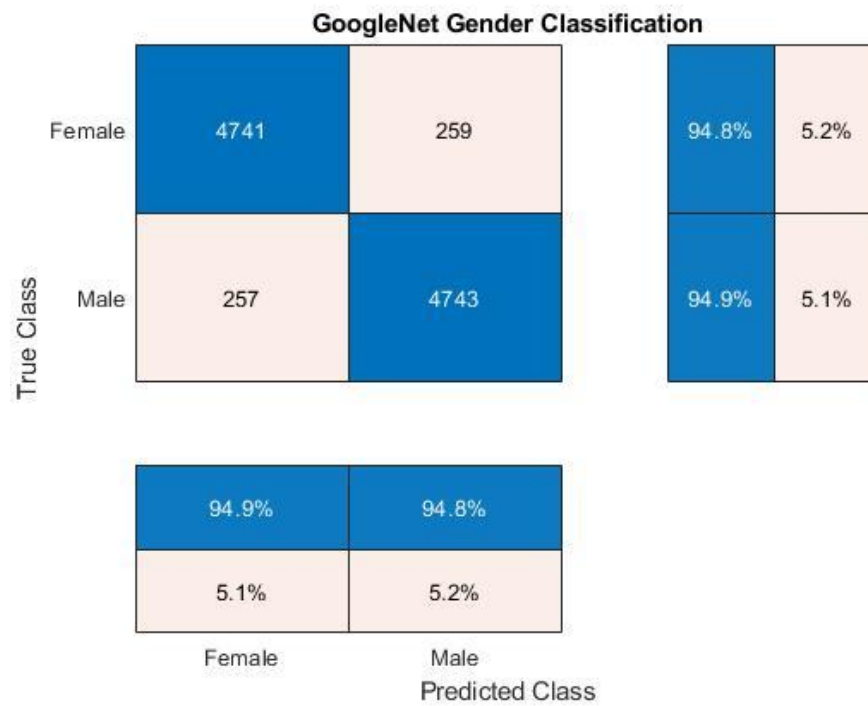


Figure 163 Confusion matrix GoogLeNet gender classification

This model has predicted following results.

$$TP = 4741$$

$$TN = 4743$$

$$FN = 259$$

$$FP = 257$$

(C) Precision, Recall, Accuracy and F1 Score

The values of accuracy, recall, precision, specificity and F1 Score are as follows.

$$Precision = 0.9486$$

$$Accuracy = 0.9484$$

$$Recall = 0.9482$$

$$F1\ Score = 0.9484$$

$$Specificity = 0.9486$$

III. Gender Classification Results using VGG-Face

With the same training data, which was used for GoogLeNet and AlexNet, the transfer learning through AlexNet produce the best results with much higher accuracy results, high precision and recall. In case of gender classification accuracy on validation data of VGG-Face is **98.5%** much higher than others. The main reason is the convolution base of pre-trained VGG-Face which is purely trained on face images data and detailed learning of facial features. Above the convolution base of VGG-Face the feedforward having convolution layers is used to classify as did in AlexNet and GoogLeNet.

(A) ROC and Precision vs Recall Curve

Figure 164, displays the ability of VGG-Face based Network to identify the genders, illustrates the effectiveness of the model and the trade-off between true positive and false positive rates with altering the threshold. The trade-off between precision and recall based on a threshold value is seen in Figure 165, in precision vs. recall curve. These curves demonstrate how accurate this classifier is in identifying the classes. It should be emphasised that both of these curves depend on the likelihood of predicted classes, and the threshold is determined from them. In comparison of previous estimators, these curves show better results.

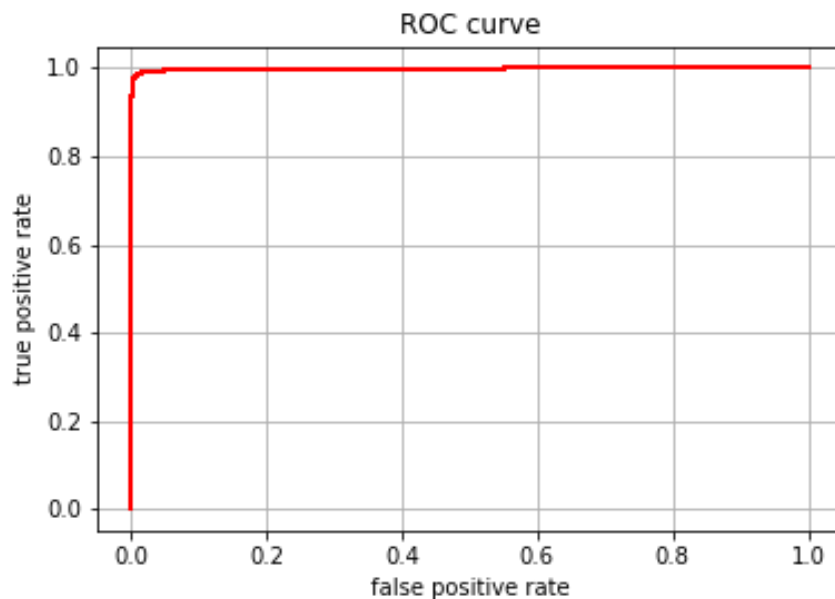


Figure 164 ROC - VGG-Face Gender Classification

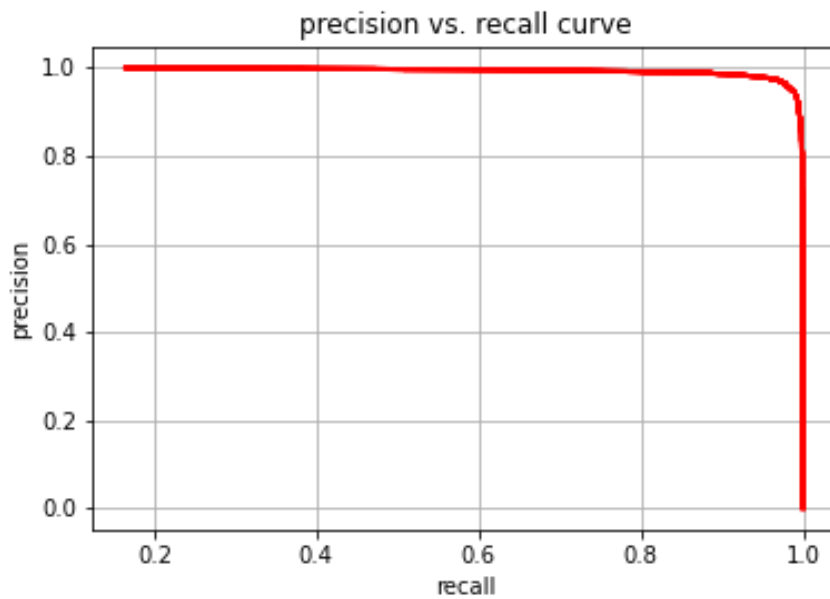


Figure 165 Precision Vs Recall - VGG-Face Gender Classification

(B) Confusion Matrix

The confusion matrix represents the recall and precision accuracy for each category. Figure 166 shows the confusion matrix generated for gender classification performed using VGG-Face. Same 5000 test images are used for estimating the performance of VGG-Face based model.

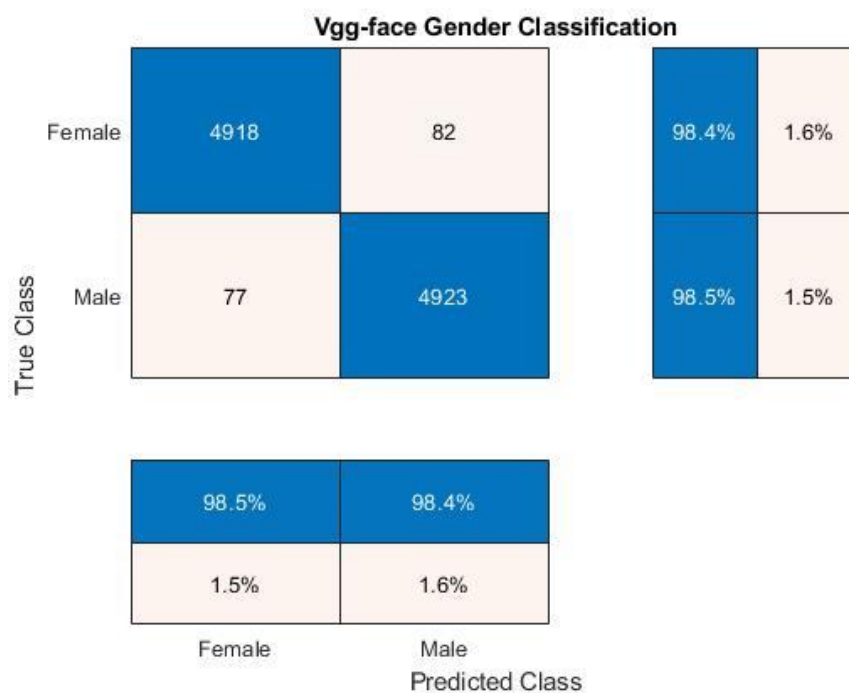


Figure 166 Confusion Matrix VGG-Face Gender Classification

This model has predicted following results.

$$TP = 4918$$

$$TN = 4923$$

$$FN = 82$$

$$FP = 77$$

(C) Precision, Recall, Accuracy and F1 Score

Values of accuracy, recall, precision, specificity and F1 Score are as follows.

$$Precision = 0.9846$$

$$Accuracy = 0.9841$$

$$Recall = 0.9836$$

$$F1\ Score = 0.9841$$

$$Specificity = 0.9846$$

6.04 Face Type Classification

Face Type classification was performed by training the dataset on AlexNet, GoogLeNet and VGG-Face. Results are show in this section.

I. Face Type Classification Results using AlexNet

Training and validation results for face type classification in AlexNet is **89.35%** which is less than GoogLeNet and VGG-Face. In general accuracy of testing data is slightly less than validation accuracy.

(A) ROC and Precision vs Recall Curve

In multiclass classification, ROC is a plot between true positive rate and false positive rate for each class. ROC curve for face type classification using AlexNet is given Figure 167 showing the performance of the model and trade-off between true positive and false positive rates with varying the threshold of each class. Precision vs recall curve for all the classes is represented in Figure 168 illustrate trade-off of the precision and recall on the basis of threshold value. These curves represent the diagnostic ability of the classifier. Blue line represents performance of model for

predicting face type I, orange for face type II, green for face type III and finally the performance of face type IV is represented by red line against multiple threshold levels.

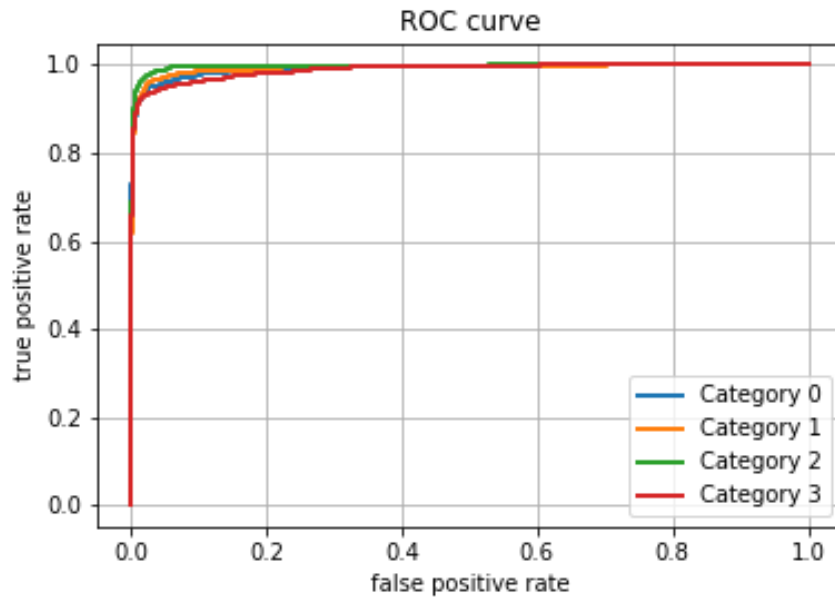


Figure 167 ROC Curve for Face Type Classification using AlexNet

ROC curve shows that the model classifies face type III more efficiently than others whereas it struggles for classifying face type IV. Similar results can be seen in precision vs recall curve.

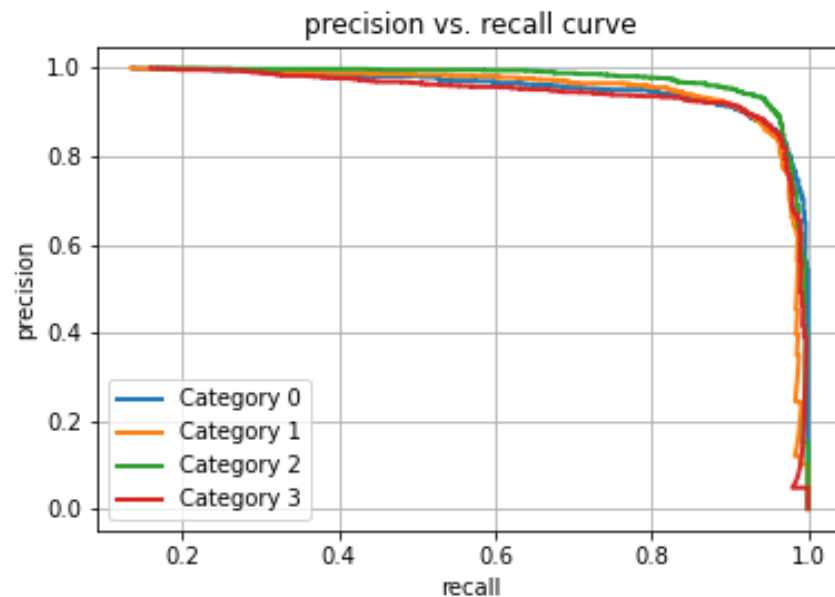


Figure 168 Precision vs Recall Curve Face Type Classification using AlexNet

(B) Confusion Matrix

Figure 169 shows the confusion matrix that also represents the precision and recall on face type classification. Confusion matrix is generated by testing the model on 2000 test images per class.

Alexnet Region Classification							
True Class	class 1	1799	100	31	70	90.0%	10.1%
	class 2	50	1810	80	60	90.5%	9.5%
	class 3	30	70	1819	81	91.0%	9.1%
	class 4	51	100	100	1749	87.5%	12.5%
		93.2%	87.0%	89.6%	89.2%		
		6.8%	13.0%	10.4%	10.8%		
		class 1	class 2	class 3	class 4		
		Predicted Class					

Figure 169 Face Type Classification Confusion Matrix

True positives are represented by the confusion matrix whereas true negatives, false positive and false negatives per class are calculated as shown below.

True Positives for Face Type using AlexNet:

$$TP_{TYPE I} = 1799$$

$$TP_{TYPE II} = 1810$$

$$TP_{TYPE III} = 1819$$

$$TP_{TYPE IV} = 1749$$

$$True\ Positive_{OVERALL} = 7177$$

True Negatives for Face Type using AlexNet:

$$TN_{TYPE\ I} = 5869$$

$$TN_{TYPE\ II} = 5730$$

$$TN_{TYPE\ III} = 5789$$

$$TN_{TYPE\ IV} = 5789$$

$$TN_{OVERALL} = 0$$

False Positives for Face Type using AlexNet:

$$FP_{TYPE\ I} = 131$$

$$FP_{TYPE\ II} = 270$$

$$FP_{TYPE\ III} = 211$$

$$FP_{TYPE\ IV} = 211$$

$$FP_{OVERALL} = 823$$

False Negatives for Face Type using AlexNet:

$$FN_{TYPE\ I} = 201$$

$$FN_{TYPE\ II} = 190$$

$$FN_{TYPE\ III} = 181$$

$$FN_{TYPE\ IV} = 251$$

$$FN_{OVERALL} = 823$$

(C) Precision, Recall, Accuracy and F1 Score

Accuracy for Face Type using AlexNet:

$$Accuracy_{TYPE\ I} = 0.9585$$

$$Accuracy_{TYPE\ II} = 0.9425$$

$$Accuracy_{TYPE\ III} = 0.951$$

$$Accuracy_{TYPE\ IV} = 0.951$$

$$Accuracy_{OVERALL} = 0.8971$$

Precision for Face Type using AlexNet:

$$Precision_{TYPE\ I} = 0.9321$$

$$Precision_{TYPE\ II} = 0.8702$$

$$Precision_{TYPE\ III} = 0.8960$$

$$Precision_{TYPE\ IV} = 0.8923$$

$$Precision_{MACRO\ AVERAGE} = 0.8977$$

Recall for Face Type using AlexNet:

$$Recall_{TYPE\ I} = 0.8995$$

$$Recall_{TYPE\ II} = 0.9050$$

$$Recall_{TYPE\ III} = 0.9095$$

$$Recall_{TYPE\ IV} = 0.8745$$

$$Recall_{MACRO\ AVERAGE} = 0.8971$$

F1-Score for Face Type using AlexNet:

$$F1\ Score_{TYPE\ I} = 0.9155$$

$$F1\ Score_{TYPE\ II} = 0.8873$$

$$F1\ Score_{TYPE\ III} = 0.9027$$

$$F1\ Score_{TYPE\ IV} = 0.8833$$

$$F1\ Score_{MACRO\ AVERAGE} = 0.8974$$

Specificity for Face Type using AlexNet:

$$Specificity_{TYPE\ I} = 0.9781$$

$$Specificity_{TYPE\ II} = 0.9550$$

$$Specificity_{TYPE\ III} = 0.9648$$

$$Specificity_{TYPE\ IV} = 0.9648$$

(D) Results summary for Facial Hair Type classification using AlexNet

Table 15 shows the results summary for Face Type classification using AlexNet.

Table 15 Results summary for Face Type classification using AlexNet

	True Positive	True Negative	False Positive	False Negative	Precision	Recall	Specificity	Accuracy	F1-Score
FACE TYPE 1	1799	5869	131	201	93.21%	89.95%	0.9782	95.85%	0.9155
FACE TYPE 2	1810	5730	270	190	87.02%	90.50%	0.9550	94.25%	0.8873
FACE TYPE 3	1819	5789	211	181	89.61%	90.95%	0.9648	95.10%	0.9027
FACE TYPE 4	1749	5789	211	251	89.23%	87.45%	0.9648	94.23%	0.8833

Figure 170 shows the tabular view of comparison of precisions of each type and the overall precision. Figure 171 represents the recalls of each type. Whereas Figure 172 gives the overall and each class accuracy. Final F1 scores comparison id given in Figure 173.

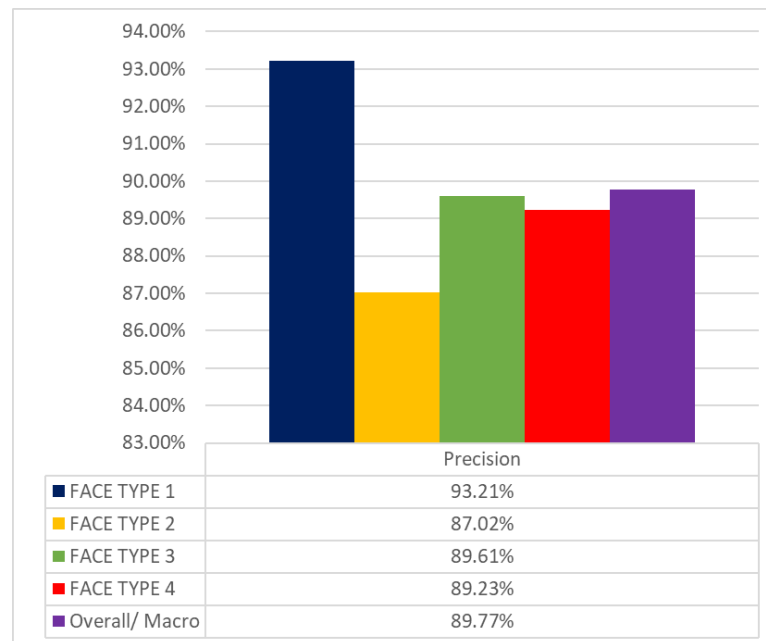


Figure 170 Precisions Comparisons for Each Face Type

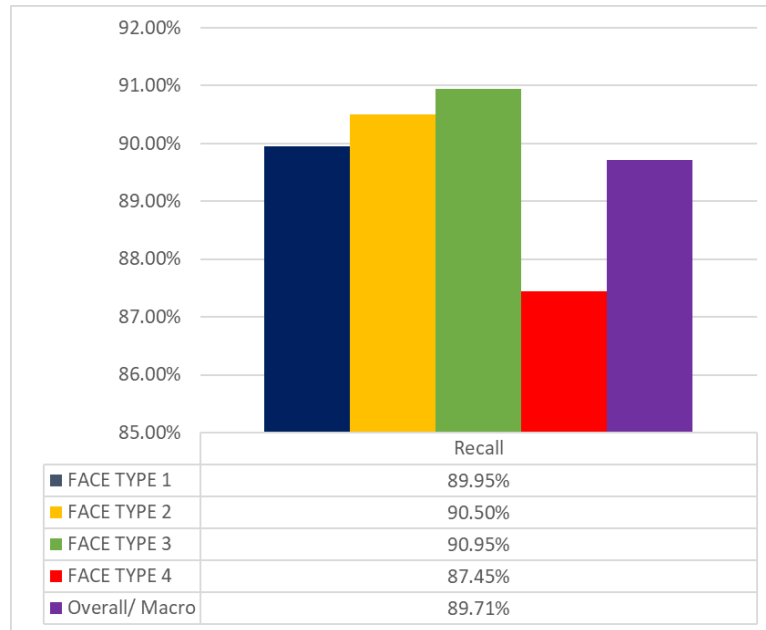


Figure 171 Recall Comparisons for Each Face Type

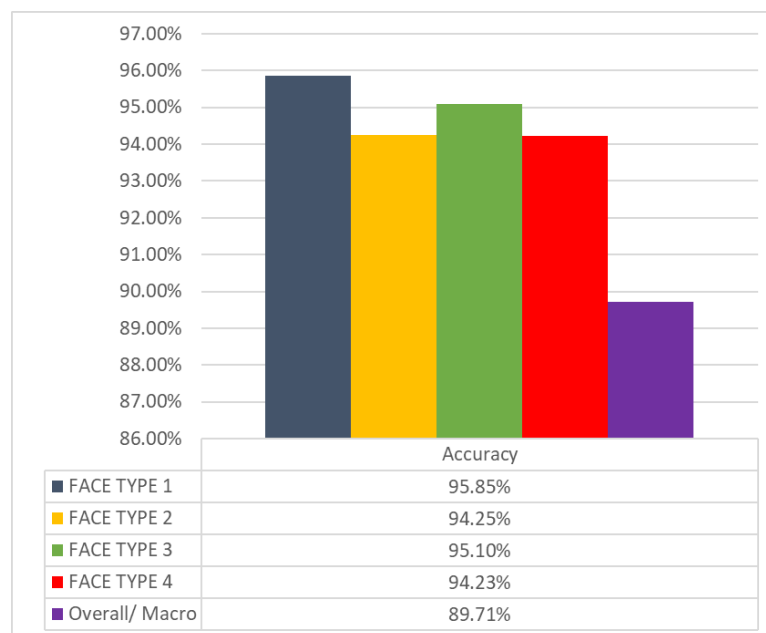


Figure 172 Accuracy Comparisons for Each Face Type

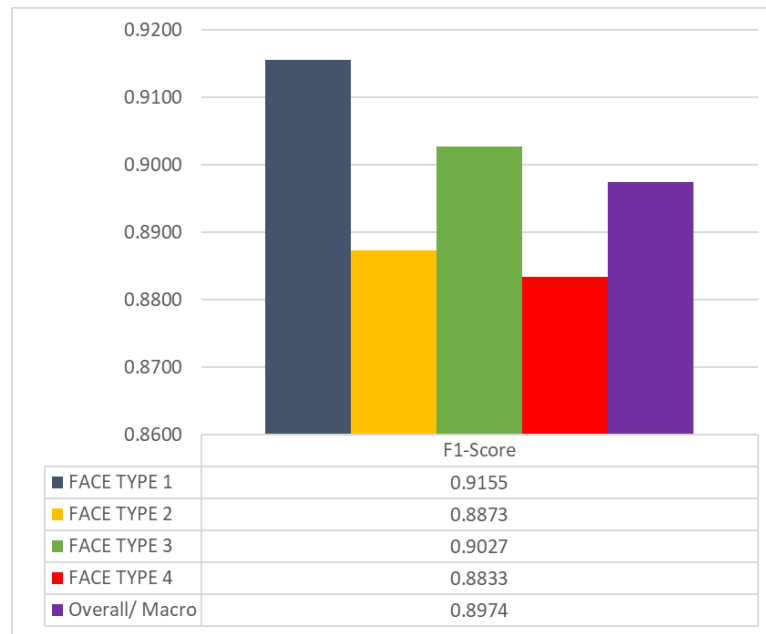


Figure 173 F1 Score Comparisons for Each Face Type

II. Face Type Classification Results using GoogLeNet

Unlike gender classification, the performance of GoogLeNet was better than AlexNet on face type classification. Accuracy of GoogLeNet is better than AlexNet but not than VGG-Face which will be discussed later. On the same data and preprocessing steps the accuracy is **95.76%** for region classification having for category

(A) ROC and Precision vs Recall Curve

. ROC curve for face type classification using GoogLeNet is given Figure 174 showing the performance of the model and trade-off between true positive and false positive rates with varying the threshold of each class. Precision vs recall curve for all the classes is represented in Figure 175, it illustrates the trade-off of the precision and recall on the basis of threshold value. These curves represent the diagnostic ability of the classifier. Blue line represents performance of model for predicting face type I, orange for face type II, green for face type III and finally the performance of face type IV is represented by red line against multiple threshold levels.

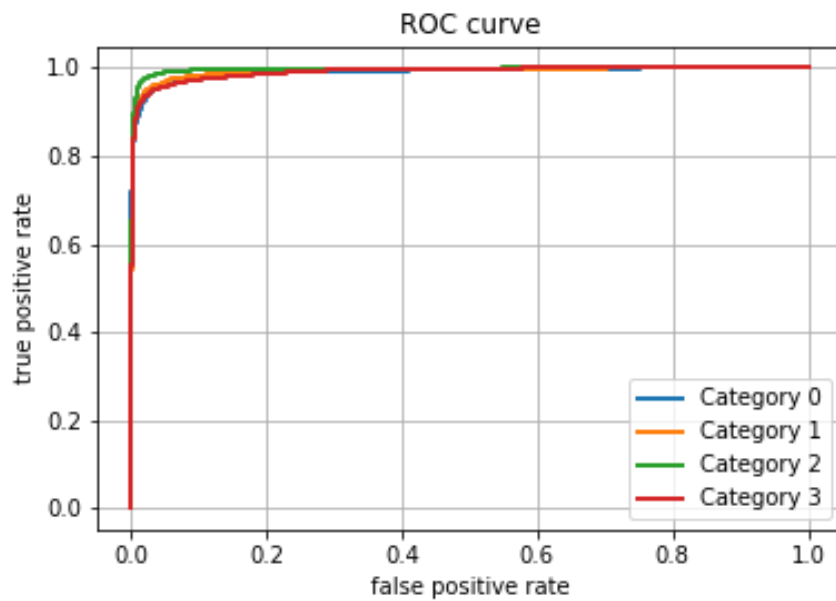


Figure 174 ROC Curve Face Type Classification GoogLeNet

Curves of GoogLeNet shows better results than AlexNet in classification of face types. ROC curve shows that the model classifies face type III more efficiently than others whereas it struggles for classifying face type I. Similar results can be seen in precision vs recall curve.

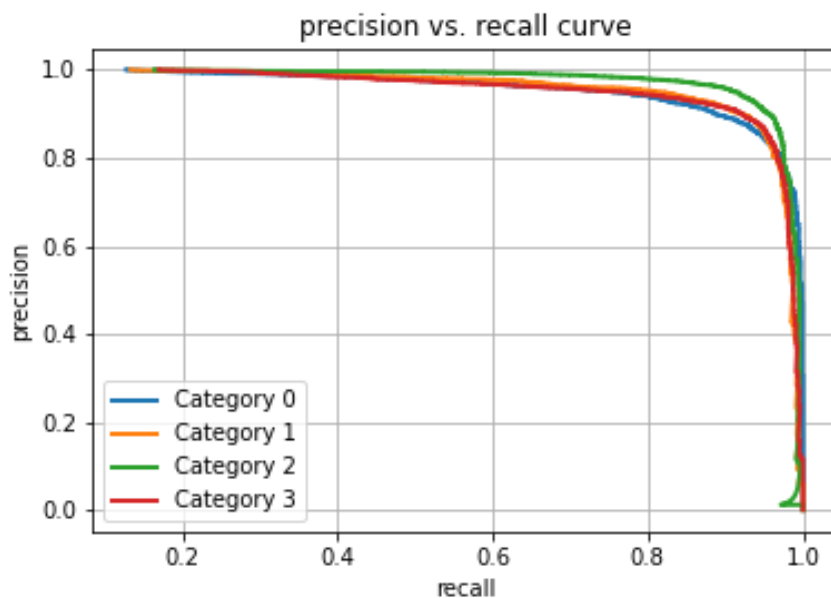


Figure 175 Precision vs Recall – Face Type Classification GoogLeNet

(B) Confusion Matrix

Figure 176 displays the confusion matrix with recall and precision for each face type category. 2000 test images per class were passed through the model for evaluation and generating the confusion matrix.

Googlenet Region Classification							
True Class	class 1	1774	125	51	50	88.7%	11.3%
	class 2	60	1815	55	70	90.8%	9.3%
	class 3	40	70	1827	63	91.3%	8.7%
	class 4	45	120	96	1739	87.0%	13.0%
		92.4%	85.2%	90.0%	90.5%		
		7.6%	14.8%	10.0%	9.5%		
		class 1	class 2	class 3	class 4		
		Predicted Class					

Figure 176 Confusion Matrix for GoogLeNet Face Type Classification

True positives are represented by the confusion matrix whereas true negatives, false positive and false negatives per class are calculated as shown below.

True Positives for Face Type using GoogLeNet:

$$TP_{TYPE I} = 1774$$

$$TP_{TYPE II} = 1815$$

$$TP_{TYPE III} = 1827$$

$$TP_{TYPE IV} = 1739$$

$$TP_{OVERALL} = 7155$$

True Negatives for Face Type using GoogLeNet:

$$TN_{TYPE\ I} = 5855$$

$$TN_{TYPE\ II} = 5685$$

$$TN_{TYPE\ III} = 5798$$

$$TN_{TYPE\ IV} = 5817$$

$$TN_{OVERALL} = 0$$

False Positives for Face Type using GoogLeNet:

$$FP_{TYPE\ I} = 145$$

$$FP_{TYPE\ II} = 315$$

$$FP_{TYPE\ III} = 202$$

$$FP_{TYPE\ IV} = 183$$

$$FP_{OVERALL} = 845$$

False Negatives for Face Type using GoogLeNet:

$$FN_{TYPE\ I} = 226$$

$$FN_{TYPE\ II} = 185$$

$$FN_{TYPE\ III} = 173$$

$$FN_{TYPE\ IV} = 261$$

$$FN_{OVERALL} = 845$$

(C) Precision, Recall, Accuracy and F1 Score

Accuracy for Face Type using GoogLeNet:

$$Accuracy_{TYPE\ I} = 0.9536$$

$$Accuracy_{TYPE\ II} = 0.9376$$

$$Accuracy_{TYPE\ III} = 0.9531$$

$$Accuracy_{TYPE\ IV} = 0.9445$$

$$Accuracy_{OVERALL} = 0.8943$$

Precision for Face Type using GoogLeNet:

$$Precision_{TYPE\ I} = 0.9244$$

$$Precision_{TYPE\ II} = 0.8521$$

$$Precision_{TYPE\ III} = 0.9004$$

$$Precision_{TYPE\ IV} = 0.9048$$

$$Precision_{MACRO\ AVERAGE} = 0.8954$$

Recall for Face Type using GoogLeNet:

$$Recall_{TYPE\ I} = 0.8870$$

$$Recall_{TYPE\ II} = 0.9075$$

$$Recall_{TYPE\ III} = 0.9135$$

$$Recall_{TYPE\ IV} = 0.8695$$

$$Recall_{MACRO\ AVERAGE} = 0.8944$$

F1-Score for Face Type using GoogLeNet:

$$F1\ Score_{TYPE\ I} = 0.9053$$

$$F1\ Score_{TYPE\ II} = 0.8789$$

$$F1\ Score_{TYPE\ III} = 0.9069$$

$$F1\ Score_{TYPE\ IV} = 0.8867$$

$$F1\ Score_{MACRO\ AVERAGE} = 0.8949$$

Specificity for Face Type using GoogLeNet:

$$Specificity_{TYPE\ I} = 0.9758$$

$$Specificity_{TYPE\ II} = 0.9475$$

$$Specificity_{TYPE\ III} = 0.9663$$

$$Specificity_{TYPE\ IV} = 0.9695$$

(D) Results summary for Face Type classification using GoogLeNet

Table 16 shows the results summary for Face Type classification using GoogLeNet.

Table 16 Results summary for Face Type classification using GoogLeNet

	True Positive	True Negative	False Positive	False Negative	Precision	Recall	Specificity	Accuracy	F1-Score
FACE TYPE 1	1774	5855	145	226	92.44%	88.70%	0.9758	95.36%	0.9053
FACE TYPE 2	1815	5685	315	185	85.21%	90.75%	0.9475	93.75%	0.8789
FACE TYPE 3	1827	5798	202	173	90.04%	91.35%	0.9663	95.31%	0.9069
FACE TYPE 4	1739	5817	183	261	90.48%	86.95%	0.9695	94.45%	0.8868

Figure 177 shows comparison of precisions of each face type and the overall precision. Figure 178 shows the recalls of against each type. Figure 179 displays the overall and each class's accuracy. Final F1 scores comparison is given in Figure 180.

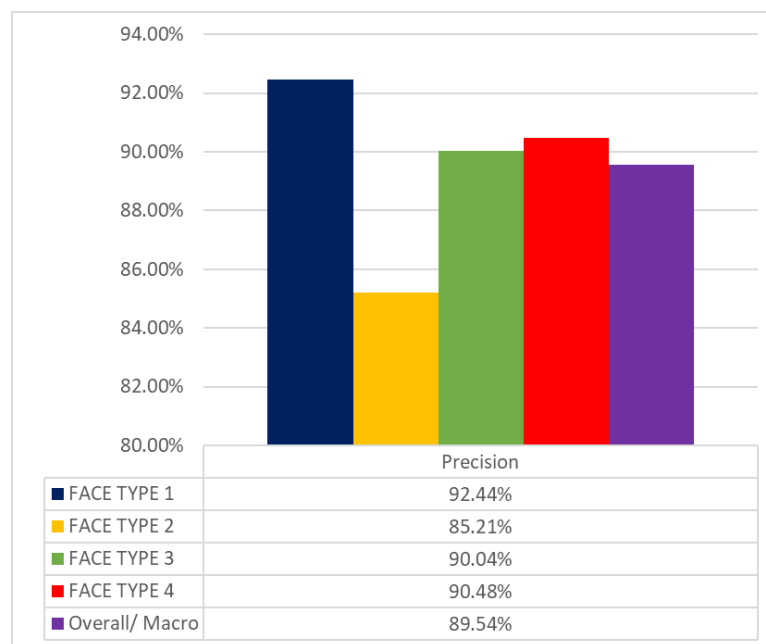


Figure 177 Precisions Comparisons for Each Face Type

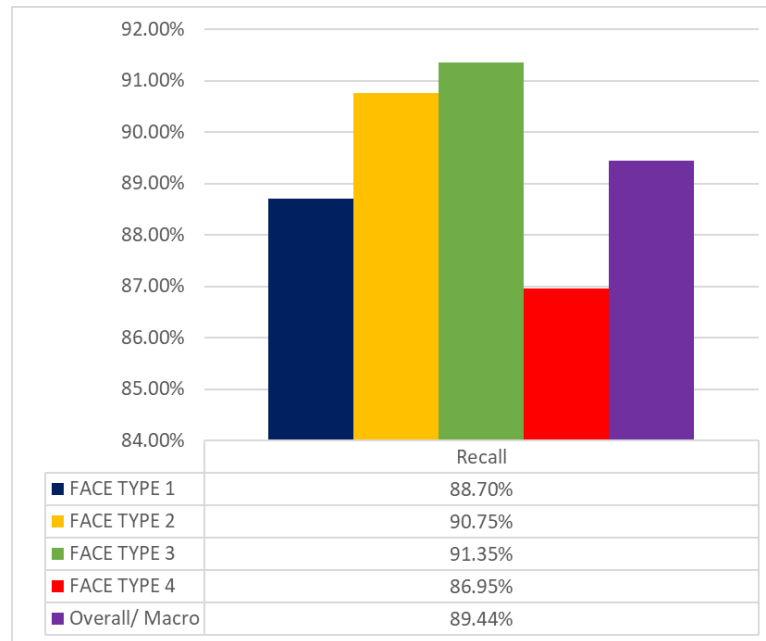


Figure 178 Recall Comparisons for Each Face Type

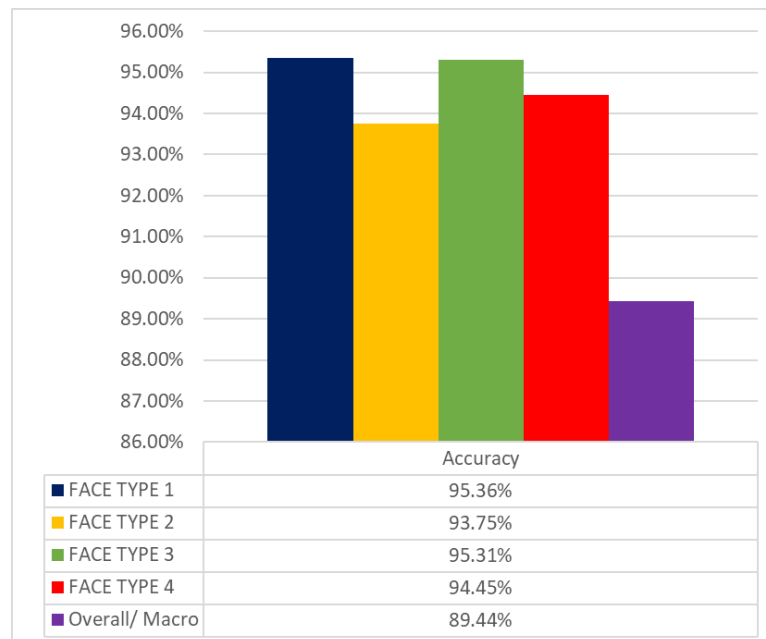


Figure 179 Accuracy Comparisons for Each Face Type

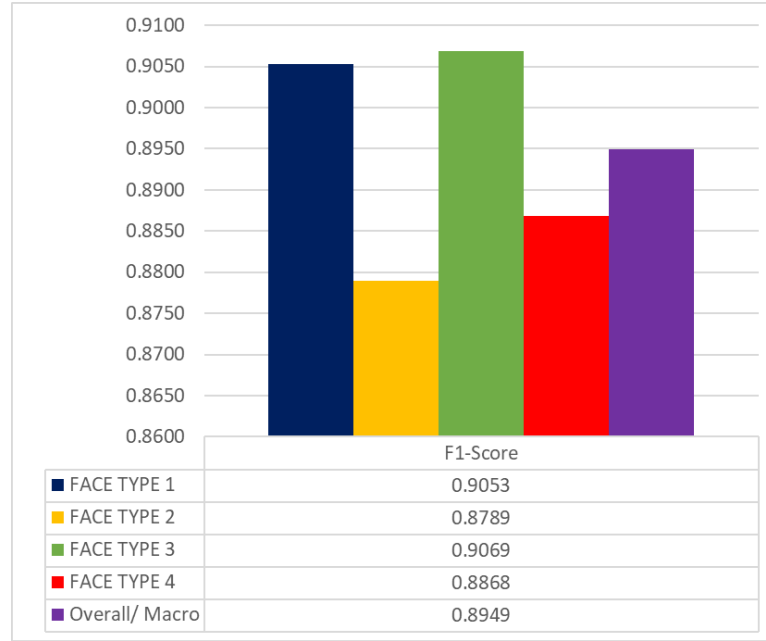


Figure 180 F1 Score Comparisons for Each Face Type

III. Face Type Classification Results using VGG-Face

Just like the use of transfer learning on VGG face for gender classification proved to be the best, the similar results found in classification of face type classification model. The accuracy of VGG face reached to **98.1%**. Using the convolution base of VGG-Face and feedforward network of 6 layers having activation function of ReLU in hidden layer and SoftMax activation function at output layer did the job. The output of convolution base is feature matrix of 4096 which passed through Convolution network of fully connected layers for classification task. So have 4 neurons at output layer.

(A) ROC and Precision vs Recall Curve

. ROC curve for face type classification using GoogLeNet is given Figure 181 showing the performance of the model and trade-off between true positive and false positive rates with varying the threshold of each class. Precision vs recall curve for all the classes is represented in Figure 182, it illustrates the trade-off of the precision and recall on the basis of threshold value. These curves represent the diagnostic ability of the classifier. Blue line represents performance of model for predicting face type I, orange for face type II, green for face type III and finally the performance of face type IV is represented by red line against multiple threshold levels.

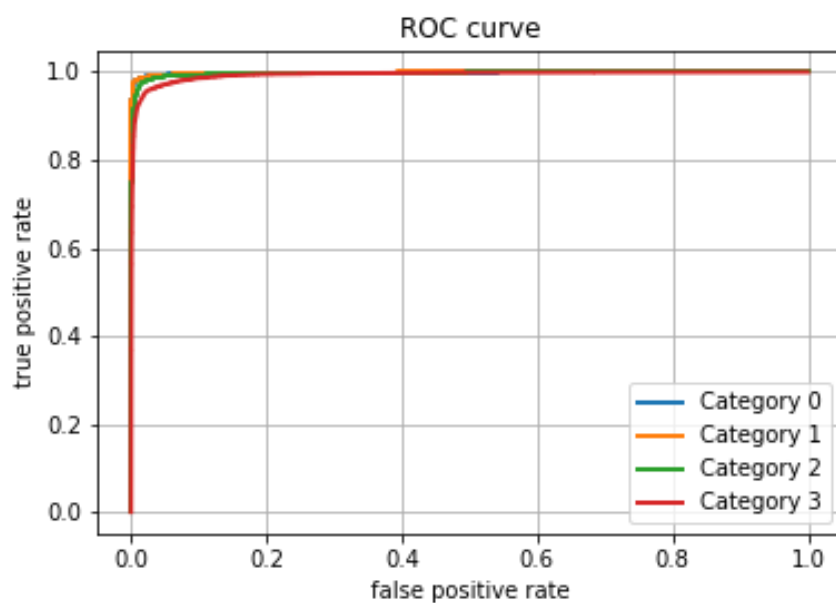


Figure 181 Face Type Classification ROC curve using VGG-Face

ROC curve shows that the model classifies face type II more efficiently than others whereas it struggles for classifying face type IV. Similar results can be seen in precision vs recall curve.

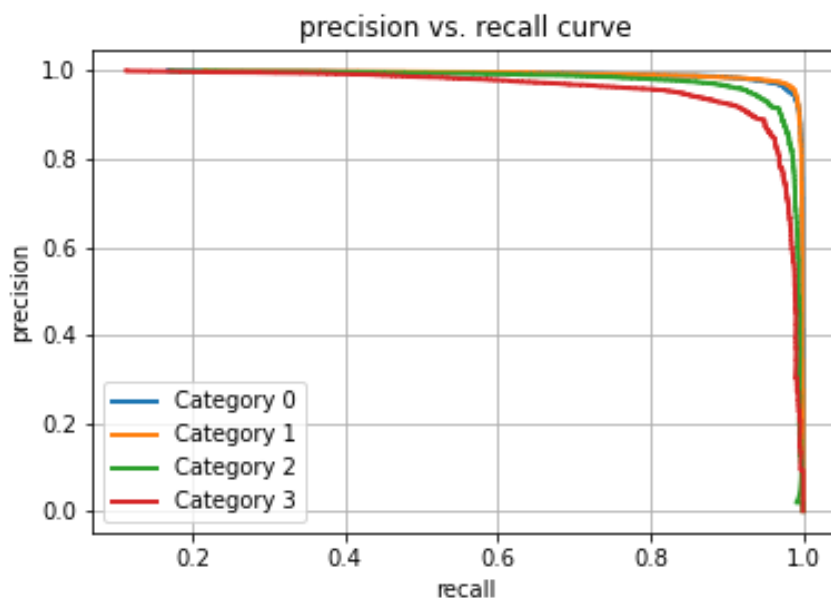


Figure 182 Face Type Classification Precision vs Recall Curve using VGG-Face

(B) Confusion Matrix

Figure 183 displays the confusion matrix with recall and precision for each face type category. 2000 test images per class were passed through the model for evaluation and generating the confusion matrix.

Vgg-face Region Classification							
True Class	class 1	1974	7	15	4	98.7%	1.3%
	class 2	6	1936	44	14	96.8%	3.2%
	class 3	40	9	1927	24	96.4%	3.6%
	class 4	35	20	6	1939	97.0%	3.0%
		96.1%	98.2%	96.7%	97.9%		
		3.9%	1.8%	3.3%	2.1%		
		class 1	class 2	class 3	class 4		
		Predicted Class					

Figure 183 Confusion Matrix for VGG Face for Face Type Classification

True positives are represented by the confusion matrix whereas true negatives, false positive and false negatives per class are calculated as shown below.

True Positives for Face Type using VGG-Face:

$$TP_{TYPE I} = 1974$$

$$TP_{TYPE II} = 1936$$

$$TP_{TYPE III} = 1927$$

$$TP_{TYPE IV} = 1939$$

$$TP_{OVERALL} = 7776$$

True Negatives for Face Type using VGG-Face:

$$TN_{TYPE\ I} = 5919$$

$$TN_{TYPE\ II} = 5964$$

$$TN_{TYPE\ III} = 5935$$

$$TN_{TYPE\ IV} = 5958$$

$$TN_{OVERALL} = 0$$

False Positives for Face Type using VGG-Face:

$$FP_{TYPE\ I} = 81$$

$$FP_{TYPE\ II} = 36$$

$$FP_{TYPE\ III} = 65$$

$$FP_{TYPE\ IV} = 42$$

$$FP_{OVERALL} = 224$$

False Negatives for Face Type using VGG-Face:

$$FN_{TYPE\ I} = 26$$

$$FN_{TYPE\ II} = 64$$

$$FN_{TYPE\ III} = 73$$

$$FN_{TYPE\ IV} = 61$$

$$FN_{OVERALL} = 224$$

(C) Precision, Recall, Accuracy and F1 Score

Accuracy for Face Type using VGG-Face:

$$Accuracy_{TYPE\ I} = 0.9866$$

$$Accuracy_{TYPE\ II} = 0.9875$$

$$Accuracy_{TYPE\ III} = 0.9828$$

$$Accuracy_{TYPE\ IV} = 0.9871$$

$$Accuracy_{OVERALL} = 0.9720$$

Precision for Face Type using VGG-Face:

$$Precision_{TYPE\ I} = 0.9605$$

$$Precision_{TYPE\ II} = 0.9817$$

$$Precision_{TYPE\ III} = 0.9674$$

$$Precision_{TYPE\ IV} = 0.9788$$

$$Precision_{MACRO\ AVERAGE} = 0.9721$$

Recall for Face Type using VGG-Face:

$$Recall_{TYPE\ I} = 0.9870$$

$$Recall_{TYPE\ II} = 0.9680$$

$$Recall_{TYPE\ III} = 0.9635$$

$$Recall_{TYPE\ IV} = 0.9695$$

$$Recall_{MACRO\ AVERAGE} = 0.9720$$

F1-Score for Face Type using VGG-Face:

$$F1\ Score_{TYPE\ I} = 0.9736$$

$$F1\ Score_{TYPE\ II} = 0.9748$$

$$F1\ Score_{TYPE\ III} = 0.9654$$

$$F1\ Score_{TYPE\ IV} = 0.9741$$

$$F1\ Score_{MACRO\ AVERAGE} = 0.9721$$

Specificity for Face Type using VGG-Face:

$$Specificity_{TYPE\ I} = 0.9865$$

$$Specificity_{TYPE\ II} = 0.994$$

$$Specificity_{TYPE\ III} = 0.9892$$

$$Specificity_{TYPE\ IV} = 0.9788$$

(D) Results summary for Face Type classification using VGG Face

Table 17 shows the results summary for Face Type classification using VGG Face.

Table 17 Results summary for Face Type classification using VGG Face

	True Positive	True Negative	False Positive	False Negative	Precision	Recall	Specificity	Accuracy	F1-Score
FACE TYPE 1	1974	5919	81	26	96.06%	98.70%	0.9865	98.66%	0.9736
FACE TYPE 2	1936	5964	36	64	98.17%	96.80%	0.9940	98.75%	0.9748
FACE TYPE 3	1927	5935	65	73	96.74%	96.35%	0.9892	98.28%	0.9654
FACE TYPE 4	1939	5958	42	61	97.88%	96.95%	0.9930	98.71%	0.9741

Figure 184 gives comparison of precisions of each face type and overall precision. Figure 185 displays recall of against each type. Figure 186 shows overall and each class's accuracy. Final F1 scores comparison is showed in Figure 187

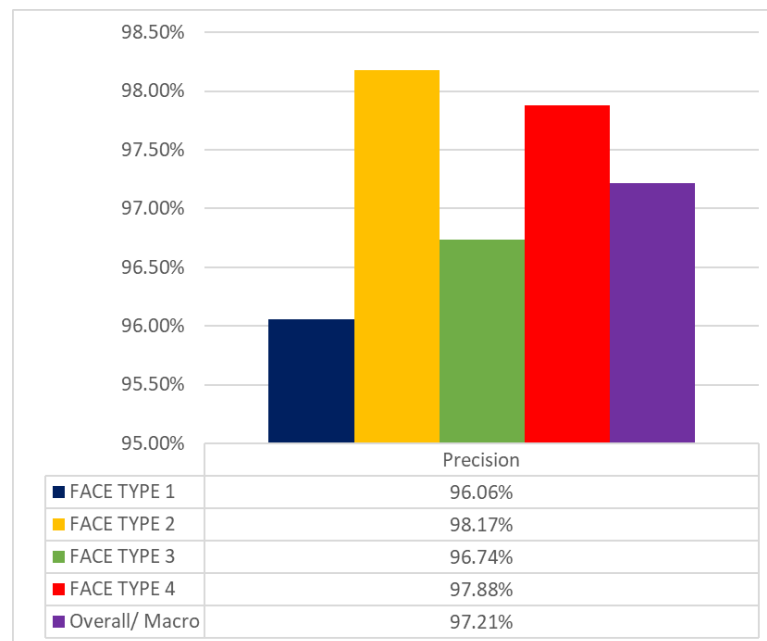


Figure 184 Precisions Comparisons for Each Face Type

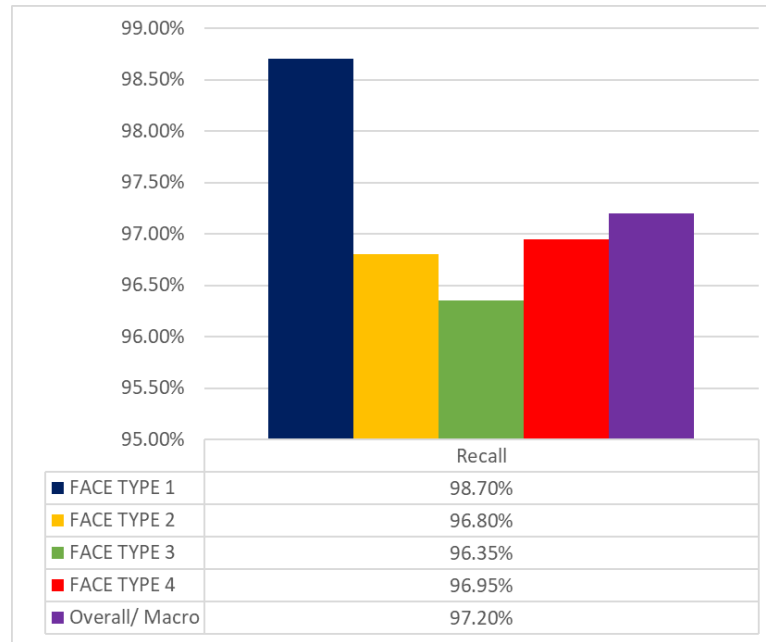


Figure 185 Recall Comparisons for Each Face Type

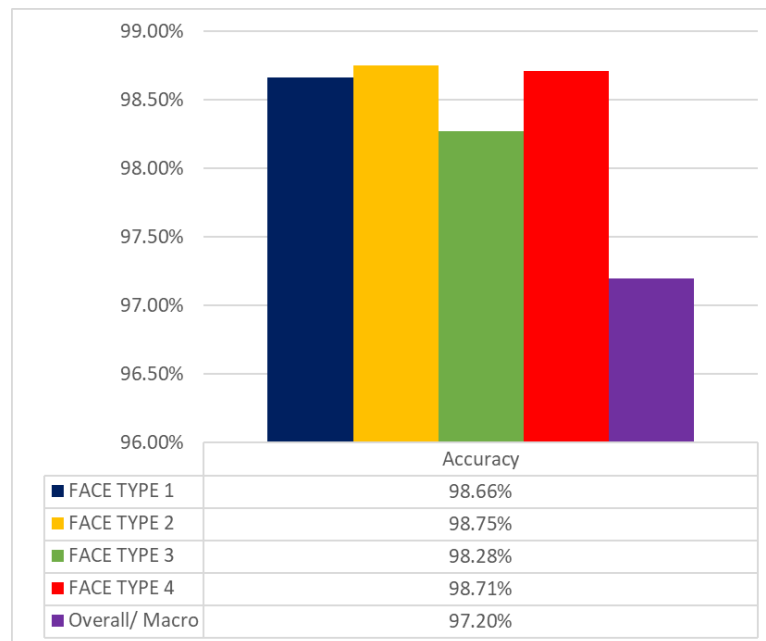


Figure 186 Accuracy Comparisons for Each Face Type

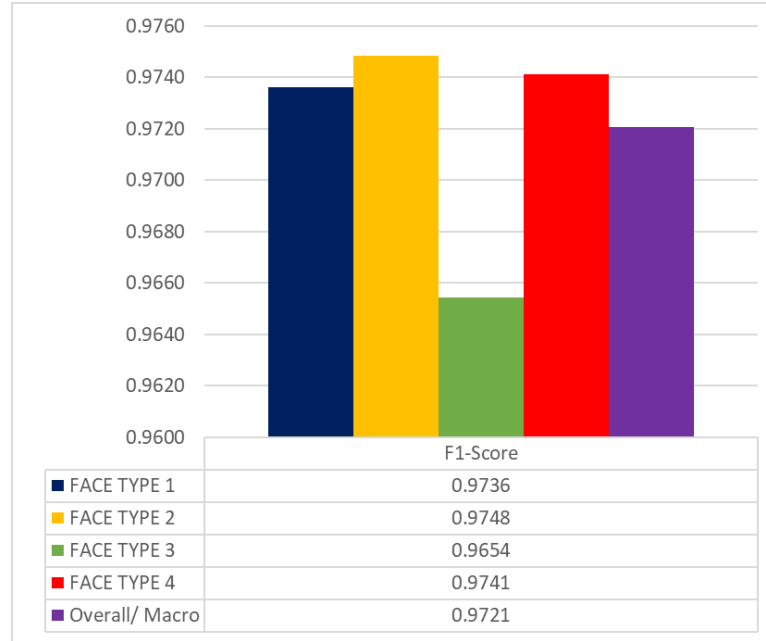


Figure 187 F1 Score Comparisons for Each Face Type

6.05 Facial Hair Classification

Performing the beard classification as a pre-processing step for identity recognition provides reduction in time. Four classes for this task were already defined as follows. Faces with no facial hairs grown are categorized as Type I, Type II is dedicated to the faces having moustache but no other facial hair. Faces with long beard are categorized as Type III and finally the face images with short beard are classified as Type IV. To perform this task transfer learning technique was adapted by using AlexNet, GoogLeNet, Customized Network and ResNet. Sample data is given in Figure 188.

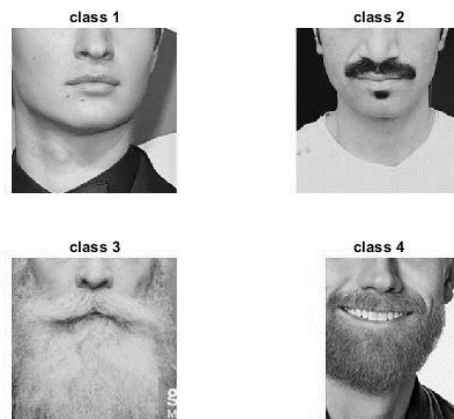


Figure 188 Facial Hair Classes Samples

The data is trained on the images collected from google. Images were downloaded by using “Furtkun extension” available in chrome. Data augmentation operations was applied including (Reflection, rotation with diff. degrees). Furthermore, resizing, normalization and converting to grayscale image was also included in data preparation.

I. Facial Hair Classification Results using GoogLeNet

On the other hand, GoogLeNet which has more complex and dense structure than sequential structured AlexNet performs much better with validation accuracy of more than 87%.

(A) ROC and Precision vs Recall Curve

Figure 189 shows multiclass classification ROC plot between true positive rate and false positive rate for each facial hair type class for GoogLeNet based model. ROC curve for facial hair type classification shows the performance of the model and trade-off between true positive and false positive rates with varying the threshold of each class. Precision vs recall curve for all the classes is represented in Figure 190 illustrates trade-off of the precision and recall on the basis of threshold value. These curves represent the diagnostic ability of the classifier to classify facial hair type categories. Blue line represents performance of model for predicting facial hair type I, orange for facial hair type II, green for facial hair type III and finally the performance of facial hair type IV is represented by red line against multiple threshold levels.

ROC curve shows that the model classifies facial hair type III more efficiently than others whereas it struggles for classifying facial hair type I. Similar results can be seen in precision vs recall curve.

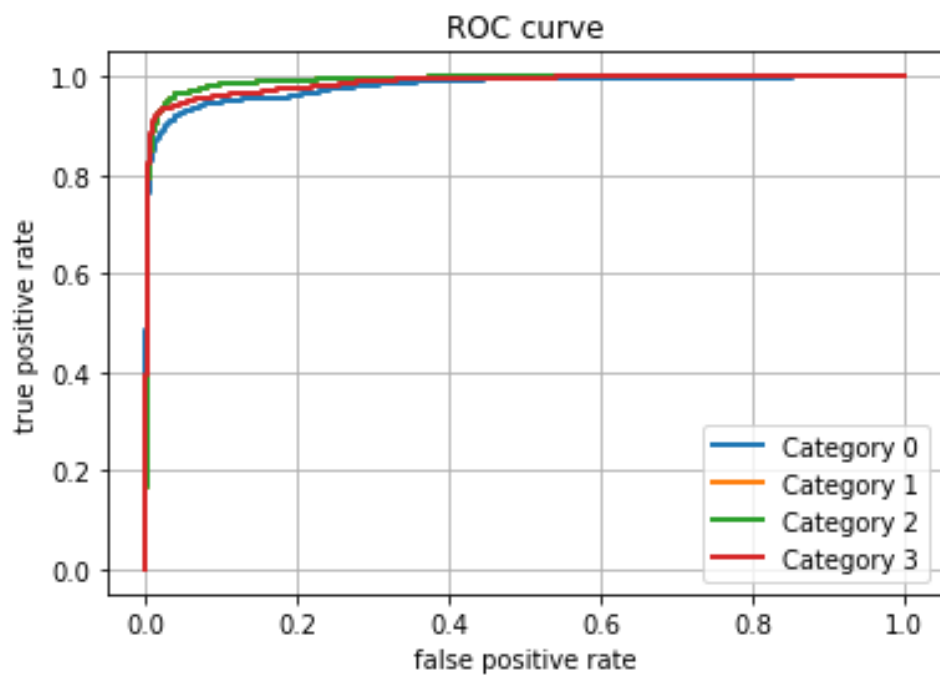


Figure 189 ROC Curve for Beard Classification using GoogLeNet

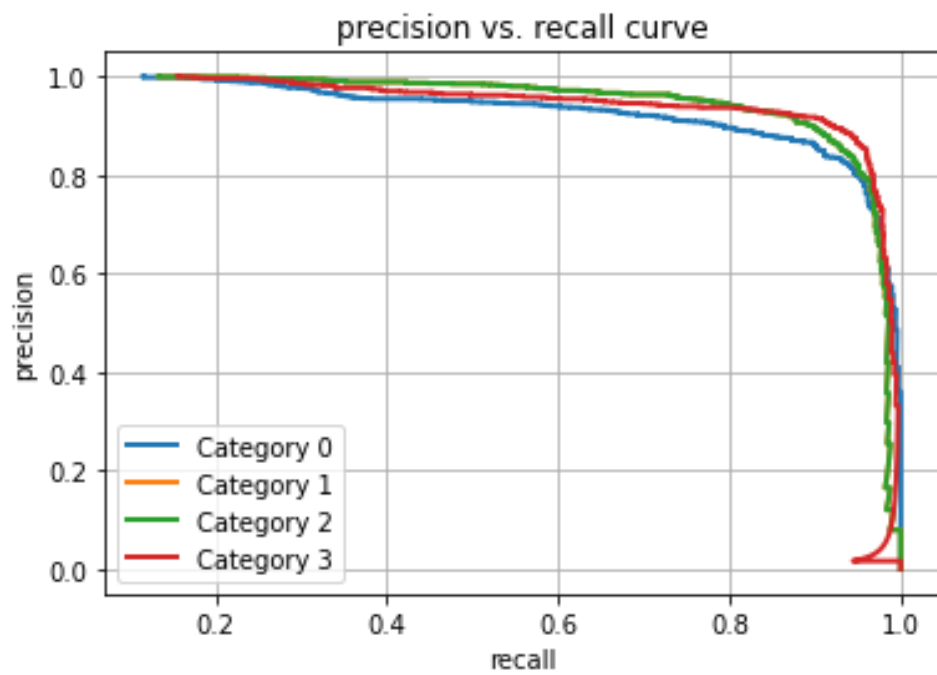


Figure 190 Precision vs Recall Curve for Beard Classification using GoogLeNet

(B) Confusion Matrix

Figure 191 displays the confusion matrix with recall and precision for each category.

Bearded classification googlenet							
True Class	class 1	1710	90	110	90	85.5%	14.5%
	class 2	40	1770	150	40	88.5%	11.5%
	class 3	70	12	1730	80	86.5%	13.5%
	class 4	70	10	50	1780	89.0%	11.0%
		class 1	class 2	class 3	class 4		
		90.5%	85.1%	84.8%	89.0%		
		9.5%	14.9%	15.2%	11.0%		
		Predicted Class					

Figure 191 Confusion Matrix for Beard Classification using GoogLeNet

(C) Precision, Recall, Accuracy and F1 Score

True Positives for Facial Hair Type using GoogLeNet:

$$TP_{TYPE I} = 1710$$

$$TP_{TYPE II} = 1770$$

$$TP_{TYPE III} = 1730$$

$$TP_{TYPE IV} = 1780$$

$$TP_{OVERALL} = 6990$$

True Negatives for Facial Hair Type using GoogLeNet:

$$TN_{TYPE I} = 5820$$

$$TN_{TYPE II} = 5690$$

$$TN_{TYPE III} = 5690$$

$$TN_{TYPE IV} = 5790$$

$$TN_{OVERALL} = 0$$

False Positives for Facial Hair Type using GoogLeNet:

$$FP_{TYPE\ I} = 180$$

$$FP_{TYPE\ II} = 310$$

$$FP_{TYPE\ III} = 310$$

$$FP_{TYPE\ IV} = 210$$

$$FP_{OVERALL} = 1010$$

False Negatives for Facial Hair Type using GoogLeNet:

$$FN_{TYPE\ I} = 290$$

$$FN_{TYPE\ II} = 230$$

$$FN_{TYPE\ III} = 270$$

$$FN_{TYPE\ IV} = 220$$

$$FN_{OVERALL} = 1010$$

Accuracy for Facial Hair Type using GoogLeNet:

$$Accuracy_{TYPE\ I} = 0.9412$$

$$Accuracy_{TYPE\ II} = 0.9325$$

$$Accuracy_{TYPE\ III} = 0.9275$$

$$Accuracy_{TYPE\ IV} = 0.9463$$

$$Accuracy_{OVERALL} = 0.8737$$

Precision for Facial Hair Type using GoogLeNet:

$$Precision_{TYPE\ I} = 0.9048$$

$$Precision_{TYPE\ II} = 0.8510$$

$$Precision_{TYPE\ III} = 0.8480$$

$$Precision_{TYPE\ IV} = 0.8945$$

$$Precision_{MACRO\ AVERAGE} = 0.8746$$

Recall for Facial Hair Type using GoogLeNet:

$$Recall_{TYPE\ I} = 0.8550$$

$$Recall_{TYPE\ II} = 0.8850$$

$$Recall_{TYPE\ III} = 0.8650$$

$$Recall_{TYPE\ IV} = 0.8900$$

$$Recall_{MACRO\ AVERAGE} = 0.8738$$

F1-Score for Facial Hair Type using GoogLeNet:

$$F1\ Score_{TYPE\ I} = 0.8792$$

$$F1\ Score_{TYPE\ II} = 0.8676$$

$$F1\ Score_{TYPE\ III} = 0.8564$$

$$F1\ Score_{TYPE\ IV} = 0.8922$$

$$F1\ Score_{MACRO\ AVERAGE} = 0.8742$$

Specificity for Facial Hair Type using GoogLeNet:

$$Specificity_{TYPE\ I} = 0.9700$$

$$Specificity_{TYPE\ II} = 0.9483$$

$$Specificity_{TYPE\ III} = 0.9483$$

$$Specificity_{TYPE\ IV} = 0.9650$$

(D) Results summary for Facial Hair Type Classification using GoogLeNet.

Table 18 shows results summary for Facial Hair Type Classification using GoogLeNet.

Table 18 Results Summary for Facial Hair Type Classification Using GoogLeNet

	True Positive	True Negative	False Positive	False Negative	Precision	Recall	Specificity	Accuracy	F1-Score
FACIAL HAIR TYPE 1	1710	5820	180	290	90.48%	85.50%	0.9700	94.13%	0.8792
FACIAL HAIR TYPE 2	1770	5690	310	230	85.10%	88.50%	0.9483	93.25%	0.8676
FACIAL HAIR TYPE 3	1730	5690	310	270	84.80%	86.50%	0.9483	92.75%	0.8564
FACIAL HAIR TYPE 4	1780	5790	210	220	89.45%	89.00%	0.9650	94.63%	0.8922

Figure 192 shows the tabular view of comparison of precisions of each type and the overall precision. Figure 193 represents the recalls of each type. Whereas Figure

194 gives the overall and each class accuracy. Final F1 scores comparison is given in Figure 195.

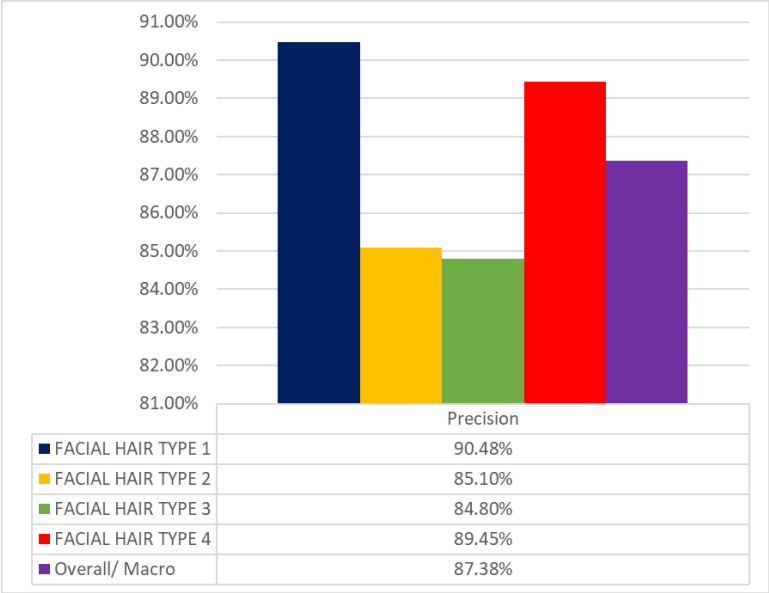


Figure 192 Precisions Comparisons for Each Facial Hair Type

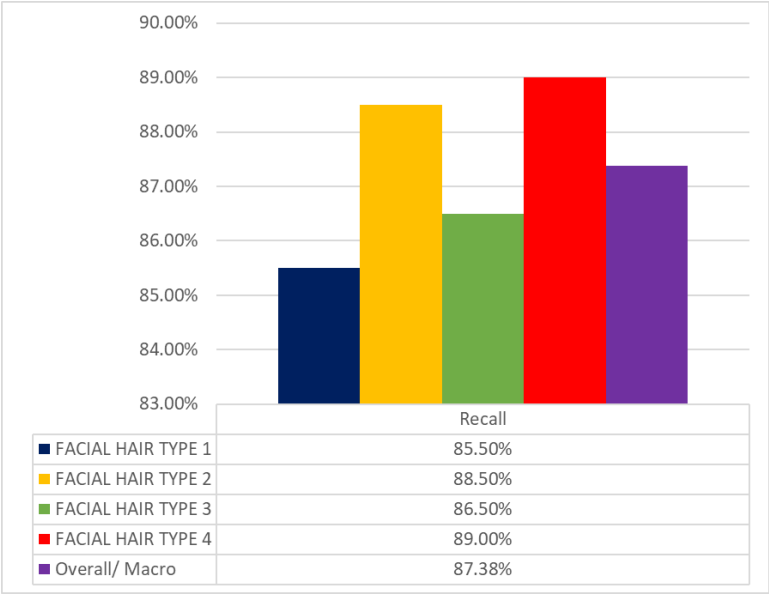


Figure 193 Recalls Comparisons for Each Facial Hair Type

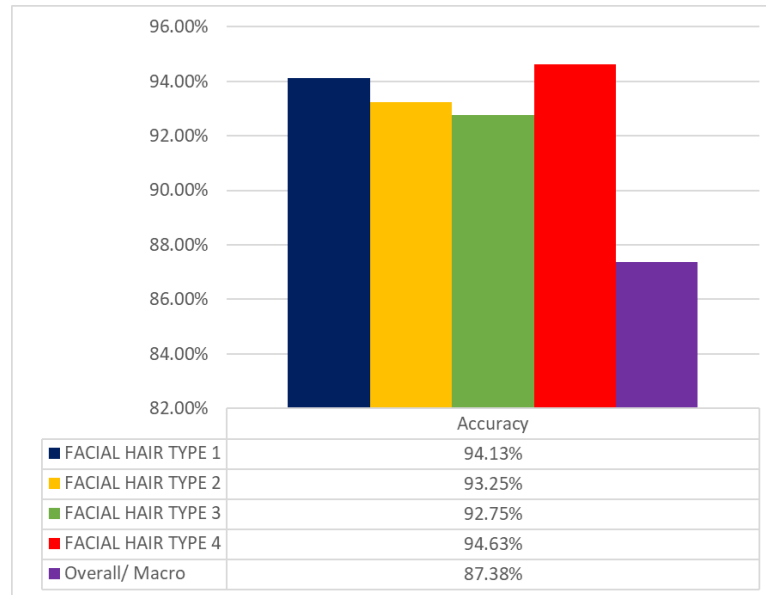


Figure 194 Accuracies Comparisons for Each Facial Hair Type

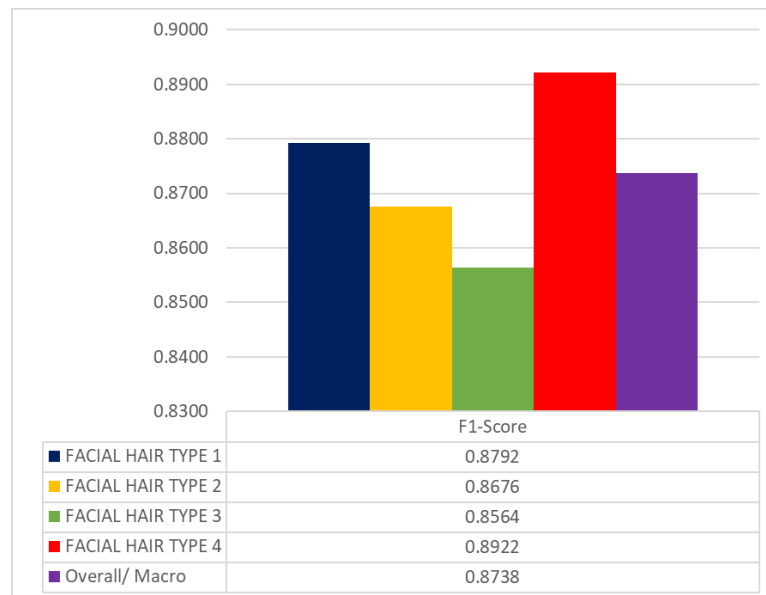


Figure 195 F1 Score Comparisons for Each Facial Hair Type

II. Facial Hair Classification Results using Custom Convolution Network

(A) ROC and Precision vs Recall Curve

Multiclass classification ROC plot between true positive rate and false positive rate for each facial hair type class for Custom CNN model is given in Figure 196. The performance of the facial hair type classification model and trade-off between true positive and false positive rates with varying the threshold of each class is estimated in this ROC curve. Figure 197 represents precision vs recall curve for all the classes, it demonstrates trade-off of the precision and recall based on threshold value. These curves represent the investigative ability of the classifier to classify facial hair type categories. Blue line represents performance of model for predicting facial hair type I, orange for facial hair type II, green for facial hair type III and finally the performance of facial hair type IV is represented by red line against multiple threshold levels.

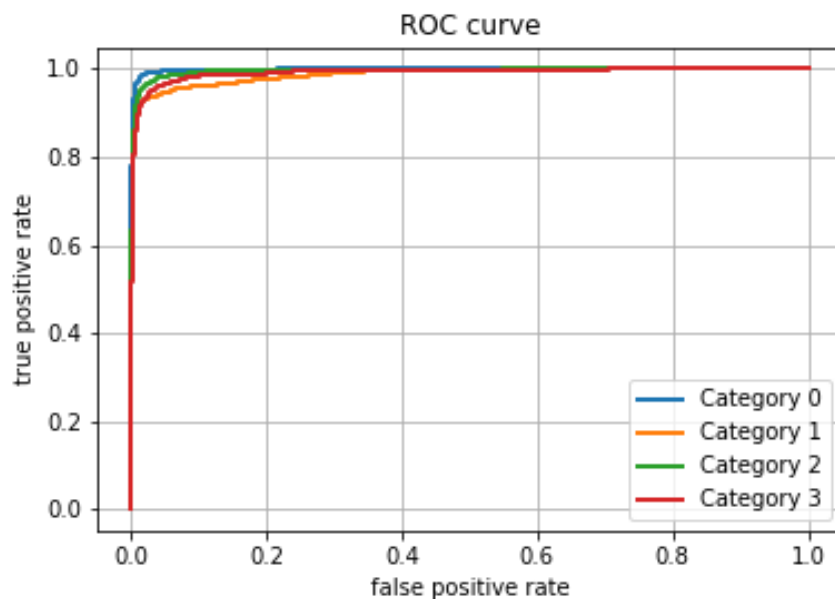


Figure 196 ROC Curve for Facial Hair Type Classification using Custom-Network

By looking at the blue line in the ROC curve, it shows that the model classifies facial hair type I more efficiently than others whereas it has difficulty in classifying facial hair type II correctly. Similar results can be seen in precision vs recall curve.

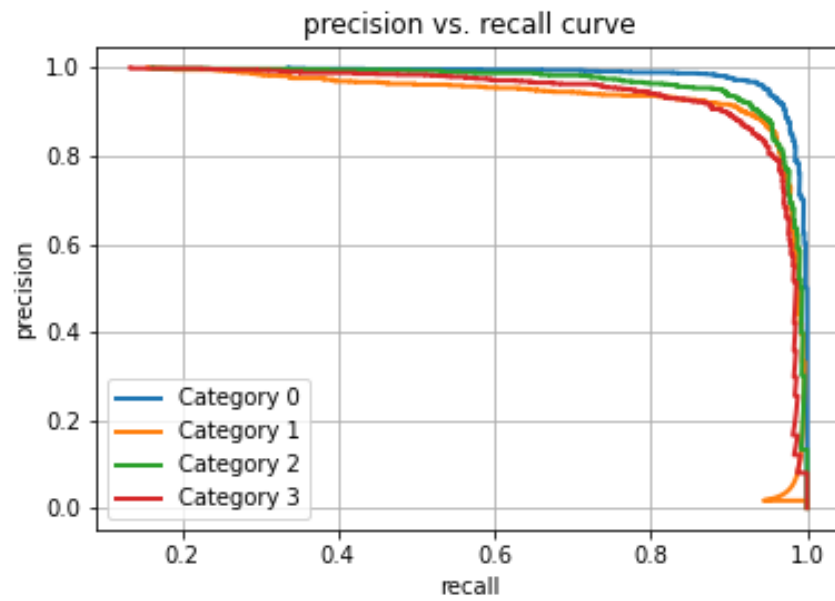


Figure 197 Precision vs Recall for Facial Hair Type Classification using Custom-Network

(B) Confusion Matrix

Confusion matrix representing precision and recall of each class is shown in Figure 198.

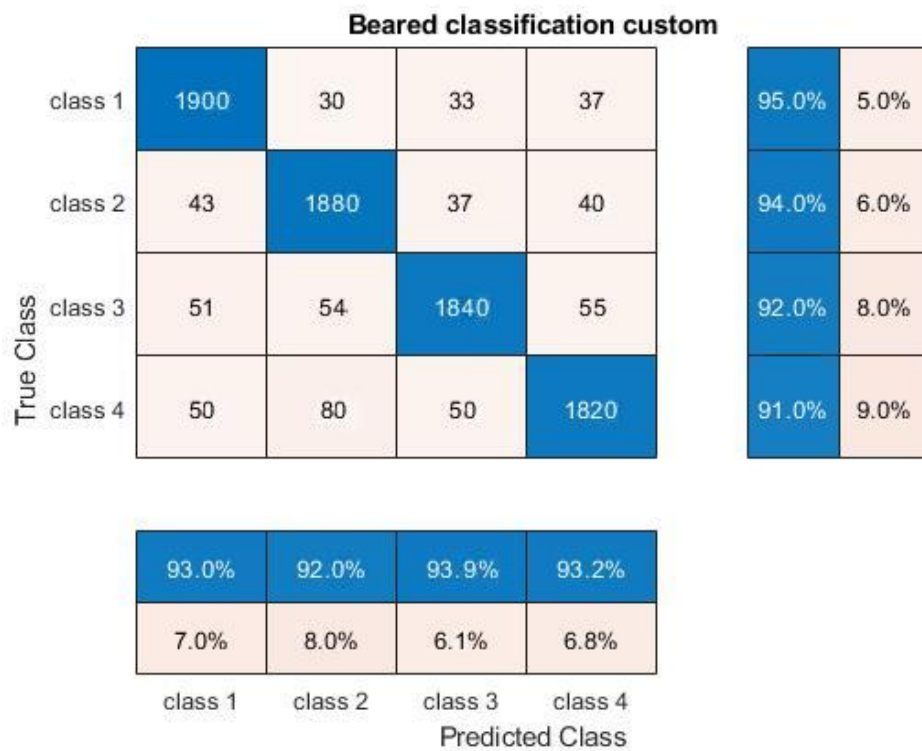


Figure 198 Confusion Matrix for Beard Classification using Custom Convolutional Network

(C) Precision, Recall, Accuracy and F1 Score

True Positives for Facial Hair Type using Custom Convolutional Network:

$$TP_{TYPE\ I} = 1900$$

$$TP_{TYPE\ II} = 1880$$

$$TP_{TYPE\ III} = 1840$$

$$TP_{TYPE\ IV} = 1820$$

$$TP_{OVERALL} = 7440$$

True Negatives for Facial Hair Type using Custom Convolutional Network:

$$TN_{TYPE\ I} = 5856$$

$$TN_{TYPE\ II} = 5836$$

$$TN_{TYPE\ III} = 5880$$

$$TN_{TYPE\ IV} = 5868$$

$$TN_{OVERALL} = 0$$

False Positives for Facial Hair Type using Custom Convolutional Network:

$$FP_{TYPE\ I} = 144$$

$$FP_{TYPE\ II} = 164$$

$$FP_{TYPE\ III} = 120$$

$$FP_{TYPE\ IV} = 132$$

$$FP_{OVERALL} = 550$$

False Negatives for Facial Hair Type using Custom Convolutional Network:

$$FN_{TYPE\ I} = 100$$

$$FN_{TYPE\ II} = 120$$

$$FN_{TYPE\ III} = 160$$

$$FN_{TYPE\ IV} = 180$$

$$FN_{OVERALL} = 560$$

Accuracy for Facial Hair Type using Custom Convolutional Network:

$$Accuracy_{TYPE\ I} = 0.9695$$

$$Accuracy_{TYPE\ II} = 0.9645$$

$$Accuracy_{TYPE\ III} = 0.9650$$

$$Accuracy_{TYPE\ IV} = 0.9610$$

$$Accuracy_{OVERALL} = 0.9300$$

Precision for Facial Hair Type using Custom Convolutional Network:

$$Precision_{TYPE\ I} = 0.9295$$

$$Precision_{TYPE\ II} = 0.9198$$

$$Precision_{TYPE\ III} = 0.9388$$

$$Precision_{TYPE\ IV} = 0.9323$$

$$Precision_{MACRO\ AVERAGE} = 0.9301$$

Recall for Facial Hair Type using Custom Convolutional Network:

$$Recall_{TYPE\ I} = 0.9500$$

$$Recall_{TYPE\ II} = 0.9400$$

$$Recall_{TYPE\ III} = 0.9200$$

$$Recall_{TYPE\ IV} = 0.9100$$

$$Recall_{MACRO\ AVERAGE} = 0.9300$$

F1-Score for Facial Hair Type using Custom Convolutional Network:

$$F1\ Score_{TYPE\ I} = 0.9396$$

$$F1\ Score_{TYPE\ II} = 0.9298$$

$$F1\ Score_{TYPE\ III} = 0.9293$$

$$F1\ Score_{TYPE\ IV} = 0.9210$$

$$F1\ Score_{MACRO\ AVERAGE} = 0.9300$$

Specificity for Facial Hair Type using Custom Convolutional Network:

$$Specificity_{TYPE\ I} = 0.9760$$

$$Specificity_{TYPE\ II} = 0.9726$$

$$Specificity_{TYPE\ III} = 0.9800$$

$$Specificity_{TYPE\ IV} = 0.9780$$

(D) Results summary for Facial Hair Type classification using Custom-Network

Table 19 shows results summary for Facial Hair Type Classification using Custom Network.

Table 19 Results Summary for Custom Network for Facial Hair Classification

	True Positive	True Negative	False Positive	False Negative	Precision	Recall	Specificity	Accuracy	F1-Score
FACIAL HAIR TYPE 1	1900	5856	144	100	92.95%	95.00%	0.9760	96.95%	0.9397
FACIAL HAIR TYPE 2	1880	5836	164	120	91.98%	94.00%	0.9727	96.45%	0.9298
FACIAL HAIR TYPE 3	1840	5880	120	160	93.88%	92.00%	0.9800	96.50%	0.9293
FACIAL HAIR TYPE 4	1820	5868	132	180	93.24%	91.00%	0.9780	96.10%	0.9211

Figure 199 shows the graphical view of comparison of precisions of each type and the overall precision. Figure 200 represents the recalls of each type. Whereas Figure 201 shows the overall and each class accuracy. Final F1 scores comparison id given in Figure 202.

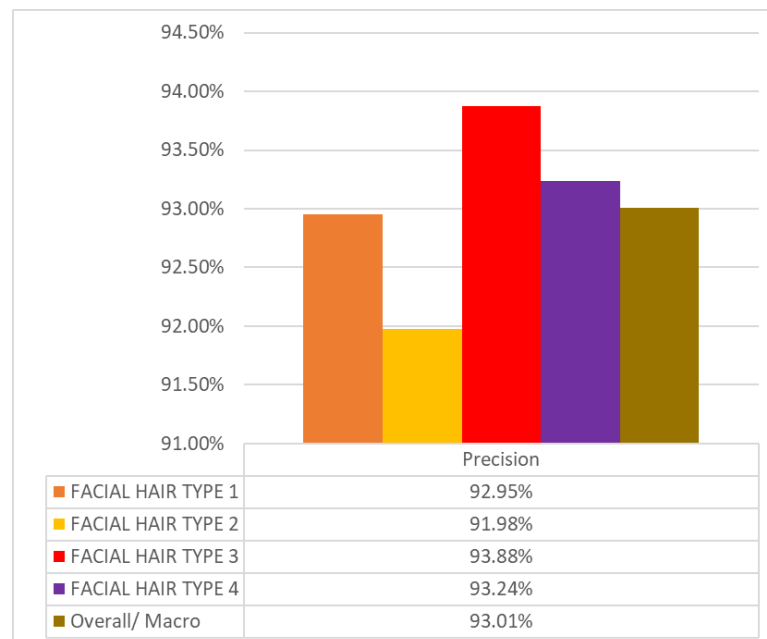


Figure 199 Precisions Comparisons for Each Facial Hair Type

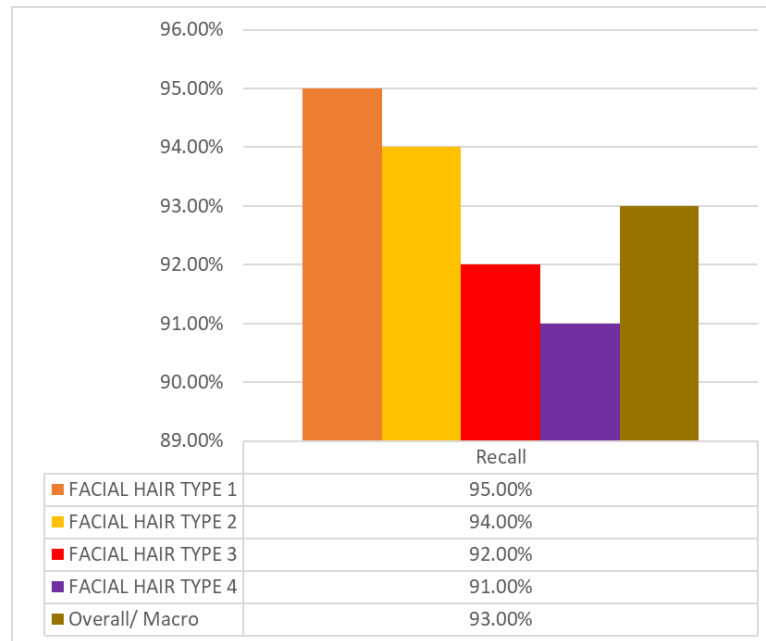


Figure 200 Recall Comparisons for Each Facial Hair Type

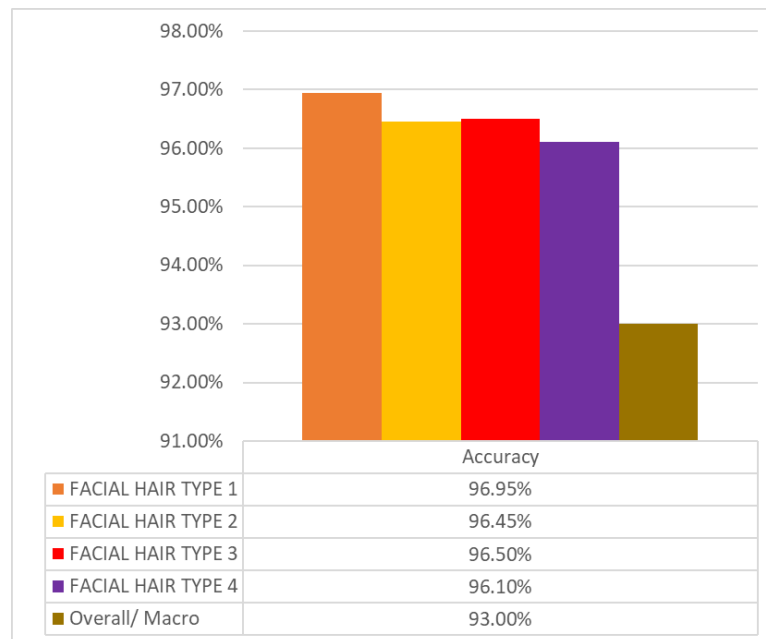


Figure 201 Accuracy Comparisons for Each Facial Hair Type

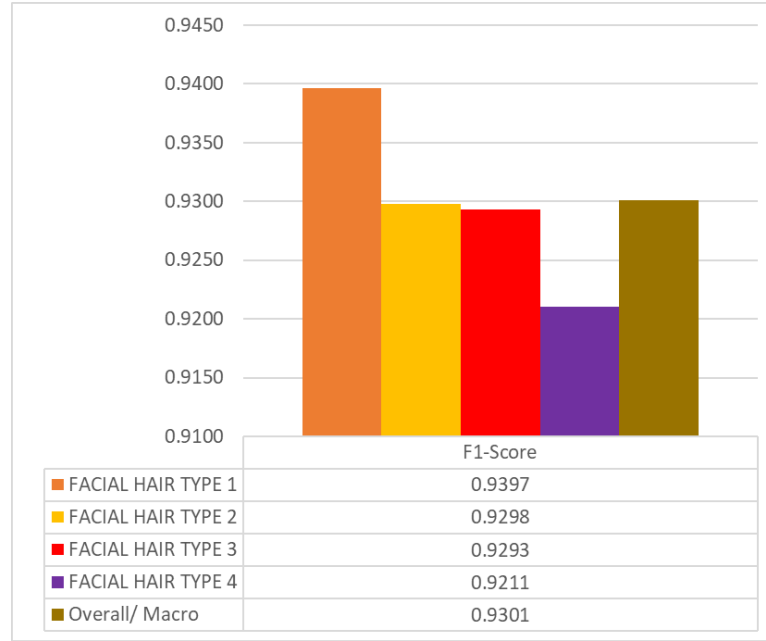


Figure 202 F1 Score Comparisons for Each Facial Hair Type

III. Facial Hair Classification Results using ResNet-50

For the task of facial hair classification problem, ResNet provide much better results than AlexNet, GoogLeNet and custom CNN model those were shown and explained earlier. The feedforward network is attached on convolution base of ResNet with 4 dense layers with neurons of 512, 256, 128 and 4 at output layer with SoftMax activation function. ReLU activation function is in each dense layer except the last one. Furthermore, the output of conv-base which is the feature matrix of input image is passed through flatten layer. To reduce overfitting effect L2 regularization and dropout layer is used with value of 0.2.

(A) ROC and Precision vs Recall Curve

Finally, the model is trained using ResNet-50 and multiclass classification ROC plot between true positive rate and false positive rate for each facial hair type class for is given in Figure 203. The trade-off between true positive and false positive rates with varying the threshold of each class is estimated in this ROC curve that depicts the performance of the model. Precision vs recall curve for all the classes is presented in Figure 204, it shows balance of the precision and recall based on threshold value. These curves represent the ability of the classifier to classify facial hair type categories. Blue line represents performance of model for predicting facial hair type I, orange for

facial hair type II, green for facial hair type III and finally the performance of facial hair type IV is represented by red line against multiple threshold levels.

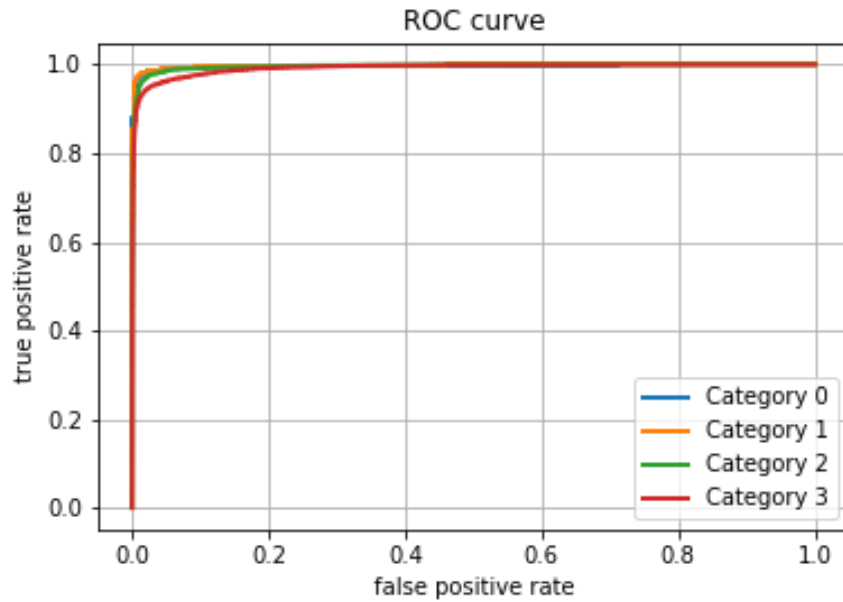


Figure 203 ROC for Facial Hair Type Classification using ResNet-50

The curves promising results and by looking at the ROC curve, it shows that the all the categories of facial hair types are classified by model well. Facial hair type I, II and III more classified efficiently than facial hair type IV. whereas it has difficulty in classifying facial hair type II correctly. Similar results can be seen in precision vs recall curve.

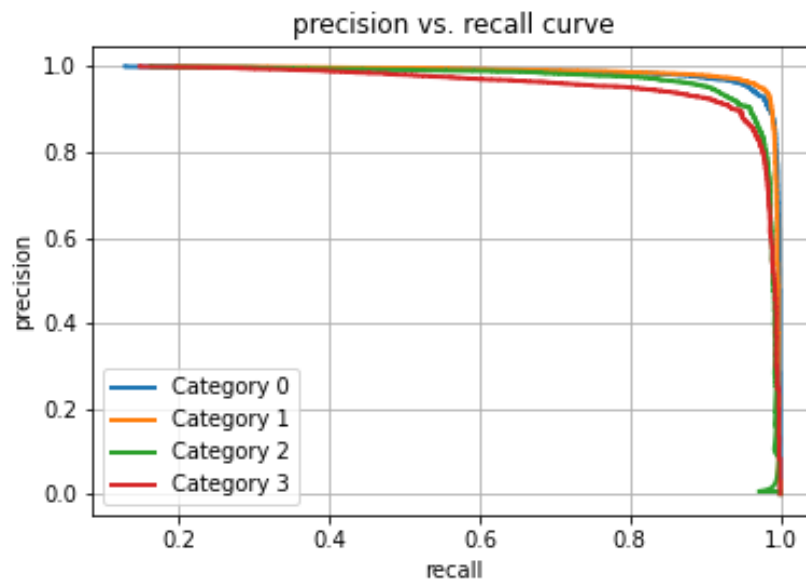


Figure 204 Precision vs Recall for Facial Hair Type Classification using ResNet-50

(B) Confusion Matrix

Confusion matrix representing precision and recall of each class is shown in Figure 205.

True Class	class 1	1937	10	26	27	96.9%	3.1%
	class 2	12	1950	13	25	97.5%	2.5%
	class 3	9	15	1956	20	97.8%	2.2%
	class 4	37	13	8	1942	97.1%	2.9%
		97.1%	98.1%	97.7%	96.4%		
		2.9%	1.9%	2.3%	3.6%		
		class 1	class 2	class 3	class 4		
		Predicted Class					

Figure 205 Confusion Matrix for Beard Classification using ResNet-50

(C) Precision, Recall, Accuracy and F1 Score

True Positives for Facial Hair Type using ResNet:

$$TP_{TYPE I} = 1937$$

$$TP_{TYPE II} = 1950$$

$$TP_{TYPE III} = 1956$$

$$TP_{TYPE IV} = 1942$$

$$TP_{OVERALL} = 7785$$

True Negatives for Facial Hair Type using ResNet:

$$TN_{TYPE I} = 5942$$

$$TN_{TYPE II} = 5962$$

$$TN_{TYPE III} = 5953$$

$$TN_{TYPE IV} = 5928$$

$$TN_{OVERALL} = 0$$

False Positives for Facial Hair Type using ResNet:

$$FP_{TYPE\ I} = 58$$

$$FP_{TYPE\ II} = 38$$

$$FP_{TYPE\ III} = 47$$

$$FP_{TYPE\ IV} = 72$$

$$FP_{OVERALL} = 215$$

False Negatives for Facial Hair Type using ResNet:

$$FN_{TYPE\ I} = 63$$

$$FN_{TYPE\ II} = 50$$

$$FN_{TYPE\ III} = 44$$

$$FN_{TYPE\ IV} = 58$$

$$FN_{OVERALL} = 215$$

Accuracy for Facial Hair Type using ResNet:

$$Accuracy_{TYPE\ I} = 0.9849$$

$$Accuracy_{TYPE\ II} = 0.9890$$

$$Accuracy_{TYPE\ III} = 0.9886$$

$$Accuracy_{TYPE\ IV} = 0.9837$$

$$Accuracy_{OVERALL} = 0.9731$$

Precision for Facial Hair Type using ResNet:

$$Precision_{TYPE\ I} = 0.9709$$

$$Precision_{TYPE\ II} = 0.9809$$

$$Precision_{TYPE\ III} = 0.9765$$

$$Precision_{TYPE\ IV} = 0.9643$$

$$Precision_{MACRO\ AVERAGE} = 0.9731$$

Recall for Facial Hair Type using ResNet:

$$Recall_{TYPE\ I} = 0.9685$$

$$Recall_{TYPE\ II} = 0.9750$$

$$Recall_{TYPE\ III} = 0.9780$$

$$Recall_{TYPE\ IV} = 0.9710$$

$$Recall_{MACRO\ AVERAGE} = 0.9731$$

F1-Score for Facial Hair Type using ResNet:

$$F1\ Score_{TYPE\ I} = 0.9697$$

$$F1\ Score_{TYPE\ II} = 0.9779$$

$$F1\ Score_{TYPE\ III} = 0.9772$$

$$F1\ Score_{TYPE\ IV} = 0.9676$$

$$F1\ Score_{MACRO\ AVERAGE} = 0.9720$$

Specificity for Facial Hair Type using ResNet:

$$Specificity_{TYPE\ I} = 0.9903$$

$$Specificity_{TYPE\ II} = 0.9936$$

$$Specificity_{TYPE\ III} = 0.9921$$

$$Specificity_{TYPE\ IV} = 0.9880$$

**(D) Results summary for Facial Hair Type classification
using ResNet-50**

Table 20 shows results summary for Facial Hair Type Classification using ResNet-50.

Table 20 Results Summary for ResNet for Facial Hair Classification

	True Positive	True Negative	False Positive	False Negative	Precision	Recall	Specificity	Accuracy	F1-Score
FACIAL HAIR TYPE 1	1937	5942	58	63	97.09%	96.85%	0.9903	98.49%	0.9697
FACIAL HAIR TYPE 2	1950	5962	38	50	98.09%	97.50%	0.9937	98.90%	0.9779
FACIAL HAIR TYPE 3	1956	5953	47	44	97.65%	97.80%	0.9922	98.86%	0.9773
FACIAL HAIR TYPE 4	1942	5928	72	58	96.43%	97.10%	0.9880	98.38%	0.9676

Figure 206 shows the comparison of precisions of each type and the overall precision. Figure 207 represents the recalls of each type. Whereas Figure 208 shows the overall and each class accuracy. Final F1 scores comparison id given in Figure 209.

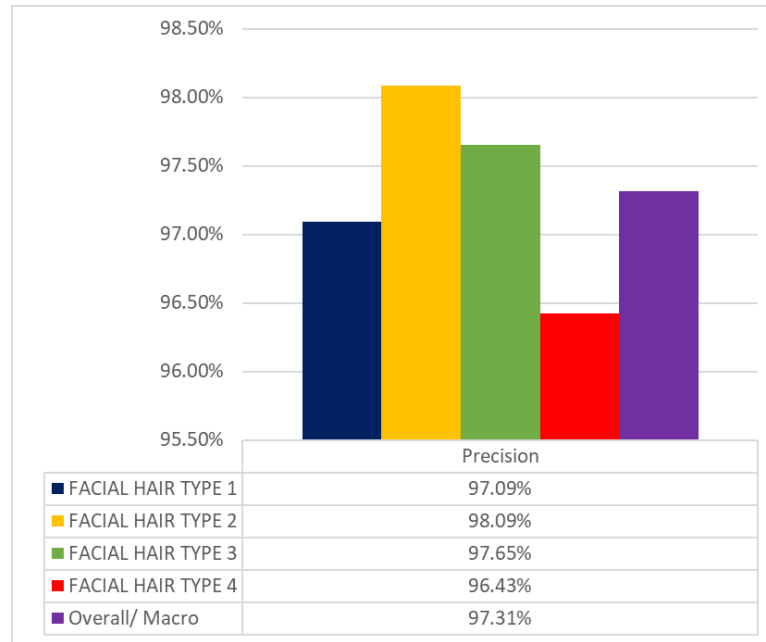


Figure 206 Precisions Comparisons for Each Facial Hair Type

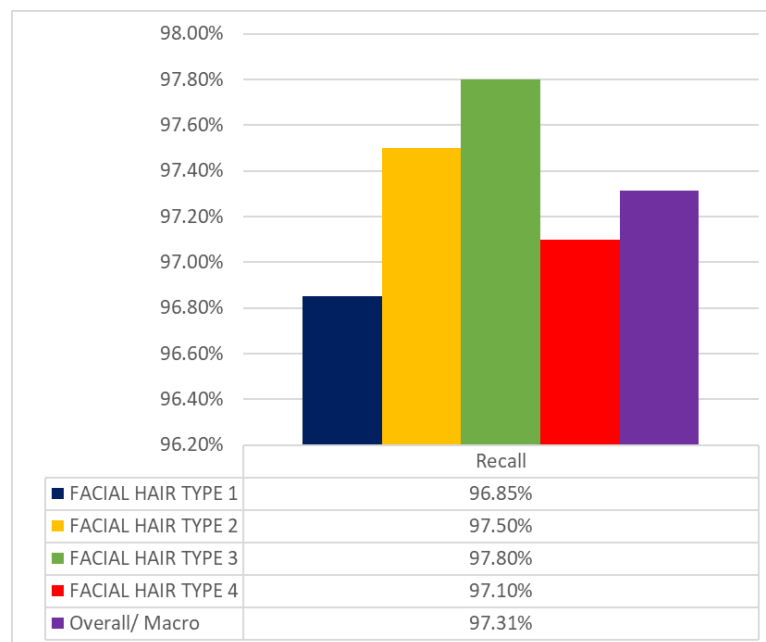


Figure 207 Recall Comparisons for Each Facial Hair Type

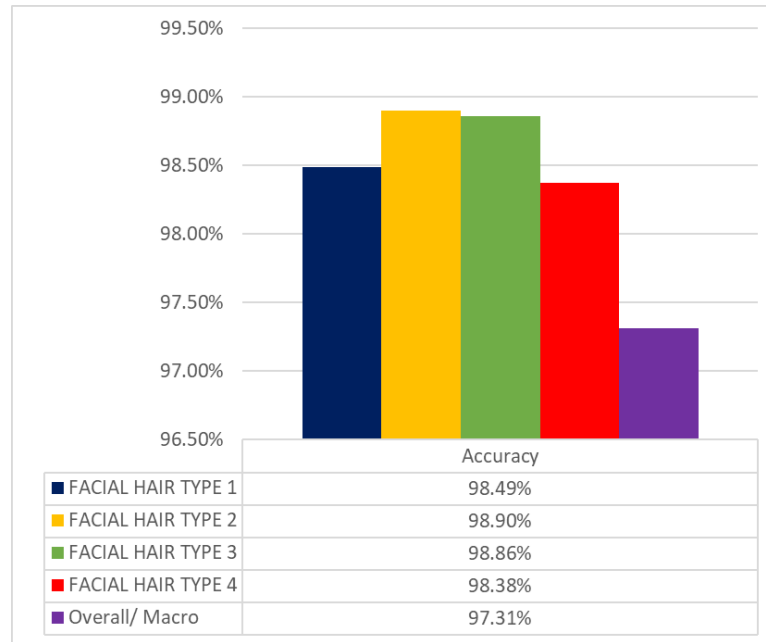


Figure 208 Accuracy Comparisons for Each Facial Hair Type

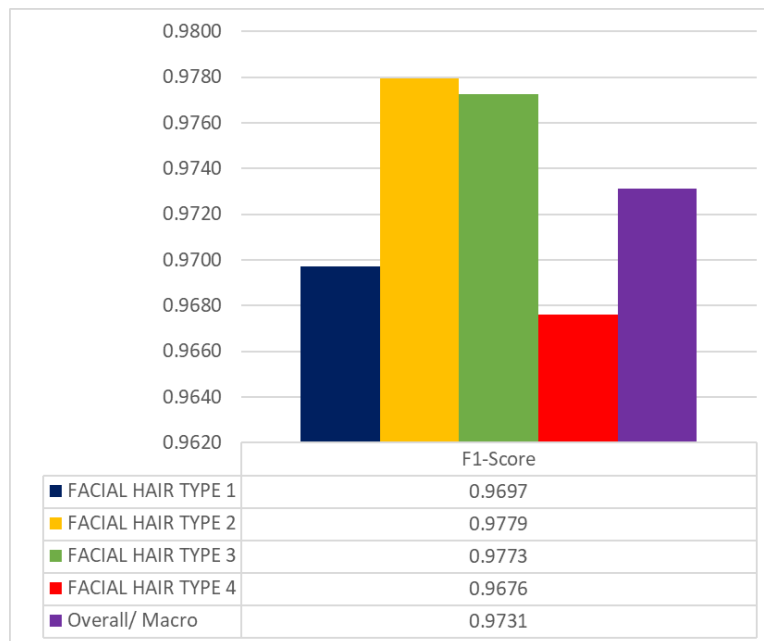


Figure 209 F1 Score Comparisons for Each Facial Hair Type

6.06 Face Alignment

The results on test data are given in Figure 210. Highly side pose face image is transformed into the front face image.

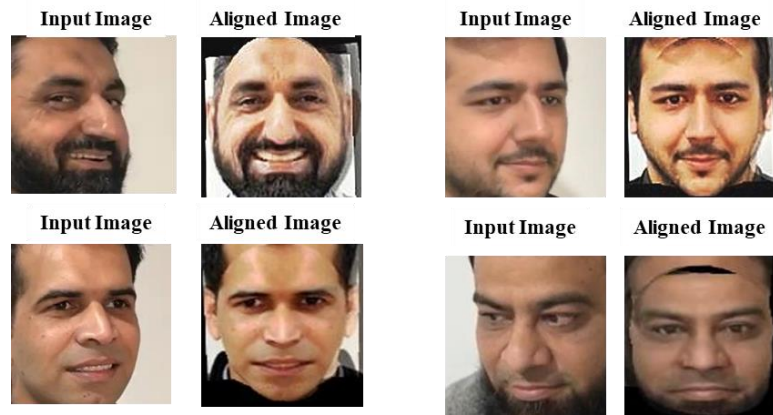


Figure 210 Face Alignment Results

From the results we can see that the result if face alignment performs impressive results. It can align a face to front pose with beard, moustache, and long beard accurately which is requirement for accurate facial recognition task.

6.07 Sketch to Hyper-Real Face Construction

For evaluating the performance of sketch to hyper-real image transformation, two methods were used. First method was to evaluate by visual inspection of the images. The second technique was to calculate MSE and RMSE.

I. Result Images

Figure 211 shows output of the model having input of sketch images and generated coloured images against each sketch image.

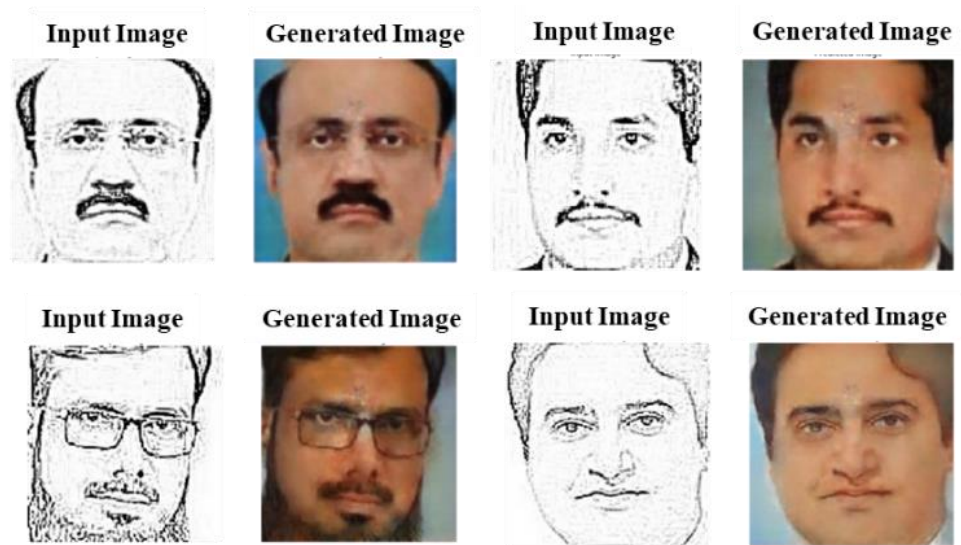


Figure 211 Sketch to Color Image Conversion Results

50 students of Artificial Intelligence class participated in a survey to compare the results of MS Cycle GAN and Cycle GAN. All these students are specialists and have multiple certifications in Artificial Intelligence. They had to rate the results for both the algorithms. Figure 212 shows the results of the poll. According to 27 participants MS Cycle GAN results are better.

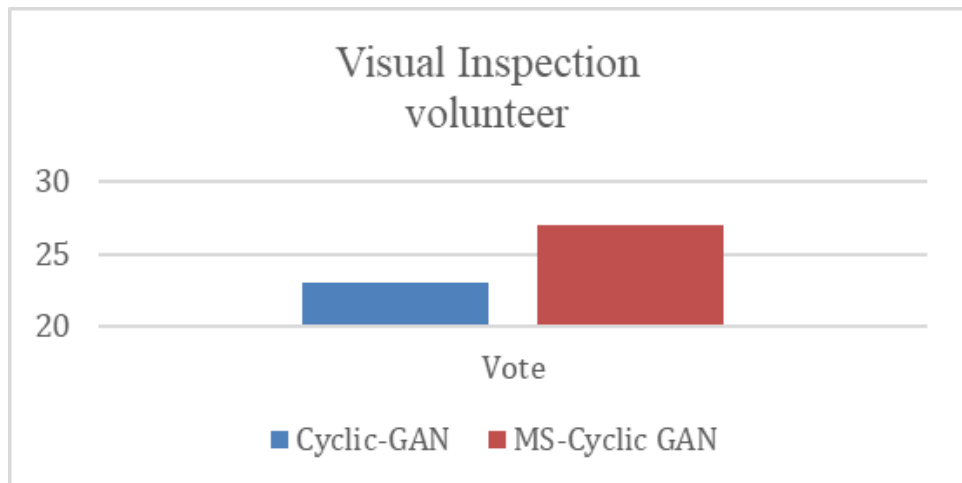


Figure 212 Visual Inception Poll Result

II. MSE and RMSE Results

MSE and RMSE results on produced and ground truth information likewise demonstrates the matchless quality of proposed strategy on Cycle GANs anyway the thing that matters isn't a lot bigger. MSE is determined pixel by pixel. So, in this manner a little balanced in picture can gravely influence the general outcomes. To

determine this issue on some degree, produced picture were manually adjusted to ground truth and foundations of pictures were additionally taken out and work out the MSE and RMSE after that. The correlation results are displayed in Table 21

Table 21 Pixel-Pixel Comparison Results for MSE and RMSE

Technique	Color to Color		Sketch to Sketch	
	MSE	RMSE	MSE	RMSE
MS Cycle-GAN	2.23	1.49	1.81	1.35
Cycle-GAN	2.31	1.52	1.90	1.38

The general result of this MS-CGAN network gives a great practical picture that is exceptionally near the ground truth. From the impressive outcomes, it tends to be seen that such kinds of organizations are exceptionally valuable for picture interpretation and make strong planning capabilities among info and result. The utilization of such models in the criminological world can create radical changes.

6.08 Blur Image to High Resolution Image Conversion

Two techniques were employed to assess the performance of the low to high resolution image conversion. The first method of evaluation was MSE and RMSE and the second method was calculating Structural Similarity Index (SSIM) and Feature Similarity Index (FSIM). Images were downgraded to lose the pixel densities and then passed through the model. Re-generated images by the model from blur images were compared with the original images to find the above-mentioned quantities.

I. Result Images

This algorithm has capability to estimate a higher resolution image from a blur/ low resolution image. Figure 213 shows couple generated image against low resolution input images.

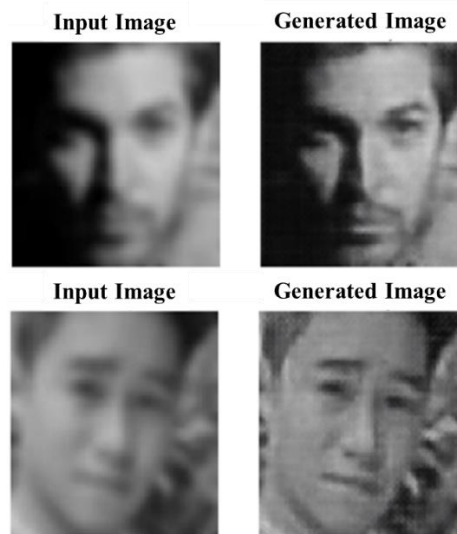


Figure 213 Low Resolution to High Resolution Image Generation Using MS Cycle GAN

II. MSE and RMSE Results

MSE and RMSE results are calculated similarly as of the low resolution/blur to high resolution. The results are displayed in Table 22.

Table 22 Pixel-Pixel MSE and RMSE Results

	MSE	RMSE
Blur Image to High Resolution Image Conversion Results	3.45	1.85

III. SSIM and FSIM Results

Structural Similarity Index, also known as SSIM, is a tool used to assess how similar two photographs are. It is frequently employed in image processing as a metric to assess the quality of compressed images. The SSIM index compares structural details like texture, contrast, and brightness to determine how similar two images are to one another. It accounts for the variances in the image's structure and the sensitivity of the human to these variations. A result of 1 shows perfect similarity between the two images, with the SSIM index producing a number between -1 and 1. Applications for image and video compression, denoising, super-resolution, and computer vision frequently use the SSIM index. Phase congruency, edge information, and picture gradient are some other elements that are incorporated into the Feature Similarity Index Metric (FSIM) method, which is an extension of the SSIM approach. A score of

1 indicates complete resemblance between the two images, and the FSIM index generates a value between 0 and 1 [318].

Required calculations can be done directly done with the help of *structural_similarity* function from *skimage.metrics* library in Python. Mean SSIM (MSSIM) value for 400 images was found to be **0.43**. Mean Structural Dissimilarity (MDSSIM) was calculated to be **0.285**. FSIM can be computed in two folds, the first one is the local similarity map and then the pool similarity map. Overall, the FSIM formula is sophisticated and accounts for both structural and feature similarities between two images, making it a more reliable and accurate metric for evaluating image quality. Using *scikit-image* and *fsim* module. Mean FSIM (MFSIM) value for 400 images was found to be **0.57**. Table 23 shows the summary of results for Blur to High Resolution Image Conversion.

Table 23 Blur to High Resolution Image Conversion Result

	MSSIM	MDSSIM	MFSIM
Blur to High Resolution	0.43	0.285	0.57

6.09 Face Matching using Cosine Similarity

Features of the images those are already sorted into their respective folders are stored in the form of .pkl file in the relevant folders. 4096 features against each face image are extracted. This feature extraction is performed only once per face and are used for both gender and face type classification layers. Once the input image has been classified and asked to search from the relevant folder; cosine similarity is used to match the features of the input image with the features of all the faces available in the folder. 60% or above confidence level images are referred as matched where less than that are missed. By physical inspection, the last category is wrongly matched or showing 60% or above similarity with some other person. It means one person is matched with high confidence score with some other face along with his/her own face. Or missed the real one but matched to some other face image. Following shows the results for the feature matching of a CCTV footage face and search photo of a target. Test was performed on 2000 database images vs 100 input images and tabulated in Table 24.

Table 24 Face Match Results

Input Images	Database Images	Correctly Matched	Missed	Wrongly Matched
100	2000	98	2	14

Figure 214 shows the few examples test results for similarity.

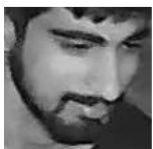
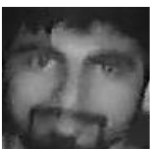
Input Image	Matched Image	Cosine Similarity	Confidence Score	Result
		0.1081	89.2%	Correct
		0.3685	63.2%	Correct
		0.2566	74.3%	Correct
		0.3374	66.3%	Correct
		0.1880	81.2%	Correct

Figure 214 Face Matching Scores Example

6.10 Summary

This section gives the summary of all the results showed in the chapter. For evaluation of YOLO v3 based face detection model multiple metrics are used. Mean IoU for the model was found to be 0.67. Average Precision for model 50% IoU was calculated as 0.856. Accuracy of the model was 90.27%, Precision was found to 92.69% and Recall was 97.18%, whereas F1 score was 0.9488.

POIRIT, demonstrates a notable advancement in the realm of detecting faces from CCTV images. Unlike previous research endeavours, POIRIT deals specifically with small-sized CCTV images and employs a distinctive training approach using degraded versions of images. This distinctive training methodology sets it apart from prior works. Additionally, POIRIT incorporates subsequent layers that automatically reject non-faces, which allows higher false positives at this stage. In terms of performance, POIRIT attains an accuracy of 90.7%.

Comparatively, previous studies have made commendable strides in the field. For instance, Ullah et al. (2022) [273] using principal component analysis (PCA) and convolutional neural networks (CNN) for feature extraction. led to an accuracy exceeding 90%. Another study by Halawa et al. (2019) [274] implemented the Faster-RCNN method, with their dataset featured a more limited scope, comprising six individuals with 50 face images per individual, they achieved a commendable detection accuracy of 89%.

Table 25 shows gender models results summary and comparison for all the models on the same dataset.

Table 25 Gender Models Performance Comparison Table

	AlexNet	GoogLeNet	VGG-Face
Precision	95.08%	94.86%	98.46%
Recall	95.01%	94.80%	98.36%
Accuracy	95.05%	94.84%	98.41%
F1-Score	0.9500	0.9484	0.9841

From the results the **VGG-Face** model has the highest precision of 98.46%, recall 98.36%, accuracy 98.41%, and F1-Score 0.9841 among the three models, near values of precision and recall shows models are performing equally good for both genders. Graphical representation of Table 25 is given in Figure 215.

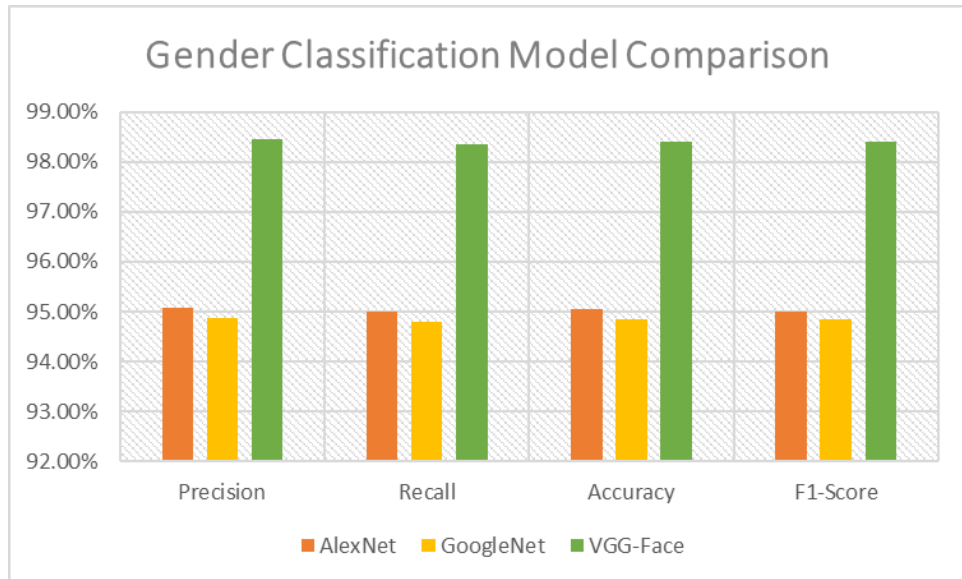


Figure 215 Gender Classification Model Comparison

POIRIT demonstrates impressive results in gender classification using small-sized CCTV images, achieving accuracies of 95.05% with AlexNet, 94.84% with GoogLeNet, and an impressive **98.60%** with VGG Face. In comparison, Thangaraj et. al. [2021] [286] achieved an accuracy of 97.4% in classifying genders in CCTV images using Inception V3. Haider et al.'s Deepgender [284], specializing in real-time gender classification using deep learning for smartphones, attained an accuracy of 97.09% and exhibits the capability to handle misaligned faces. Additionally, Kumar et al. [285] innovative algorithm for automatic live Gender Recognition employs SVM classification along with feature extractors like Bag of Words (Bow), SIFT, and K-means clustering. Their approach yielded efficient results across various standard datasets, achieving notably high detection rates: 98% in FEI dataset, 94% in Live/Own dataset, and 91% in SCIEN dataset, respectively. It's important to note that POIRIT had advantage that it deals with small-sized CCTV images and is trained on degraded versions of images sourced from Google image gallery, further emphasizing its noteworthy performance and can work on the live streams.

Table 26 shows face type models results summary and comparison of all the three deep learning models (AlexNet, GoogLeNet, and VGG Face) for face classification of four different types (FACE TYPE 1, FACE TYPE 2, FACE TYPE 3, and FACE TYPE 4). The models are evaluated using four common classification metrics: precision, recall, accuracy, and F1-Score.

Table 26 Face Type Models Performance Comparison Table

	FACE TYPE 1			FACE TYPE 2			FACE TYPE 3			FACE TYPE 4		
	AlexNet	GoogLeNet	VGG Face	AlexNet	GoogLeNet	VGG Face	AlexNet	GoogLeNet	VGG Face	AlexNet	GoogLeNet	VGG Face
Precision	93.21 %	92.44 %	96.06 %	87.02 %	85.21 %	98.17 %	89.61 %	90.04 %	96.74 %	89.23 %	90.48 %	97.88 %
Recall	89.95 %	88.70 %	98.70 %	90.50 %	90.75 %	96.80 %	90.95 %	91.35 %	96.35 %	87.45 %	86.95 %	96.95 %
Accuracy	95.85 %	95.36 %	98.66 %	94.25 %	93.75 %	98.75 %	95.10 %	95.31 %	98.28 %	94.23 %	94.45 %	98.71 %
F1-Score	0.9155	0.9053	0.9736	0.8873	0.8789	0.9748	0.9027	0.9069	0.9654	0.8833	0.8868	0.9741

Overall, the VGG Face model has the highest precision, recall, accuracy, and F1-Score across all face types, indicating that it is the best-performing model among the three. For Face Type I accuracy is 98.66%, precision is 96.06%, recall is 98.70%, and F1-Score is 0.9736. For Face Type II accuracy is 98.75%, precision is 98.17%, recall is 96.80%, and F1-Score is 0.9748. For Face Type III accuracy is 98.28%, precision is 96.74%, recall is 96.35%, and F1-Score is 0.9654. For Face Type IV accuracy is 98.71%, precision is 97.88%, recall is 96.95%, and F1-Score is 0.9741. Whereas the overall accuracy is 97.20%, precision is 97.21%, recall is 97.20%, and F1-Score is 0.9721. Figure 216 shows the comparison of precision, recall, accuracy and F1 score for Face Type I class for all three models.

Figure 216 shows the comparison of precision, recall, accuracy and F1 score for Face Type I class for all three models.

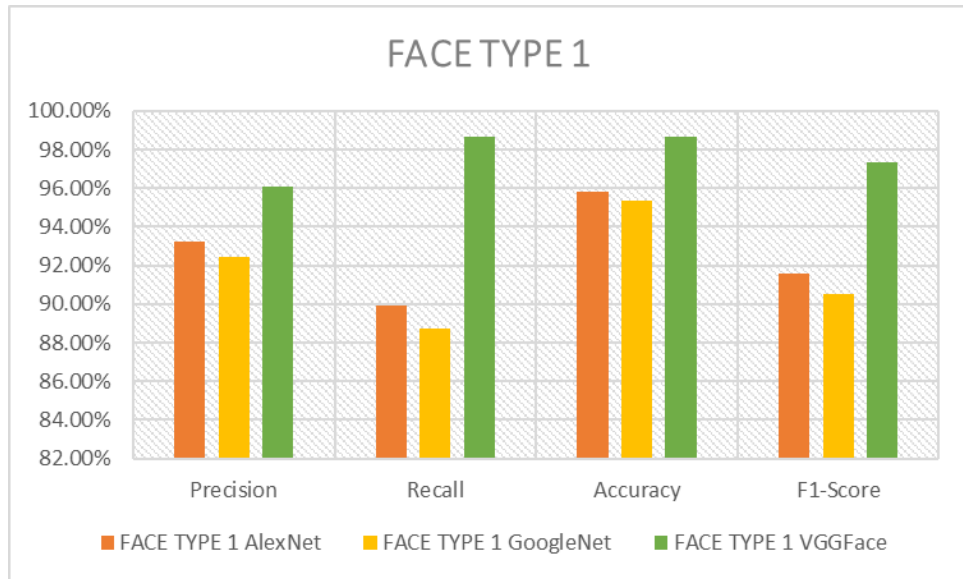


Figure 216 Face Type I Comparison

Figure 217 shows the comparison of precision, recall, accuracy and F1 score for Face Type II class for all three models.

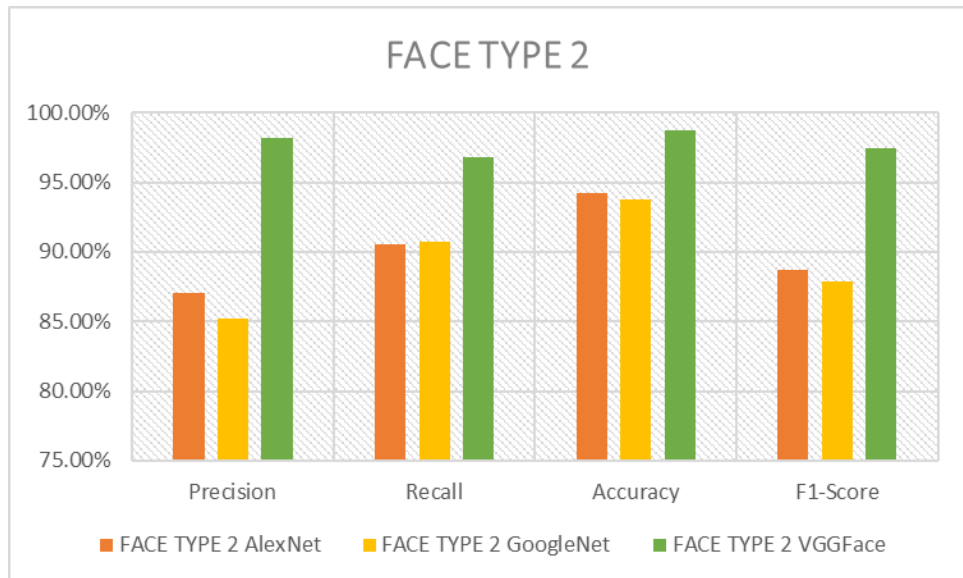


Figure 217 Face Type II Comparison

Figure 218 shows the comparison of precision, recall, accuracy and F1 score for Face Type III class for all three models.

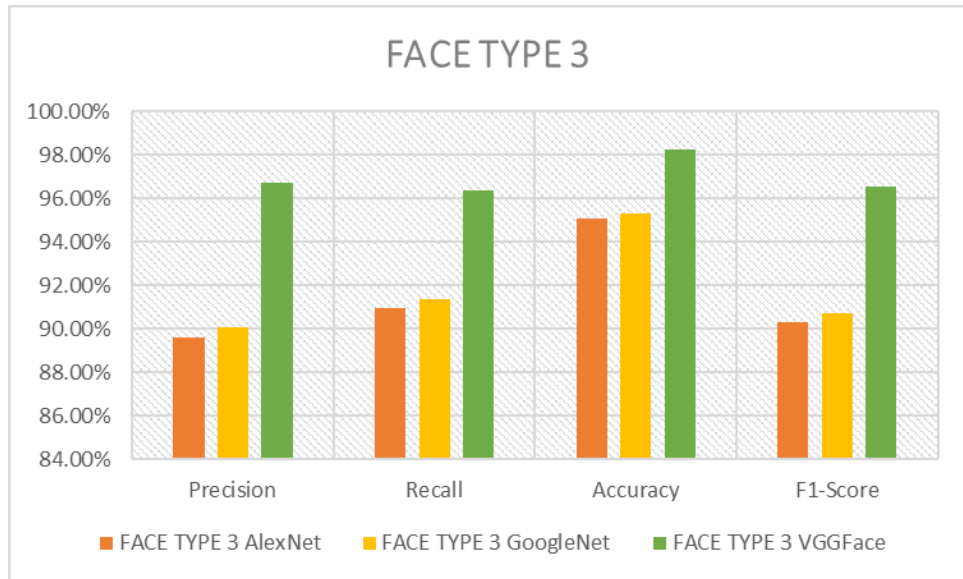


Figure 218 Face Type III Comparison

Figure 219 shows the comparison of precision, recall, accuracy and F1 score for Face Type IV class for all three models.

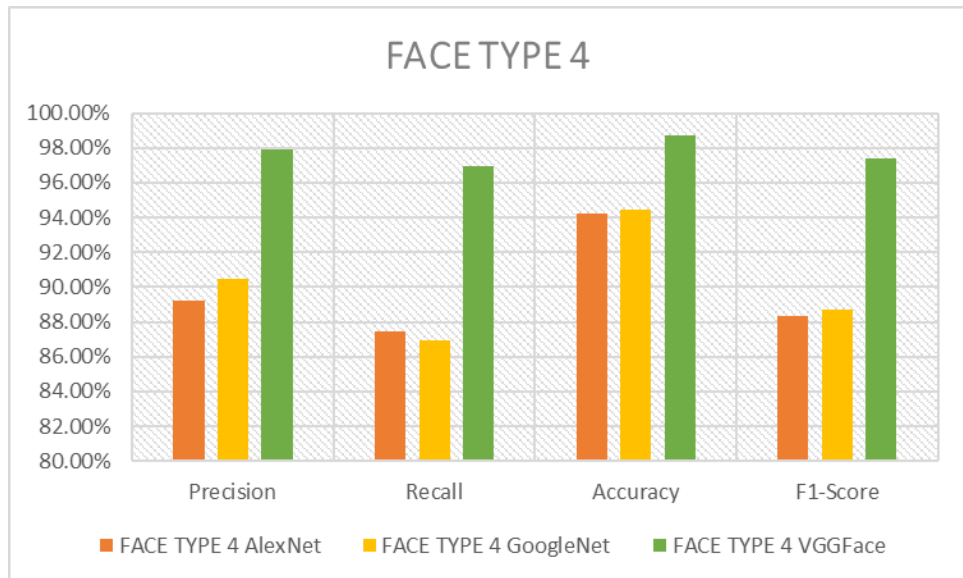


Figure 219 Face Type IV Comparison

Figure 220 shows the comparison of precision, recall, accuracy and F1 score for overall performance of all three models.

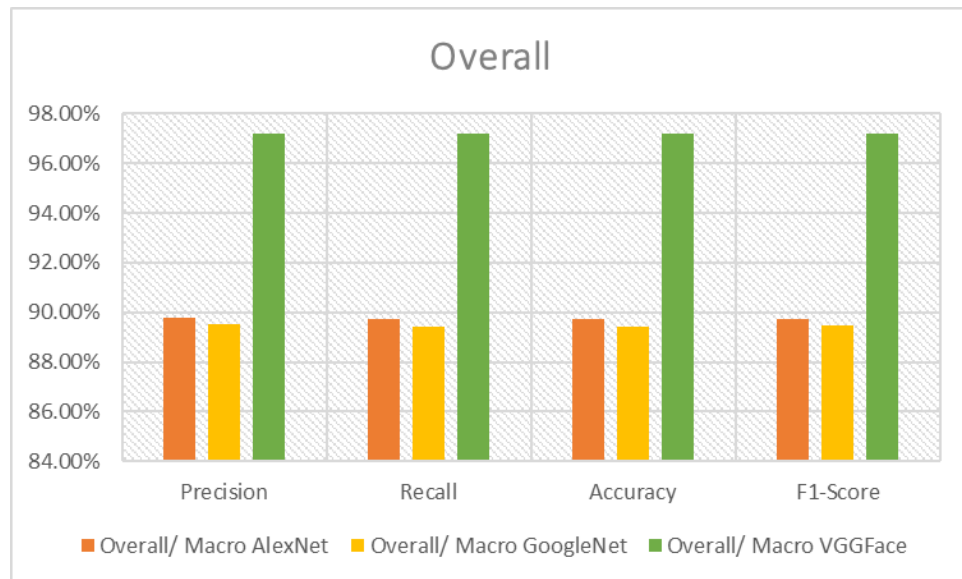


Figure 220 Overall Face Type Comparison

The results of the new research, utilizing the POIRIT methodology, represent a significant advancement in ethnicity classification, particularly for the four distinct ethnicities prevalent in Pakistan. These include Balti and Hazara people, Pashtun and Baloch people, Punjabi and Brahui people, and Makrani and South Indian race. The achieved accuracies are noteworthy: AlexNet demonstrated an accuracy of 94.86%, GoogLeNet followed closely with 94.72%, and VGG Face exhibited exceptional performance with an accuracy of **98.60%**. Therefore, selected as the final classification model for the system. In comparison, earlier studies discussed in 4.05, in the field of ethnicity classification employed a range of techniques, including PCA, SVM, facial feature discovery frameworks, Gabor filters, regression models, and deep convolutional neural networks (DCNNs), each applied on diverse datasets. While these studies achieved accuracies ranging from 88.8% to 92% for specific ethnic classifications trained on datasets like FERET, CAS-PEAL, VNFACE, and VMER to classify a broader range of races. The new research surpassed them by training models on indigenous datasets. This represents a significant leap forward in the accuracy and effectiveness of ethnicity classification, particularly in the context of the four prominent ethnicities residing in Pakistan.

Table 27 lists the performance comparison of three deep learning models (GoogLeNet, Custom Network and ResNet-50) for facial type classification into four different types (FACIAL HAIR TYPE 1, FACIAL HAIR TYPE 2, FACIAL HAIR

TYPE 3, and FACIAL HAIR TYPE 4). All the models are evaluated by training the same dataset on four models. AlexNet couldn't pass the criteria. Following table and graphs shows the comparison of precision, recall, accuracy and F1 score of each class, and overall values. The models are evaluated using four common classification metrics: precision, recall, accuracy, and F1-Score.

Table 27 Facial Hair Models Performance Comparison Table for Facial Hairs

	FACIAL HAIR TYPE 1			FACIAL HAIR TYPE 2			FACIAL HAIR TYPE 3			FACIAL HAIR TYPE 4		
	GoogLeNet	Custom Network	ResNet	GoogLeNet	Custom Network	ResNet	GoogLeNet	Custom Network	ResNet	GoogLeNet	Custom Network	ResNet
Precision	90.48 %	92.95 %	97.09 %	85.10 %	91.98 %	98.09 %	84.80 %	93.88 %	97.65 %	89.45 %	93.24 %	96.43 %
Recall	85.50 %	95.00 %	96.85 %	88.50 %	94.00 %	97.50 %	86.50 %	92.00 %	97.80 %	89.00 %	91.00 %	97.10 %
Accuracy	94.13 %	96.95 %	98.49 %	93.25 %	96.45 %	98.90 %	92.75 %	96.50 %	98.86 %	94.63 %	96.10 %	98.38 %
F1-Score	0.8792	0.9397	0.9697	0.8676	0.9298	0.9779	0.8564	0.9293	0.9773	0.8922	0.9211	0.9676

Overall, the ResNet model has the highest precision, recall, accuracy, and F1-Score across all face types, indicating that it is the best-performing model among the three. For Facial Hair Type I accuracy is 98.49%, precision is 97.09%, recall is 96.85%, and F1-Score is 0.9697. For Facial Hair Type II accuracy is 98.90%, precision is 98.09%, recall is 97.50%, and F1-Score is 0.9779. For Facial Hair Type III accuracy is 98.86%, precision is 97.65%, recall is 97.80%, and F1-Score is 0.9773. For Facial Hair Type IV accuracy is 98.38%, precision is 96.43%, recall is 97.10%, and F1-Score is 0.9676. Whereas the overall accuracy is 97.31%, precision is 97.31%, recall is 97.31%, and F1-Score is 0.9731. Figure 221 shows the comparison of precision, recall, accuracy and F1 score for Facial Hair Type I class for all three models.

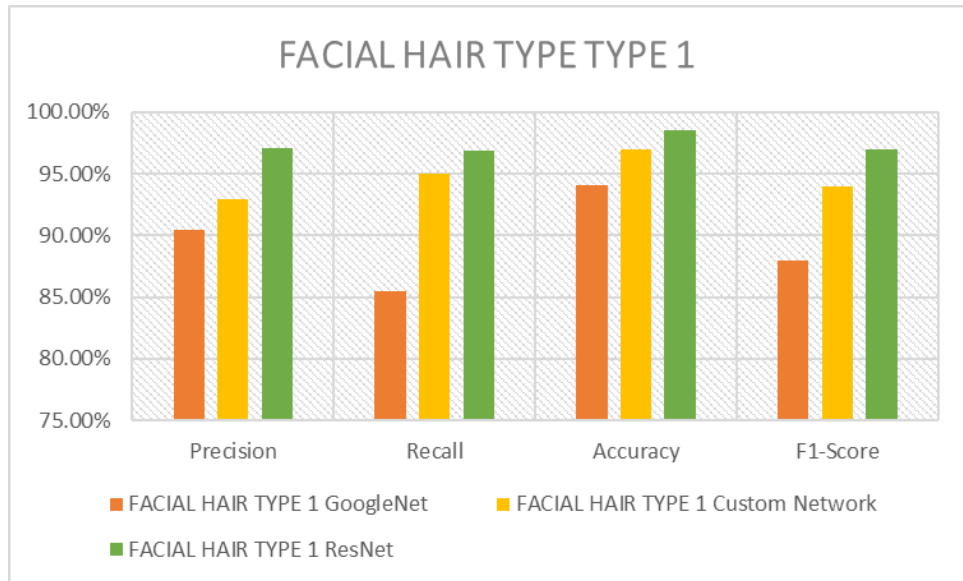


Figure 221 Facial Hair Type I Comparison

Figure 222 shows the comparison of precision, recall, accuracy and F1 score for Facial Hair Type II class for all three models.

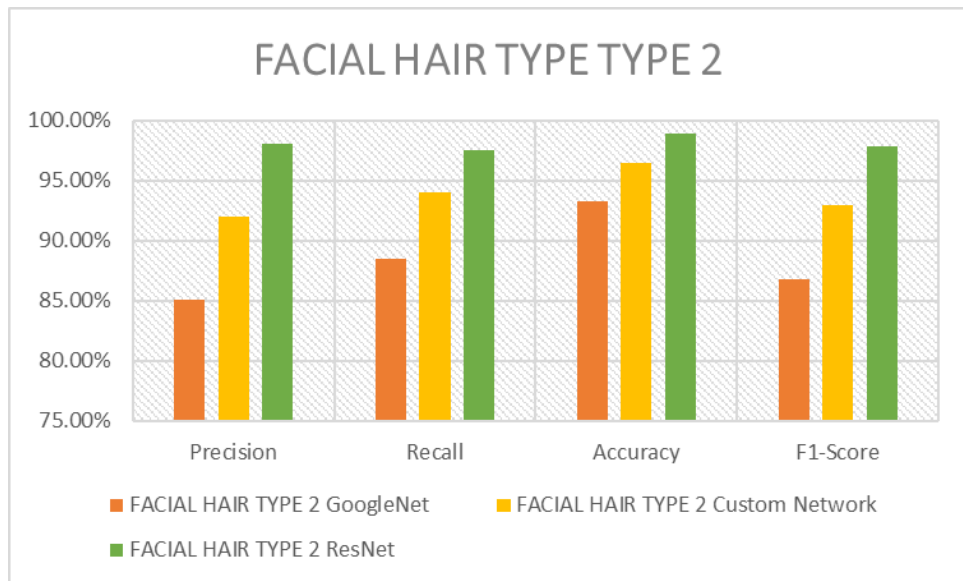


Figure 222 Facial Hair Type II Comparison

Figure 223 shows the comparison of precision, recall, accuracy and F1 score for Facial Hair Type III class for all three models.

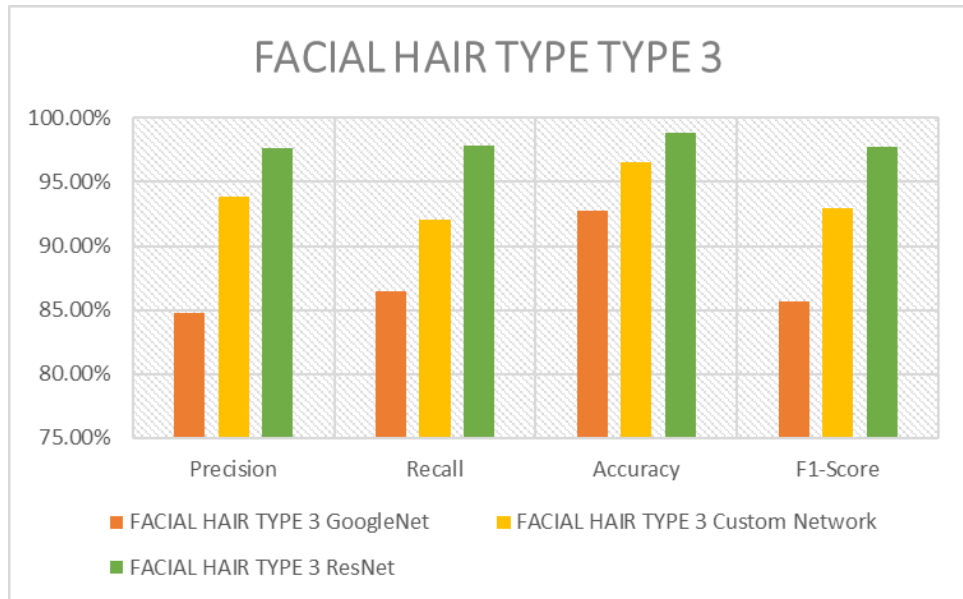


Figure 223 Facial Hair Type III Comparison

Figure 224 shows the comparison of precision, recall, accuracy and F1 score for Facial Hair Type IV class for all three models.

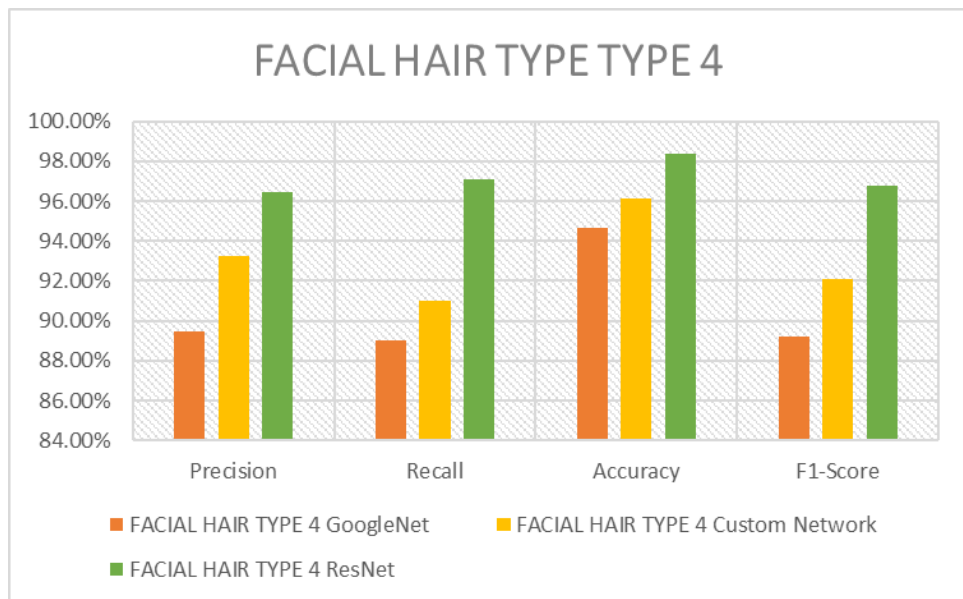


Figure 224 Facial Hair Type IV Comparison

Figure 225 shows the comparison of precision, recall, accuracy and F1 score for overall performance of all three models.

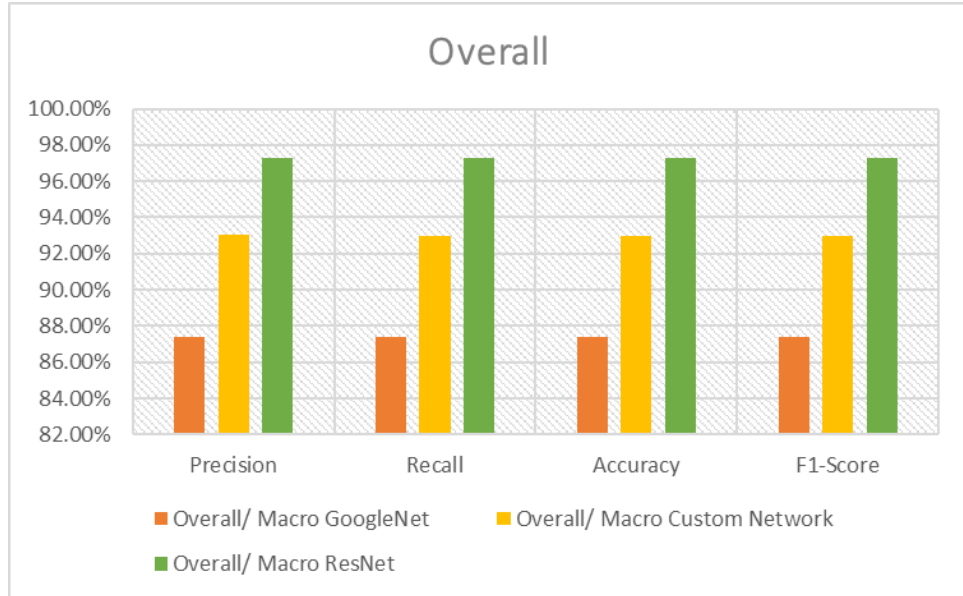


Figure 225 Facial Hair Overall Comparison

The proposed POIRIT system demonstrates remarkable performance in both detection and classification of facial hair types. Specifically, it achieves an accuracy of **98.66%** using ResNet-50, 96.50% using a custom network, and 93.69% using GoogLeNet. This significant accuracy demonstrates the effectiveness of the POIRIT system in identifying and classifying facial hair types. In comparison, previous research by Ubaid et. al. [303] focused solely on the detection of facial hair, achieving an accuracy of 91.2% using a modified version of the Mask R-CNN model. The POIRIT system not only outperforms in terms of accuracy but also adds the crucial aspect of classifying facial hair into distinct types, further enhancing its utility in gender identification, age assessment, and facial recognition applications. This distinction sets the POIRIT system apart from previous work, showcasing its advanced capabilities in facial hair detection and classification.

Two generative adversarial networks (GANs) - Cycle-GAN and MS Cycle-GAN - in two tasks for evaluation of sketch to hyper-real face construction. Pixel to pixel comparison for colour-colour and sketch-sketch translation were performed on test images. The evaluation metrics used were mean squared error (MSE) and root mean squared error (RMSE). In both tasks, MS Cycle-GAN outperformed Cycle-GAN, as it achieved lower MSE and RMSE values. Specifically, in the colour-colour task, MS Cycle-GAN achieved an MSE of 2.232 and an RMSE of 1.494, while Cycle-GAN achieved an MSE of 2.312 and an RMSE of 1.521. In the sketch-sketch task, MS

Cycle-GAN achieved an MSE of *1.81* and an RMSE of *1.345*, while Cycle-GAN achieved an MSE of *1.901* and an RMSE of *1.379*. For evaluating the Blur to High Resolution Transformation model, Mean Squared Error (MSE) and Root Mean Squared Error (RMSE) values were calculated for transformation from a blurred image to a high-resolution image using a MS Cycle-GAN. The MSE value of *3.45* and RMSE value of *1.85* indicate the average squared difference and root of squared difference between the predicted high-resolution image and the ground truth high-resolution image, respectively. Mean Structural Similarity Index (MSSIM) and Mean Feature Similarity Index (MFSIM) values calculated for a transformation of blurred image to a high-resolution image using MS-CGAN. The MSSIM value was found to be *0.43* and MFSIM value was *0.57*. Finally, cosine similarity index is found for the test images, and it was found that system can match the target images with *98%* accuracy.

Chapter 7

Conclusion and Future Work

7.01 Overview of the Thesis

Humans spend a large amount of money on safety and security in many ways. Pakistan is a third-world country hosting millions of Afghan refugees. This has led to multiple problems related to law and order. Moreover, it has been observed in recent years that recession hit all the countries of the world after COVID-19. An increase in prices and unemployment also caused street crime. These problems have made Pakistan vulnerable in terms of public safety. Therefore, the government started to take proactive measures that included increased patrolling and surveillance. A vast network of cameras has been laid in the cities. These systems are limited to manual surveillance resulting in inefficient and slow progress. Therefore, a cascaded system with the ability to detect, recognize, and search for a target is proposed in this research. The main goal of the research was to identify schemes to develop methods to efficiently search targets in less time. This research identifies the topologies to acquire images from various geographical locations efficiently and detect human faces from CCTV images. Evaluate how resizing the frames and identifying the region of interest from the face can reduce the computational costs. This research proposes cascaded classification layers to make an effective categorization system for the Indian subcontinent by using datasets for training deep learning models and comparing various Deep Learning algorithms for recognition systems. database that can be accessed by different users to search for a person of interest. Modern deep learning techniques along with image pre-processing were used. Multiple platforms like MATLAB and Python (TensorFlow, Keras, Tkinter, Panda, PyTorch, etc.) were used. Literature is included in Chapter 2. The methodology adopted in the research is explained in Chapter 3. Its emphasis on the need for this research and supported by the evidence that includes the relevant organizations' working mechanism and the public feedback. Ethical considerations were also considered and documented in the same chapter. In the same chapter, the novelty of the research was explained, and further methodology adopted for the conduct of the research was described. Chapter 4

describes the entire system design. It includes the selection criteria for the imaging devices. All the imaging devices are network based therefore the bandwidth estimation for the system was also discussed in this chapter. Being a distributed computing system, big data must be managed. The size of the face images in terms of occupying memory was also a big concern that was highlighted in the system design section. The contribution of the Raspberry Pi at the distributed nodes along with PIR sensors was elaborated. PIR pulse width calculations are also included here. Frame-level video quality settings, resizing, and frame naming rules were explained for stage I. Details of parallel computing stage II including the division of the zones to distribute tasks along with the computer systems with GPU devices were justified. An important part of the research was found here; the optimized size of the frame to detect faces without disturbing the accuracy of the system and identify the face cropping regions to select the areas to keep for the upcoming classification layers. All the cropped faces had to be the same for all the face images. It was also calculated here. Cascaded classification layers were highlighted, and dataset collection methods were discussed. Transfer learning techniques used in the research along with the layer setting were explained. Finally, the image search system features including a low-resolution to high-resolution image translation, sketch image-to-color image translation, and face alignment were described. In Chapter 5 experimentation, computing devices along with programming platforms used for the research are explained, it also shows various methods adopted including placement of the cameras; lighting conditions and pose variances to replicate multiple scenarios. Training and validation accuracies along with the training loss curves were also presented in this chapter. In Chapter 6, results for all the trained models were shown. Using the confusion matrix, the necessary calculations were done to find important parameters including accuracy, class, and macro average precision; class and macro average recall; and F1 scores. Precision vs Recall and ROC curves for each classification layer and model were also presented. Summary graphs showing inter-category and inter-model comparisons were given in the same section.

Chapter 7 proceeds further with a review of the important findings achieved in this investigation, followed by recommendations for future work.

7.02 Important Findings

A question-wise summary of the contribution of the thesis and important findings for research questions in Section 1.04 are given below.

I. How will images be acquired from various cameras placed at different geographical locations?

Section 4.02 shows the method adopted to create nodes with the help of network cameras with a decreased frame rate that is controlled by Raspberry Pi. PIR sensors are used in parallel to add a hardware layer to detect the frame change allowing the processor to work on other tasks. Following the same topology parallel nodes will work and upload images to the cloud. The performance of the system is speeded up by reducing the frame rate to **2 fps** for the frame capturing node (not the major stream) and the PIR trigger also eliminates the extra burden on the processing devices by avoiding creating redundant data.

II. How images from different resolutions of CCTV cameras will be resized to reduce the computational cost without affecting the performance?

Section 4.02 part I explains that an 8MP camera with a resolution of **3480 x 2160** was taken into consideration. The working distance was set to **50 feet** and the horizontal FOV was kept at **50 degrees**. The horizontal distance covered at 50 feet was calculated to be **46.63 feet** and verified by the IP Video System Design Tool. At the maximum distance, pixels per foot with the above arrangements were **83 PPF**. DORI rule was applied to resize the frame to **2792 x 1571** to attain **60 PPF**. By this operation, **47%** of memory and computations on pixels were saved. This value is for the mentioned parameters only. But by the equations given in the design, any CCTV camera stream can be optimized.

In section 4.03-part V it is shown that as established by the Hikvision research team, explained in the DORI rule, for detection of faces from a frame **20 PPF** was enough for the good results. Assuming the face at the maximum range the input for the detection of the face the frame was adjusted to **2161 x 1216** to get **46 PPF**. This resolution can be changed when the working distance or FOV is increased. Faces were

rescaled to **188 pixels** per face and an additional **18 pixels** on both sides. Making the face region of **224 x 224**. This system is designed to classify the targets by facial hair styles in addition to gender and face type. With the additional 63% pixels at the lower side, the final image to be stored was calculated to be **224 x 365**.

III. What region along with the face should be cropped and stored by keeping all the essential features?

Section 4.03 Part VI explains that how cropping the relevant face region out of the uploaded frame was calculated. Features of the face are present in the face width area but **bizygomatic** was considered. **10%** additional area was added on both sides of the face to compensate for the tilt. For vertical pixels, instead of **menton-sellion/ face length** additional area was required, which is why **Menton-crinion Length** was selected. An additional region of **63%** of the face length was added to the bottom of the face.

IV. What techniques will be used to detect human faces from CCTV images?

The size of the faces at a maximum distance is so small and a real-time system was to be proposed in this research. In the system design section, various object detection techniques are reviewed. That includes RCNN, Fast-RCNN, Faster-RCNN, YOLO v2 and YOLO v3. As explained in Section 4.03 Part IV, **YOLO v3** was used as it outperforms its predecessors while detecting small images. NVIDIA GPU was used while creating **ROIs** and training the model. Input image size for YOLO v3 is **2161 x 1216**, this image is divided into eight **608 x 608** images with additional padding on the right side. Training details and results are provided in Section 5.04.

As given in Section 6.10, POIRIT achieves an impressive accuracy rate of 90.7%. In comparison, prior studies discussed in Section 4.03, IV, have also demonstrated commendable achievements. Ullah et al. (2022) [273] using PCA and CNNs achieved an accuracy surpassing 90%. Meanwhile, Halawa et al. (2019) [274] employed the Faster-RCNN detection accuracy of 89% with a restricted dataset.

V. Which cascaded classification layers will be introduced to make an efficient categorization system for the Indian subcontinent?

In section 4.05, three layers for classification were introduced in this research work. These are evaluated based on the demographics of Pakistan. It is valid for any other region where multiple ethnicities live together. But this model is limited to Pakistan only. The first layer is for **gender**, the second one is for the **face type (ethnicity)** and the last is dedicated to **facial hair styles**.

For the face types, four face types were considered, Type I is for **Hazara and Balti** people. **Pashtun and Baloch** people are categorized as Type II. Type III represents **Punjabis and Brahuis**. Type IV characterizes the **Makrani and South Indian** races. **The upper 224 x 224 matrix** contains the face type information, so only this part is passed to the classification system. Details for the ethnicity selection for Pakistan are presented in Section 4.05-IV.

Four facial hair styles were proposed in Section 4.05-V as per the diversity common in the region. Faces with **no facial hairs** at all are Type 1, and Type II faces with **moustaches** only. Type III are **long beard** people and Type IV is for **short beard**. In Section 4.03-VI matrices cropping is explained. **Rhinion point** (mid nose) to the end of the 224 x 365 matrix that is lower **224 x 263** matrix is passed to the facial hair classification layer. This matrix is resized to **224 x 224** as per the input requirement of the network. This method allows multiple computing systems to classify the images. Only relevant data containing relevant features for the layer is passed.

VI. How datasets will be prepared for training deep learning models?

Section 4.03-IV explains the dataset preparation for Face Detection. The dataset consisted of 10,000 samples of single or multiple people in various frames, including streets and public places. Images from *Large Scale Multi-Camera Detection Dataset Kaggle*. *Wider Face* and *CUHK Person* are used in addition to these datasets, images are collected from the experimental setup. The frames were taken from CCTV video footage acquired from the experimental setup, with the consent of participants who were students at the University of the Punjab. ROI is generated with the help of *MATLAB Image Labeller*. *Viola-Jones* algorithm is used to automate the process,

whereas the missing ones are labeled manually. As explained in Section 4.04-III, for target classification, low-resolution to high-resolution and sketch to colour image conversion; open-source images from *Cleb_A Kaggle* and *Google* are used, and areas of interest are cropped from the images by the YOLO v3. For image transformation algorithms, sketch to hyper-real datasets were developed by transforming the coloured images into greyscale images, then Gaussian blur was applied to the images and then adaptive thresholding was applied to create binary images and finally the inverted images give sketch-like images. A similar approach was used for the development of blur/low-resolution images but does not include the last two steps.

VII. Which Deep Learning algorithm will be used for the recognition system?

This research proposed a cascaded system in Section 4.05 that has multiple layers for classification that help in sorting the images to relevant locations. As discussed in Section 5.05, various models for transfer learning like **AlexNet**, **GoogLeNet**, **ResNet**, and **VGG-Face** were evaluated on the collected dataset.

Section 6.03 shows detailed results for the gender classification, it was found that **VGG Face** shows the best results having **98.41%** accuracy. Other results can be viewed in the mentioned section. Comparison details can be seen in Summary Section 6.10. By comparison, the Thangaraj et al. [286] study achieved a 97.4% accuracy in gender classification using CCTV images with Inception V3. The Deepgender system by Haider et al. [284], attained a competitive 97.09% accuracy and demonstrated proficiency in handling misaligned faces. Furthermore, Kumar et al. [285], algorithm, employing SVM achieved detection rates of 98% in the FEI dataset, 94% in the Live/Own dataset, and 91% in the SCIEN dataset, respectively.

Section 6.04 provides details for the face type classification results, again VGG Face was found to be the best performer with an overall accuracy of **97.2%** and proved to be a better performer in other metrics as well. Class-wise accuracy was also found to be highest for VGG-Face. Class precision, recall, and F1 score are also given in the results summary. In contrast, earlier studies, as discussed in Section 4.05, adopted a range of techniques including PCA, SVM, facial feature discovery frameworks, Gabor filters, regression models, and deep convolutional neural networks (DCNNs), each

applied to diverse datasets. While these studies achieved accuracies spanning from 88.8% to 92% for specific ethnic classifications trained on datasets like FERET, CAS-PEAL, VNFACE, and VMER to classify a broader range of races. The new research outperformed them by training models on indigenous datasets, signifying a substantial stride in the accuracy and efficacy of ethnicity classification, particularly within the context of the four prominent ethnicities residing in Pakistan.

The last layer of the system is facial hair type classification. VGG-Face is trained on the face images that extract 4096 features from a face. Here system must deal with non-face images. A dataset containing a lower matrix of a face was used to train **AlexNet**, **GoogLeNet**, **ResNet**, and **Custom CNN** as discussed in Section 5.05.

ResNet-50 outperforms all the other models. Results in Section 6.05 show an overall accuracy of **97.31%**. Comparison of precision, recall, and F1 score for all the models are given in Section 6.10. Prior research by Ubaid et al. [303] focused primarily on facial hair detection, achieving an accuracy of 91.2% using a modified Mask R-CNN model. POIRIT not only surpasses this accuracy but also introduces the critical dimension of classifying facial hair into distinct types, enhancing its applicability in gender identification, age assessment, and facial recognition endeavours. This distinct feature distinguishes POIRIT as an advanced system in facial hair detection and classification.

Each class layer returns the name of the class with the highest probability and the confidence level of each class. Images can be placed in multiple categories if the confidence level is close to others. Details are given in the form of a flowchart in Section 4.05-V.

VIII. What image enhancements are required for the surveillance system and what techniques will be used to address the problem(s)?

Being an intelligent assistance system, it offers image enhancement capabilities. That includes **low-resolution/blur to high-resolution** image estimation. Images with a low resolution of **32x32** with blur can be transformed to **128x128** using the system. **MS Cycle GAN** was used in this regard. Section 4.06-III explains the working. Similarly, sketch images are converted into colour images using MS-CGAN explained in section 4.06-II. Another feature for **face alignment** is introduced to align the faces and

elaborated in Section 4.06-IV. MSE and RMSE were employed to quantify the discrepancies between the original and enhanced images. The pixel-to-pixel comparison results for MSE and RMSE between different techniques for colour-to-colour and sketch-to-sketch image transformations are presented in Section 6.07-II. For the colour-to-colour transformation, MS Cycle-GAN demonstrates superior performance with an MSE of **2.23** and RMSE of **1.49**, outperforming Cycle-GAN which yields slightly higher values of **2.31** for MSE and **1.52** for RMSE. Similarly, in the sketch-to-sketch transformation, MS Cycle-GAN exhibits better results with an MSE of **1.81** and RMSE of **1.35**, compared to Cycle-GAN which shows slightly higher values of **1.90** for MSE and **1.38** for RMSE. For resolution improvement, the results given in Section 6.08-I and 6.08-II, MSE and RMSE values of **3.45** and **1.85**, respectively, indicate a moderate level of error in the enhancement process. Additionally, MDSSIM and MFSIM were used to assess the similarity between the original and enhanced images. The enhancement technique demonstrated an MSSIM value of **0.43**, an MDSSIM value of **0.285**, and an MFSIM value of **0.57** given in Section 6.08, signifying a reasonable degree of structural similarity achieved through the process of converting blurred images to high-resolution counterparts.

IX. How a central database will be created to store cropped images and their attributes?

A central cloud was introduced to store frames, uploaded from all the nodes working at different geographical locations through high-speed internet. Initially, frames are stored in dedicated *frame folders* for each camera subdivided into date and hours. **960 KB** is to be written every second by each camera for a transformed frame image. Bitrate was calculated to be **7680 kbits per second** in Section 4.02-VIII. The bandwidth required for one camera was found to be **0.96 MB/s** in addition to the bandwidth requirement of the video stream from the camera. This database is accessible by high-performance computing devices powered by multiple GPUs. Faces are detected by these devices and stored in the *face database* further divided similarly as of the frames. For storing a jpeg image of **224 x 365** image consumes around **16 KB** of size on disk. The face categorization method is given in Section 4.05. Data gathered from the face detection nodes is stored in hourly folders and is subdivided into both

genders. Each gender has sub-folders for each **face type** (ethnicity). And finally, for all the male ethnicities, a folder for each **facial hair type** is present. High-performance computers access the cropped faces and firstly categorize the faces into genders, then classify them into ethnicity, and finally sort males by the facial hair styles. It helps users to find the target from the relevant field and save time.

X. How different users will access the database and search for a person of interest?

This research proposes a distributed computing system in which multiple computing devices like Raspberry Pi, CPUs, and GPUs can work together. All of them write data on the cloud. Frames are stored in the relevant folders for each camera for each day of the year in an hourly manner. That data is accessed by multiple systems to detect faces and classify the faces into above mentioned categories. Separate folders for each category and subcategory are created and accessible by all the computing systems working in parallel. It is explained in detail in Section 4.05.

Fog and Smog are crucial factors in the city under study which is Lahore, Pakistan. It has the worst AQI in the world. **Retinex Algorithm** in combination with **Wavelet transform** was used for fog removal. Histogram equalization, light control, etc. are also taken into consideration. Details can be found in Section 4.03-III

Person of interest can be searched by passing the input image to all these layers. Starting point (Camera name) with time and date is passed. It starts searching the target from the camera and the category assessed by the system. It works radially and matches using **cosine similarity**. Details are available in Section 4.06.

7.03 Recommendations and Future Work

There is always an infinite amount of room for innovation and improvement in the field of emerging technologies. Upon the completion of this research, we must look forward and consider potential directions for future investigation and improvement. The recommendations and possible directions for future research are outlined in the following section. By pushing the envelope of what is currently possible, these proposals hope to expand the field of intelligent surveillance systems and image

processing. These discoveries are expected to inspire researchers in the future to pursue even more advanced and significant solutions.

I. Face Detection using Low-Powered Computing Devices

The proposed system uses Raspberry Pi for frame capturing and image transformation. This image is then uploaded to the cloud. The face detection task is done at the next stage by distributed systems having GPUs. Nvidia Jetson Xavier can replace Raspberry Pi and do the face detection task on real-time video streams. Recently Yolo v3 comparison has been made and found out that Jetson Xavier NS gives twice as much fps as Raspberry Pi 4B. Energy consumption is an issue with GPU-based devices that can also be figured out in future work [319]. This extended work can increase the efficiency of the system in terms of time by adding another task to distributed computing with a little increased cost for the system.

II. Inclusion of Transgenders in Genders

Lahore, Pakistan is a city where a big population of transgenders live. They are an important part of society and have their separate respected identity in Pakistan. As per the “Transgender Person (Protection of Rights) Act 2018” Chapter II, clause 3 states that *“Recognition of identity of Transgender person. - (1) A transgender person shall have a right to be recognized as per his or her self-perceived gender identity, as such, by the provisions of this Act.”* According to the statement, a new class for the classification of genders can be added as an extended work.

III. Expanding Applicability to Diverse Ethnicities and Regions

This research is specifically providing solutions to Pakistan. Its applicability is limited to this region only. Other researchers can extend this work to many other nations having a mix of ethnicities living together. Moreover, ethnicities that are in the minority in Pakistan can be added to the same scheme.

IV. Enhancements for Face Mask Detection

Additionally, in terms of thesis work, four classes are identified for the facial hair classification. After COVID-19, rules and regulations have been changed in various parts of the world. At the face detection layer proposed system has only one class

which is face. Classes with face masks can be added and computed by YOLO v3. It has already been tested during this research but not included in the results to lessen the computational cost at processing stage II. As mentioned above if Nvidia Xavier NS is used then at the first stage this classification can be performed and a feature can be added.

V. Age Classification in Ethnicity-Specific Facial Recognition

Age is another parameter that classifies the targets. Further models can be trained for this purpose. Here one thing should be considered each ethnicity age group must be trained with the datasets of that ethnicity because age effects on facial features vary in ethnicities.

VI. Alternative Deep Learning Architectures for Enhanced Accuracy

In this thesis, architectures like AlexNet, GoogLeNet, ResNet, and VGG-Face were evaluated. Further researchers can extend this work by training models on other networks like *VarGFaceNet*, *FaceNet*, *VarGNet*, *CosFace*, etc., and increase accuracy.

VII. Expansions in Image Enhancements System

Two more features are added to this research to assist the user of this system. One was to create color images from sketch images. Datasets can be prepared by getting the sketch artist on board and training the system with the sketches from the archives of law enforcement agencies. MS Cycle GAN was used in research that does not use paired data. So, this can be achieved easily. Similarly, for low/blur to high-resolution image conversion, more accuracy can be achieved by improving the dataset. Similar techniques can be applied to create other features like face estimation of people wearing masks.

VIII. System Scalability through Participation of Public-Owned Devices

The proposed system can distribute the tasks in various computing systems located at different geographical locations dealing with the fixed camera nodes. Topology can be created where public-owned network CCTV cameras can be added to the main system and public-owned computing devices with GPUs installed can serve. An

encrypted system can be developed that can use the participant's computing power and in return, participants can earn money, like private solar houses feeding national grids and earning money.

IX. Enhanced Applications and Potential Extensions

By adopting this method new research can be performed to tag the identities/classes of all the people in an enclosed space. This can be used in shopping malls and residential compounds to find or track a target. Lost children and people with mental impairment can be tracked by using this system with minor modifications. Moreover, it can be extended to pick out the frame and apply gesture recognition on the tracked face and monitor its activity. Activity monitoring can also be implemented in prisons and shopping malls for market research.

Finally, it must be noted that this system and its future branches are all intelligent assistance systems that are to assist human workers, it would not be a cause of unemployment. Smart cities are the future of the world, and they require artificial intelligence.

REFERENCES

-
- [1] Akhtar, N., "Pakistan, Afghanistan, and the Taliban," *International Journal on World Peace*, pp. 49-73, 2008.
- [2] Kellner, D., "9/11, Spectacles of Terror, and Media Manipulation: A Critique of Jihadist and Bush Media Politics," *Critical Discourse Studies*, vol. 1, no. 1, pp. 41-64, 2004.
- [3] Mazarr, M.J., "The Folly of 'Asymmetric War'," *Washington Quarterly*, vol. 31, no. 3, pp. 33-53, 2008.
- [4] Cole, A., "Rules of Engagement Handbook," *International Institute of Humanitarian Law*.
- [5] Ali, A., "Economic Cost of Terrorism: A Case Study of Pakistan," *Strategic Studies*, vol. 30, no. 1-2, 2010.
- [6] Konar, R., Kumar, J., and Hussain, K., "Determinants of the Emerging Terrorist Threats to Luxury Hotels: A Case Study of Pakistan," *TEAM Journal of Hospitality and Tourism*, vol. 11, no. 1, pp. 41-53, 2014.
- [7] Khan, R.A., "Friendly Exchanges and People-to-People Contact Between Pakistan and India: Imperatives and Impediments," *Strategic Studies*, vol. 34, no. 2/3, 2014.
- [8] Khalid, I., and Roy, M.I., "Pakistan's Military Operations: The Counter-Terrorism Strategy (2001-2013) Prospects and Implications," *Journal of the Research Society of Pakistan*, vol. 53, no. 2, 2016.
- [9] Basit, A., "The Lahore Attack: Limits of Pakistan's Counter-Terrorism Policy," *RSIS Commentary*, 84., 2016.
- [10] Ahsan, S.J., "Lahore Attacked," *Defence Journal*, vol. 19, no. 9, p. 13, 2016.
- [11] Oxford Analytica, "China-Pakistan Corridor Faces Multiple Security Risks," *Emerald Expert Briefings* (oxan-db), 2017.

-
- [12] Bajoria, J., and Masters, J., "Pakistan's New Generation of Terrorists," *Council on Foreign Relations*, 2010.
- [13] McCarthy, N., "The Countries with The Most Terrorist Attacks," Accessed: June 5, 2017. [Online]. Available: <https://www.statista.com/chart/4969/the-countries-with-the-most-terrorist-attacks/>.
- [14] Ganor, B., "Artificial or Human: A New Era of Counterterrorism Intelligence?" *Studies in Conflict & Terrorism*, vol. 44, no. 7, pp. 605-624, 2021.
- [15] Punjab Safe Cities Authority, "About Us," Accessed: Oct. 9, 2017. [Online]. Available: <https://psca.gop.pk/PSCA/about/overview/>.
- [16] Yudkowsky, E., "Artificial Intelligence as A Positive and Negative Factor in Global Risk," *Global Catastrophic Risks*, vol. 1, p. 184, 2008.
- [17] Wang, L., Sng, D., "Deep Learning Algorithms with Applications to Video Analytics for A Smart City: A Survey," arXiv preprint arXiv:1512.03131, 2015.
- [18] Sajjad, M., Nasir, M., Muhammad, K., Khan, S., Jan, Z., Sangaiah, A. K., Elhoseny, M., Baik, S. W., "Raspberry Pi Assisted Face Recognition Framework for Enhanced Law-Enforcement Services in Smart Cities," *Future Generation Computer Systems*, [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0167739X17307832>.
- [19] LeCun, Y., Bengio, Y., Hinton, G., "Deep Learning," *Nature*, vol. 521, no. 7553, pp. 436-444, 2015.
- [20] Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet Classification with Deep Convolutional Neural Networks," *Advances in neural information processing systems*, 2012, pp. 1097-1105.
- [21] Al Ramahi, N., "Cameras with Facial Recognition Software Will Identify Wrongdoers in Dubai," [Online]. Available: <https://www.thenational.ae/uae/cameras-with-facial-recognition-software-will-identify-wrongdoers-in-dubai-1.699321>, 2018.

-
- [22] Sun, Y., "Meet the Company That's Using Face Recognition to Reshape China's Tech Scene," Accessed: Sep. 9, 2017. [Online]. Available: <https://www.technologyreview.com/s/608598/when-a-face-is-worth-a-billion-dollars/>
- [23] Loh, K. "Singapore Deploys Robot 'Dog' to Enforce Social Distancing." *Reuters*. Accessed: Nov. 2, 2023. [Online]. Available: <https://www.reuters.com/article/us-health-coronavirus-singapore-robot-idUSKBN21Y0AX>.
- [24] Transport for London. "CCTV on the TfL Network." Accessed: Nov. 2, 2023. [Online]. Available: <https://tfl.gov.uk/info-for/security-and-safety/cctv-on-the-tfl-network>.
- [25] New York City Police Department. "Domain Awareness System (DAS)." Accessed: Nov. 2, 2023. [Online]. Available: <https://www.nyc.gov/site/nypd/services/law-enforcement/technology.page>.
- [26] Seoul Metropolitan Government. "Seoul's Smart City." Nov. 2, 2023. [Online]. Available: <https://english.seoul.go.kr/seoul-to-rise-beyond-its-status-as-worlds-top-e-government-to-become-a-smart-city/smart-city/>.
- [27] Chen, X., Gong, S., and Lao, S., "Automatic Visual Tracking: A survey," *IEEE Transactions on Intelligent Transportation Systems*, vol. 13, no. 3, pp. 951-962, Sep. 2012.
- [28] Bouachir, O., Bilodeau, G. A., and Gagnon, G., "Urban Tracking in A Camera Network: A Statistical Approach," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 25, no. 5, pp. 868-881, May 2015.
- [29] Yi, K. M., Trulls, E., Onofre, J., Mordohai, P., and Torr, P. H., "3D Deep Learning for Efficient and Robust Object Pose Estimation in Cluttered Scenes," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017, pp. 2921-2930.
- [30] Luo, W., Yang, B., and Urtasun, R., "Fast and Furious: Real Time End-To-End 3D Detection, Tracking and Motion Forecasting with a Single Convolutional Net," in

Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2018, pp. 3569-3577.

[31] Huang, C., Wang, Y., and Neumann, U., "Context-Aware Visual Tracking," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 1394-1403.

[32] Myneni, S., Agrawal, G., Deng, Y., Chowdhary, A., Vadnere, N., and Huang, D., "SCVS: On AI and Edge Clouds Enabled Privacy-preserved Smart-city Video Surveillance Services," *ACM Transactions on Internet of Things*, vol. 3, no. 4, pp. 1-26, 2022.

[33] Bajwa, M. U., Khan, A., and Nadeem, M., "Empirical Study on Women Safety Concerns at Public Places: Case Study of Lahore City," *Architecture and urban planning*, vol. 14, no. 1, pp. 27-34, 2018.

[34] Crime Investigation Branch, Punjab Safe City Authority, 2019. [Online]. Available: <https://psca.gop.pk/>.

[35] Tariq, "Privacy Breach Fears Loom over Punjab Safe City Project," *Pakistan Today*, 2018. [Online]. Available: <https://www.pakistantoday.com.pk/2018/08/18/privacy-breach-fears-loom-over-punjab-safe-city-project/>.

[36] ACLU, "What's Wrong with Public Video Surveillance?" March 2002. [Online]. Available: <https://www.aclu.org/>.

[37] Lahore Safe City Authority, "About Us," 2019. [Online]. Available: <https://psca.gop.pk/AboutUs/Index>.

[38] Khan, J. A., "Lahore Safe City Authority Denies Pictures Being Circulated on Social Media Are from Their Cameras," *SAMAA NEWS*, 2019. [Online]. Available: <https://www.samaa.tv/news/2019/01/lahore-safe-city-authority-denies-pictures-being-circulated-on-social-media-are-from-their-cameras/>.

-
- [39] Gonzalez, R. C., Woods, R. E., "Elements of Visual Perception," *Digital image processing*.
- [40] Biederman, "Recognition-By-Components: A Theory of Human Image Understanding," *Psychol. Rev.*, vol. 94, pp. 115–147, 1987.
- [41] Haxby, V., Hoffman, E. A., Gobbini, M. I., "Human Neural Systems for Face Recognition and Social Communication," *Biological psychiatry*, vol. 51, no. 1, pp. 59–67, 2002.
- [42] Morris, S. et al., "A Differential Neural Response in The Human Amygdala to Fearful and Happy Facial Expressions," *Nature*, vol. 383, pp. 812–815, 1996.
- [43] Adolphs, R. et al., "Impaired Recognition of Emotion in Facial Expressions Following Bilateral Damage to The Human Amygdala," *Nature*, vol. 372, pp. 669–672, 1994.
- [44] Ikkai, T. A. Jerde, and C. E. Curtis, "Perception and Action Selection Dissociate Human Ventral and Dorsal Cortex," *J. Cogn. Neurosci.*, vol. 23, pp. 1494–1506, 2011.
- [45] Zahniser, D., *Digital Image Processing*, Keneth R. Castleman, Prentice-Hall, Inc., Englewood Cliffs, NJ, 1979.
- [46] Bovik, C., *Handbook of Image and Video Processing*, Academic press, 2010.
- [47] Chellappa, R., Wilson, C. L., Sirohey, S., "Human and Machine Recognition of Faces: A Survey," *Proceedings of the IEEE*, vol. 83, no. 5, pp. 705-741, 1995.
- [48] Turaga, P., Chellappa, R., Subrahmanian, V. S., Udrea, O., "Machine Recognition of Human Activities: A Survey," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 18, no. 11, pp. 1473-1488, 2008.
- [49] Gonzalez, R. C., Woods, R. E., "Light and Electromagnetic Spectrum," in *Digital image processing.*, 2012.
- [50] Leahy, R. M., Clackdoyle, R., Noo, F., "Computed Tomography," in *The Essential Guide to Image Processing*, pp. 741-776, 2009.

-
- [51] Merchant, F.A., Castleman, K.R., "Computer-Assisted Microscopy," in *The Essential Guide to Image Processing*, pp. 777-831, 2009.
- [52] Curry, G.R., "Radar Essentials: A Concise Handbook for Radar Design and Performance," *The Institution of Engineering and Technology*, 2012.
- [53] Gonzalez, R.C., Woods, R.E., "Imaging in Radio Band," in *Digital Image Processing*, 2012.
- [54] Du, E., Ives, R., van Nevel, A. and She, J.H., "Advanced Image Processing for Defence and Security Applications." *EURASIP Journal on Advances in Signal Processing*, 2010(1), pp.1-1.
- [55] Singh, P., Chadha, R.S., "A Survey of Digital Watermarking Techniques, Applications and Attacks," *International Journal of Engineering and Innovative Technology (IJEIT)*, vol. 2, no. 9, pp. 165-175, 2013.
- [56] Gupta, P., "Cryptography Based Digital Image Watermarking Algorithm to Increase Security of Watermark Data," *International Journal of Scientific & Engineering Research*, vol. 3, no. 9, pp. 1-4, 2012.
- [57] Jain, A., Hong, L., Pankanti, S., "Biometric Identification," *Communications of the ACM*, vol. 43, no. 2, pp. 90-98, 2000.
- [58] Ross, A., Nandakumar, K., Jain, A.K., "Introduction to Multibiometrics," in *Handbook of Biometrics*, pp. 271-292, 2008.
- [59] Sahidullah, M., "Enhancement of Speaker Recognition Performance using Block Level, Relative and Temporal Information of Subband Energies," Ph.D. dissertation, IIT Kharagpur, 2014.
- [60] Ahmad, F., Najam, A., Ahmed, Z., "Image-Based Face Detection and Recognition: State of the Art," *arXiv preprint arXiv:1302.6379*, 2013.
- [61] Owayjan, M., Dergham, A., Haber, G., Fakih, N., Hamoush, A., Abdo, E., "Face Recognition Security System," in *New Trends in Networking, Computing, E-learning, Systems Sciences, and Engineering*, pp. 343-348, Springer, Cham, 2015.

-
- [62] Find Biometrics, "Facial Recognition," Accessed: June 11, 2019. [Online]. Available: <http://findbiometrics.com/solutions/facial-recognition>.
- [63] InfoDif Information Diffusion, Accessed: Nov. 18, 2018. [Online]. Available: <http://www.infodif.com/en/solutions/military-applications>.
- [64] Dufour, J.Y., Martin, V., "Active/Passive Cooperative Image Segmentation for Automatic Target Recognition," in *Applications of Digital Image Processing XVII*, vol. 2298, pp. 552-561, International Society for Optics and Photonics, September 1994.
- [65] Wellfare, M.R., Norris-Zachery, K., "Characterization of Articulated Vehicles Using Ladar Seekers," in *Laser Radar Technology and Applications II*, vol. 3065, pp. 244-255, International Society for Optics and Photonics, August 1997.
- [66] Vasile, A.N., Marino, R.M., "Pose-Independent Automatic Target Detection and Recognition Using 3D Laser Radar Imagery," *Lincoln Laboratory Journal*, vol. 15, no. 1, pp. 61-78, 2005.
- [67] Ratches, J.A., Walters, C.P., Buser, R.G., Guenther, B.D., "Aided and Automatic Target Recognition Based Upon Sensory Inputs from Image Forming Systems," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 19, no. 9, pp. 1004-1019, 1997.
- [68] Schowengerdt, R.A., *Remote Sensing: Models and Methods for Image Processing*, Elsevier, 2006.
- [69] Olsen, R.C., *Remote Sensing from Air and Space*, vol. 229, SPIE Press, 2007.
- [70] Shah, V.V.D., "Image Processing and its Military Applications," *Defence Science Journal*, vol. 37, no. 4, pp. 457-468, 2014.
- [71] Jain, A.K., Ross, A.A., Nandakumar, K., "Face Recognition," in *Introduction to Biometrics*, pp. 97-139, Springer, Boston, MA, 2011.

-
- [72] Jain, A.K., Ross, A., Prabhakar, S., "An Introduction to Biometric Recognition," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 14, no. 1, pp. 4-20, 2004.
- [73] Jain, A.K., Ross, A.A., Nandakumar, K., *Introduction to Biometrics*, Springer Science & Business Media, 2011.
- [74] Jafri, R., Arabnia, H.R., "A Survey of Face Recognition Techniques," *Journal of Information Processing Systems*, vol. 5, no. 2, pp. 41-68, 2009.
- [75] Jain, A.K., Bolle, R., Pankanti, S., *Biometrics: Personal Identification in Networked Security*, Kluwer Academic Publishers, 1999.
- [76] Alim, M.A., Baig, M.M., Mehboob, S., Naseem, I., "Method for Secure Electronic Voting System: Face Recognition-Based Approach," in *Second International Workshop on Pattern Recognition*, vol. 10443, p. 104430H, International Society for Optics and Photonics, June 2017.
- [77] Aviad, R., Zamir, A., "System and Method for Interception of IP Traffic Based on Image Processing," U.S. Patent 9,742,812, 2017.
- [78] Oloyede, M.O., Hancke, G.P., Kapileswar, N., "Evaluating the Effect of Occlusion in Face Recognition Systems," in *AFRICON, 2017 IEEE*, pp. 1547-1551, IEEE, September 2017.
- [79] Naseem, I., Togneri, R., Bennamoun, M., "Linear Regression for Face Recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 32, no. 11, pp. 2106-2112, 2010.
- [80] Malusare, P., Shewale, S., Birla, S., Ghavate, S., Arote, M., "Face and Person Recognition from Unconstrained Video," *International Journal of Engineering Science*, 4295, 2017.
- [81] Vinay, A., Shekhar, V.S., Rituparna, J., Aggrawal, T., Murthy, K.B., Natarajan, S., "Cloud Based Big Data Analytics Framework for Face Recognition in Social

Networks using Machine Learning," *Procedia Computer Science*, vol. 50, pp. 623-630, 2015.

[82] Rawlinson, T., Bhalerao, A., Wang, L., "Principles and Methods for Face Recognition and Face Modelling," in *Handbook of Research on Computational Forensics, Digital Crime, and Investigation: Methods and Solutions*, p. 53.

[83] Nastar, C., Mitschke, M., "Real Time Face Recognition using Feature Combination," in *Third IEEE International Conference on Automatic Face and Gesture Recognition*, Nara, Japan, 1998, pp. 312-317.

[84] Yan, Ping, Kevin W. Bowyer, "Biometric Recognition using 3D Ear Shape," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 29, no. 8, pp. 1297-1308, 2007.

[85] Chen, Hui, Bir Bhanu, "Human Ear Recognition in 3D," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 29, no. 4, pp. 718-737, 2007.

[86] Jayachandra, C., Patel, B.S., "An Ear Recognition Approach Using Edge Detection," *Technology*, vol. 1, no. 1, pp. 38-45, 2013.

[87] Adini, Y., Moses, Y., Ullman, S., "Face Recognition: The Problem of Compensating for Changes in Illumination Direction," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 19, no. 7, pp. 721-732, 1997.

[88] Pentland, A., Choudhury, T., "Face Recognition for Smart Environments," *Computer*, vol. 33, no. 2, pp. 50-55, 2000.

[89] Daugman, J., "How Iris Recognition Works," in *The Essential Guide to Image Processing*, pp. 715-739, 2009.

[90] Wakamatsu, K., Hu, D.N., McCormick, S.A., Ito, S., "Characterization of Melanin in Human Iridal and Choroidal Melanocytes from Eyes with Various Colored Irides," *Pigment Cell & Melanoma Research*, vol. 21, no. 1, pp. 97-105, 2008.

[91] Daugman, J. G., "Biometric Personal Identification System Based on Iris Analysis," U.S. Patent No. 5,291,560, Mar. 1994.

-
- [92] Borgen, H., Bours, P., and Wolthusen, S. D., "Visible-Spectrum Biometric Retina Recognition," *Intelligent Information Hiding and Multimedia Signal Processing*, 2008. IIHMSP'08 International Conference on. IEEE, 2008, pp. 121-126.
- [93] Maltoni, D., Maio, D., Jain, A. K., and Prabhakar, S., *Handbook of Fingerprint Recognition*, Springer-Verlag, 2003.
- [94] Cole, S. A., "History of Fingerprint Pattern Recognition," in *Automatic Fingerprint Recognition Systems*. Springer New York, 2004, pp. 1-25.
- [95] Alice, I., "Biometric Recognition: Security and Privacy Concerns," *IEEE Security & Privacy*, 2003.
- [96] Maltoni, D., et al., *Handbook of fingerprint recognition*, Springer Science & Business Media, 2009.
- [97] Piuri, V., and Scotti, F., "Fingerprint Biometrics Via Low-Cost Sensors and Webcams," in *Biometrics: Theory, Applications and Systems*, 2008. BTAS 2008. 2nd IEEE International Conference on. IEEE.
- [98] Ramchandra, A., and Kumar, R., "Overview of Face Recognition System Challenges," *International Journal of Scientific & Technology Research*, vol. 2, no. 8, pp. 234-236, 2013.
- [99] Yang, M. H., Kriegman, D. J., and Ahuja, N., "Detecting Faces in Images: A Survey," *IEEE Trans. on PAMI*, vol. 24, no. 1, pp. 34-58, 2002.
- [100] Vincent, O. R., and Folorunso, O., "A Descriptive Algorithm for Sobel Image Edge Detection," in *Proceedings of Informing Science & IT Education Conference (InSITE)*, vol. 40.
- [101] Bakshi, U., and Singhal, R., "A Survey of Face Detection Methods and Feature Extraction Techniques of Face Recognition," *IJETTCS*, vol. 3, no. 3, 2014.
- [102] Rezq, O., and Al-Sayed, A., "Geometrical Approach for Face Detection and Recognition A Review: Face Detection Methods and Algorithms," *IJERT*, vol. 2, no. 6, 2013.

-
- [103] Shrivakshan, G. T., and Chandrasekar, Dr. C., "A Comparison of various Edge Detection Techniques used in Image Processing," *IJCSI International Journal of Computer Science Issues*, vol. 9, no. 5, 2012.
- [104] Turk, M., and Pentland, A., "Face Recognition using Eigenfaces," in *Conference on Computer Vision and Pattern Recognition*, 1991, pp. 586-591.
- [105] Paul, L. C., and Sumam, A. A., "Face Recognition using Principal Component Analysis Method," *International Journal of Advanced Research in Computer Engineering & Technology (IJARCET)*, vol. 1, no. 9, pp. 135-142, 2012.
- [106] Mita, T., Kaneko, T., and Hori, O., "Joint Haar-Like Features for Face Detection," in *Tenth IEEE International Conference on Computer Vision (ICCV'05)*, vol. 1, Beijing, 2005, pp. 1619-1626 Vol. 2.
- [107] Chen, Q., Georganas, N. D., and M. Emil, "Real-Time Vision-Based Hand Gesture Recognition using Haar-Like Features," in *Instrumentation and Measurement Technology Conference - IMTC 2007*, Warsaw, Poland, 2007.
- [108] Pandya, K. D., "Face Detection—A Literature Survey," *International Journal of Computer Techniques*, vol. 3, no. 1, Jan-Feb 2016.
- [109] Viola, P., and Jones, M. J., "Robust Real-Time Face Detection," *International Journal of Computer Vision*, vol. 57, no. 2, pp. 137-154, 2004.
- [110] Viola, P., and Jones, M. J., "Rapid Object Detection using a Boosted Cascade of Simple Features," in *Conference on Computer Vision and Pattern Recognition*, 2001.
- [111] Jain, V., Miller, E., Fddb: A benchmark for face detection in unconstrained settings. Vol. 2. No. 6. UMass Amherst technical report, 2010.
- [112] Finkelstein, A., Sharma, N., "Double Integrals by Summing Values of Cumulative Distribution Function," *Int. J. Comput. Tech* 3, no. 1 (2016): 67-70., 2010.
- [113] Binesh, T. R., and Simon, P., "Fast Pedestrian Detection using smart ROI separation and Integral image-based Feature Extraction," *IJCSE*, vol. 3, no. 11, 2012.

-
- [114] Gaikwad, V., Lokhande, S., Sanket, M., and Pravin, "New Improved Methodology for Pedestrian Detection in Advanced Driver Assistance System," *IJCA*, 2012.
- [115] Gupta, T., and Garg, L., "Face Recognition using SIFT," *International Journal of Emerging Technology and Advanced Engineering*, vol. 4, no. 5, pp. 358-363, 2014.
- [116] Pedersen, J. T., "SURF: Feature Detection & Description," Technická zpráva, AARHUS University, 2011.
- [117] Bay, H., et al., "Speeded-Up Robust Features (SURF)," *Computer vision and image understanding*, vol. 110, no. 3, pp. 346-359, 2008.
- [118] MathWorks, "vision.pointtracker-class," Accessed: Sep. 11, 2019. [Online]. Available: <https://www.mathworks.com/help/vision/ref/vision.pointtracker-class.html>.
- [119] Mohri, M., Rostamizadeh, A., and Talwalkar, A., *Foundations of machine learning*, MIT press, 2012.
- [120] Murphy, K. P., *Overview of Machine Learning*, Vancouver: University of British Columbia., 2007.
- [121] Kotsiantis, S., and Pintelas, P., "Recent Advances in Clustering: A Brief Survey," *WSEAS Transactions on Information Science and Applications*, vol. 1, no. 1, pp. 73-81, 2004.
- [122] Zhang, S., Zhang, C., and Yang, Q., "Data Preparation for Data Mining," *Applied artificial intelligence*, vol. 17, no. 5-6, pp. 375-381, 2003.
- [123] Yu, L., and Liu, H., "Efficient Feature Selection Via Analysis of Relevance and Redundancy," *Journal of machine learning research*, vol. 5, Oct, pp. 1205-1224, 2004.
- [124] Kersaints, S. B., Zaharakis, I., and Pintelas, P., "Supervised Machine Learning: A Review of Classification Techniques," *Emerging artificial intelligence applications in computer engineering*, vol. 160, pp. 3-24, 2007.

-
- [125] Murthy, S. K., "Automatic Construction of Decision Trees from Data: A Multi-Disciplinary Survey," *Data mining and knowledge discovery*, vol. 2, no. 4, pp. 345-389, 1998.
- [126] Quinlan, J. R., *C4. 5: Programs for Machine Learning*, Elsevier, 1994.
- [127] Zhang, G. P., "Neural Networks for Classification: A Survey," *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, vol. 30, no. 4, pp. 451-462, 2000.
- [128] Jensen, F. V., *An introduction to Bayesian networks*, UCL press, 1996.
- [129] Bouckaert, R. R., "Naive Bayes Classifiers That Perform Well with Continuous Variables," in *Australasian Joint Conference on Artificial Intelligence*, 2004, pp. 1089-1094.
- [130] Michalski, R. S., Carbonell, J. G., and Mitchell, T. M., *Machine learning: An Artificial Intelligence Approach*, Springer Science & Business Media, 2013.
- [131] Bijalwan, V., Kumar, V., Kumari, P., and Pascual, J., "KNN Based Machine Learning Approach for Text and Document Mining," *International Journal of Database Theory and Application*, vol. 7, no. 1, pp. 61-70, 2014.
- [132] Galar, M., Fernández, A., Barrenechea, E., Bustince, H., and Herrera, F., "An Overview of Ensemble Methods for Binary Classifiers in Multi-Class Problems: Experimental Study on One-vs-One and One-vs-All Schemes," *Pattern Recognition*, vol. 44, no. 8, pp. 1761-1776, 2011.
- [133] Fei, B., and Liu, J., "Binary Tree Of SVM: A New Fast Multiclass Training and Classification Algorithm," *IEEE transactions on neural networks*, vol. 17, no. 3, pp. 696-704, 2006.
- [134] Cristianini, N., and Scholkopf, B., "Support Vector Machines and Kernel Methods: The New Generation of Learning Machines," *AI Magazine*, vol. 23, no. 3, p. 31, 2002.

-
- [135] MathWorks, "Machine Learning,". Accessed: Sep. 18, 2020. [Online]. Available: <https://www.mathworks.com/solutions/machine-learning.html>.
- [136] Bengio, Y., Courville, A., and Vincent, P., "Representation Learning: A Review and New Perspectives," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 35, no. 8, pp. 1798-1828, 2013.
- [137] Schmidhuber, J., "Deep Learning in Neural Networks: An overview," *Neural Networks*, vol. 61, pp. 85-117, 2015.
- [138] Deng, L., and Yu, D., "Deep Learning: Methods and Applications," *Foundations and Trends in Signal Processing*, vol. 7, no. 3-4, pp. 197-387, 2014.
- [139] Pouyanfar, S., et al., "A Survey on Deep Learning: Algorithms, Techniques, and Applications," *ACM Comput. Surv.*, vol. 51, no. 5, 2018.
- [140] Brownlee, J., *Deep Learning With Python Develop Deep Learning Models On Theano And TensorFlow Using Keras i Deep Learning With Python*, 2016.
- [141] Abeykoon, V., et al., "Real Time Identification of Electrical Devices Through Power Consumption Pattern Detection," *Pervasive Computing*, vol. 10, no. 1, pp. 40-48, 2016.
- [142] Guo, J., Fan, Y., Pang, L., Yang, L., Ai, Q., Zamani, H., Wu, C., Croft, W.B., and Cheng, X., "A Deep Look into Neural Ranking Models for Information Retrieval," *Information Processing & Management*, vol. 57, no. 6, p. 102067, 2020.
- [143] He, K., Zhang, X., Ren, S., & Sun, J., "Spatial Pyramid Pooling in Deep Convolutional Networks for Visual Recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 37, no. 9, pp. 1904-1916, 2015.
- [144] He, K., Zhang, X., Ren, S., & Sun, J., "Deep Residual Learning for Image Recognition," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 770-778.

-
- [145] Sirojan, T., Phung, T., & Ambikairajah, E., "Intelligent Edge Analytics for Load Identification in Smart Meters," in *2017 IEEE Innovative Smart Grid Technologies-Asia (ISGT-Asia)*, 2017, pp. 1-5.
- [146] Liu, X., et al., "Smart Meter Data Analytics: Systems, Algorithms, and Benchmarking," *ACM Transactions on Database Systems (TODS)*, vol. 42, no. 1, 2016.
- [147] Jo, H., and Yoon, Y. I., "Intelligent Smart Home Energy Efficiency Model using Artificial TensorFlow Engine," *Human-centric Computing and Information Sciences*, vol. 8, no. 1, pp. 1-18, 2018.
- [148] Bodla, N., Singh, B., Chellappa, R., and Davis, L. S., "Soft-NMS--Improving Object Detection with One Line of Code," in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 5561-5569.
- [149] Cuñado, J. R., Linsangan, N. B., "A Supervised Learning Approach to Appliance Classification Based on Power Consumption Traces Analysis," in *IOP Conference Series: Materials Science and Engineering*, vol. 517, no. 1, p. 012011, IOP Publishing, April 2019.
- [150] Sze, V., Chen, Y. H., Yang, T. J., Emer, J. S., "Efficient Processing of Deep Neural Networks: A Tutorial and Survey," *Proceedings of the IEEE*, vol. 105, no. 12, pp. 2295-2329, 2017.
- [151] Li, J., Chang, H., Yang, J., "Sparse Deep Stacking Network for Image Classification," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 29, no. 1, 2015.
- [152] Mosavi, A., Salimi, M., Faizollahzadeh Ardabili, S., Rabczuk, T., Shamshirband, S., Varkonyi-Koczy, A. R., "State of The Art of Machine Learning Models in Energy Systems, A Systematic Review," *Energies*, vol. 12, no. 7, p. 1301, 2019.
- [153] Dahl, G. E., Yu, D., Deng, L., Acero, A., "Context-Dependent Pre-Trained Deep Neural Networks for Large-Vocabulary Speech Recognition," *IEEE Trans.*

- [154] Bell, S., Zitnick, C. L., Bala, K., Girshick, R., "Inside-Outside Net: Detecting Objects in Context with Skip Pooling and Recurrent Neural Networks," in *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, 2016, pp. 2874-2883.
- [155] Particke, F., Kolbenschlag, R., Hiller, M., Patiño-Studencki, L., Thielecke, J., "Deep Learning for Real-Time Capable Object Detection and Localization on Mobile Platforms," in *IOP Conference Series: Materials Science and Engineering*, vol. 261, no. 1, p. 012005, Oct. 2017.
- [156] Dai, J., Li, Y., He, K., Sun, J., "R-Fcn: Object Detection Via Region-Based Fully Convolutional Networks," in *Advances in neural information processing systems*, 2016, vol. 29.
- [157] Karpathy, A., Leung, T., "Karpathy_Large-scale_Video_Classification_2014_CVPR_paper," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2014, pp. 23-28.
- [158] OLAVI, S., "Object Detection from Images Using Convolutional Neural Networks," Master's thesis, Aalto University, 2017.
- [159] Chinaea, A., "Understanding the Principles of Recursive Neural Networks: A Generative Approach to Tackle Model Complexity," in *Artificial Neural Networks–ICANN 2009: 19th International Conference*, Limassol, Cyprus, September 14-17, 2009, Proceedings, Part I, vol. 19, pp. 952-963, Springer Berlin Heidelberg, 2009.
- [160] Bianchini, M., Gori, M., Mazzoni, P., Sarti, L., Scarselli, F., "Face Localization with Recursive Neural Networks," in *WIRN*, 2003, vol. 2859, pp. 99-105.
- [161] Bianchini, M., Maggini, M., Sarti, L., Scarselli, F., "Recursive Neural Networks Learn to Localize Faces," *Pattern recognition letters*, vol. 26, no. 12, pp. 1885-1895, 2005.
- [162] Bianchini, M., Mazzoni, P., Sarti, L., Scarselli, F., "Face Spotting in Colour Images using Recursive Neural Networks," in *Proceedings of the 1st ANNPR*

Workshop, 2003.

[163] Bengio, Y., Laufer, E., Alain, G., Yosinski, J., "Deep Generative Stochastic Networks Trainable by Backprop," in *International Conference on Machine Learning*, 2014, pp. 226-234.

[164] Voulodimos, A., Doulamis, N., Doulamis, A., Protopapadakis, E., "Deep Learning for Computer Vision: A Brief Review," *Comput. Intell. Neurosci.*, vol. 2018, Article ID 7068349, 2018.

[165] Wang, W., Liu, P., Ying, R., Wang, J., Qian, J., Jia, J., Gao, J., "A High-Computational Efficiency Human Detection and Flow Estimation Method Based on Tof Measurements," *Sensors*, vol. 19, no. 3, p. 72.

[166] Shelhamer, E., Long, J., Darrell, T., "Fully Convolutional Networks for Semantic Segmentation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 4, pp. 640–651, 2017.

[167] Donahue, J., et. al., "Long-Term Recurrent Convolutional Networks for Visual Recognition and Description," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 4, pp. 677–691, 2017.

[168] Joseph, A., Geetha, P., "Facial Emotion Detection using Modified Eyemap–Mouthmap Algorithm on an Enhanced Image and Classification With TensorFlow," *Vis. Comput.*, vol. 36, no. 3, pp. 529–539, 2020.

[169] Chai, J., Zeng, H., Li, A., Ngai, E. W., "Deep Learning in Computer Vision: A Critical Review of Emerging Techniques and Application Scenarios," *Machine Learning with Applications*, vol. 6, p. 100134

[170] Li, Y., "Research and Application of Deep Learning in Image Recognition," in *2022 IEEE 2nd International Conference on Power, Electronics and Computer Applications (ICPECA)*, 2022, pp. 994-999.

[171] Belhassen, H., Fresse, V. and Bourennane, E.B., "Comparative Study of Face and Person Detection Algorithms: Case Study of Tramway in Lyon." In *2019*

International Conference on Advanced Systems and Emergent Technologies 2019, March. (IC_ASET) (pp. 154-159). IEEE.

[172] Choi, J., Chun, D., Kim, H., Lee, H. J., "Gaussian YOLOv3: An Accurate and Fast Object Detector using Localization Uncertainty for Autonomous Driving," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2019, pp. 502-511.

[173] Jiang, P., Ergu, D., Liu, F., Cai, Y. and Ma, B., 2022. A Review of YOLO Algorithm Developments. *Procedia Computer Science*, 199, pp.1066-1073.

[174] Kumar, C. and Punitha, R., "YOLOv3 and YOLOv4: Multiple Object Detection for Surveillance Applications." In *2020 Third international conference on smart systems and inventive technology (ICSSIT)*. 2020, August. (pp. 1316-1321). IEEE.

[175] Srithar, S., Priyadharsini, M., Sharmila, F. M., & Rajan, R., "YOLOv3 Supervised Machine Learning Framework for Real-Time Object Detection and Localization," *J. Phys.: Conf. Ser.*, vol. 1916, no. 1, p. 012032, 2021.

[176] Yang, Q., Zhang, Y., Dai, W., & Pan, S. J., *Transfer Learning*, Cambridge University Press, 2020.

[177] Pujara, A., "Concept of AlexNet: Convolutional Neural Network," Accessed: Dec. 29, 2022. [Online]. Available: <https://medium.com/analytics-vidhya/concept-of-alexnet-convolutional-neural-network-6e73b4f9ee30>. Published on November 3, 2020. Last

[178] Krizhevsky, A., Sutskever, I., & Hinton, G. E., "ImageNet Classification with Deep Convolutional Neural Networks," *Commun. ACM*, vol. 60, no. 6, pp. 84-90, 2017.

[179] Simonyan, K., & Zisserman, A., "Very Deep Convolutional Networks for Large-Scale Image Recognition," *arXiv preprint arXiv:1409.1556*, 2014.

-
- [180] Wei, J., "VGG Neural Networks: The Next Step After AlexNet," Accessed: Dec. 29, 2022. [Online]. Available: <https://towardsdatascience.com/vgg-neural-networks-the-next-step-after-alexnet-3f91fa9ffe2c>. Published on July 4, 2019.
- [181] Serengil, S. I., "Deep Face Recognition with Keras," Accessed: Dec. 30, 2022., [Online]. Available: <https://sefiks.com/2018/08/06/deep-face-recognition-with-keras/>. Published on August 6, 2018.
- [182] He, K., Zhang, X., Ren, S., & Sun, J., "Deep Residual Learning for Image Recognition," in *Proc. IEEE Conf. on Comp. Vis. and Pattern Recog.*, 2016, pp. 770-778.
- [183] Sarker, I. H., "Deep Learning: A Comprehensive Overview on Techniques, Taxonomy, Applications and Research Directions," *SN Computer Science*, vol. 2, no. 6, pp. 1-20, 2021.
- [184] Nguyen, G., Dlugolinsky, S., Bobák, M., Tran, V., López García, Á., Heredia, I., Malík, P., & Hluchý, L., "Machine Learning and Deep Learning Frameworks and Libraries for Large-Scale Data Mining: A Survey," *Artificial Intelligence Review*, vol. 52, no. 1, pp. 77-124, 2019.
- [185] Goodfellow, I., Bengio, Y., & Courville, A., *Deep Learning*, MIT press, 2016.
- [186] NVIDIA, "CUDA Zone—NVIDIA Development," Accessed: Dec. 20, 2022. [Online]. Available: <https://developer.nvidia.com/cuda-zone>.
- [187] NVIDIA, "NVIDIA CUDA Toolkit," Accessed: Dec. 20, 2022. [Online]. Available: <https://developer.nvidia.com/cuda-toolkit>.
- [188] NVIDIA, "NVIDIA cuDNN—GPU Accelerated Deep Learning," Accessed: Dec. 20, 2022. [Online]. Available: <https://developer.nvidia.com/cudnn>.
- [189] Rampasek, L., & Goldenberg, A., "TensorFlow: Biology's Gateway to Deep Learning?" *Cell systems*, vol. 2, no. 1, pp. 12-14, 2016.
- [190] Zeng, Z., Gong, Q., & Zhang, J., "CNN Model Design of Gesture Recognition Based on TensorFlow Framework," in *2019 IEEE 3rd Information Technology*,

Networking, Electronic and Automation Control Conference (ITNEC), pp. 1062-1067, March 2019.

[191] Mishra, P., *PyTorch Recipes: A Problem-Solution Approach*, Apress, 2019.

[192] Paszke, A., Gross, S., Massa, F., Lerer, A., & Bradbury, J. P., "An Imperative Style, High-Performance Deep Learning Library," in *Advances in Neural Information Processing Systems*, 2019, vol. 32.

[193] Steppa, C., & Holch, T. L., "HexagDLy—Processing Hexagonally Sampled Data with CNNs in PyTorch," *SoftwareX*, vol. 9, pp. 193-198, 2019.

[194] Stooke, A., & Abbeel, P., "Rlpyt: A Research Code Base for Deep Reinforcement Learning in PyTorch," *arXiv preprint arXiv:1909.01500*, 2019.

[195] Vasilev, I., Slater, D., Spacagna, G., Roelants, P. and Zocca, V., *Python Deep Learning: Exploring deep learning techniques and neural network architectures with Pytorch, Keras, and TensorFlow*. Packt Publishing Ltd. 2019.

[196] Gulli, P. S., *Deep learning with Keras: implement neural networks with Keras on Theano and Tensor-Flow*, Birmingham Mumbai: Packt Publishing, 2017.

[197] Zebin, T., *Training Deep Neural Networks in Python Keras Framework*, 2018.

[198] Jia, Y., Shelhamer, E., & Donahue, J., "Caffe: Convolutional Architecture for Fast Feature Embedding," *arXiv e-prints*, 2014.

[199] Kishore, A., Jindal, S., & Singh, S., *Designing Deep Learning Neural Networks using Caffe*, 2015.

[200] Girshick, R., "Fast R-CNN," in *Proceedings of the IEEE international conference on computer vision*, 2015, pp. 1440-1448.

[201] Shatnawi, A., Al-Bdour, G., Al-Qurran, R., & Al-Ayyoub, M., "A Comparative Study of Open-Source Deep Learning Frameworks," in *2018 9th international conference on information and communication systems*

-
- [202] Gannon, D., *TensorFlow Meets Microsoft's CNTK*, 2016.
- [203] Yoon, Y. I., & Chun, J. A., "Tracking System for Mobile User Based on CCTV," in *The International Conference on Information Networking 2014 (ICOIN2014)*, 2014, pp. 374-378.
- [204] Stankovic, S., Gajic, D., & Stankovic, R. S., "GPU Architecture and the Programming Environment," in *GPU Computing with Applications in Digital Logic*, 2012.
- [205] Jaffe, R. S., Li, H., Kienzle, M. M., & Sheng, M. C., "Input/Output System for A Massively Parallel, Single Instruction, Multiple Data (SIMD) Computer Providing for the Simultaneous Transfer of Data Between a Host Computer Input/Output System and All SIMD Memory Devices," U.S. Patent 5,410,727, Apr. 25, 1995.
- [206] Geer, D., "Chip Makers Turn to Multicore Processors," *Computer*, vol. 38, no. 5, pp. 11-13, 2005.
- [207] Garland, M., et al., "Parallel Computing Experiences with CUDA," *IEEE micro*, vol. 28, no. 4, 2008.
- [208] Krishnaprasad, S., "Uses and Abuses of Amdahl's Law," *Journal of Computing Sciences in Colleges*, vol. 17, no. 2, pp. 288-293, 2001.
- [209] Tarditi, D., Puri, S., & Oglesby, J., "Accelerator: Using Data Parallelism to Program GPUs for General-Purpose Uses," *ACM SIGPLAN Notices*, vol. 41, no. 11, pp. 325-335, 2006.
- [210] Chatterjee, D., DeOrio, A., & Bertacco, V., "Event-Driven Gate-Level Simulation with GP-GPUs," in *Proceedings of the 46th Annual Design Automation Conference*, pp. 557-562, ACM, 2009.
- [211] Van der Steen, A. J., and Dongarra, J. J., *Overview of Recent Supercomputers*, NCF, 2008.
- [212] Nickolls, J., and Dally, W. J., "The GPU Computing Era," *IEEE micro*, vol. 30, no. 2, 2010.

-
- [213] Sanders, J., and Kandrot, E., *CUDA by Example: An Introduction to General-Purpose GPU Programming*. Addison-Wesley Professional, 2010.
- [214] Catanzaro, B., Sundaram, N., and Keutzer, K., "Fast Support Vector Machine Training and Classification on Graphics Processors," in *Proceedings of the 25th international conference on Machine learning*, pp. 104-111, ACM, 2008.
- [215] Fang, J., Varbanescu, A. L., and Sips, H., "A Comprehensive Performance Comparison of CUDA and OpenCL," in *Parallel Processing (ICPP), 2011 International Conference on*, pp. 216-225, IEEE, 2011.
- [216] Stankovic, S., and Astola, J., "An Overview of Miscellaneous Applications of GPU Computing," in *GPU Computing with Applications in Digital Logic*, p. 191, 2012.
- [217] Cheng, K. T., and Wang, Y. C., "Using Mobile GPU for General-Purpose Computing—A Case Study of Face Recognition on Smartphones," in *VLSI Design, Automation and Test (VLSI-DAT), 2011 International Symposium on*, pp. 1-4, IEEE, 2011.
- [218] Graf, H. P., Cosatto, E., Bottou, L., Dourdanovic, I., and Vapnik, V., "Parallel Support Vector Machines: The Cascade SVM," in *Advances in neural information processing systems*, pp. 521-528, 2005.
- [219] Mathworks, "GPU Coder," Accessed: Nov. 18, 2018. [Online]. Available: <https://www.mathworks.com/products/gpu-coder.html>.
- [220] Khadse, R., Gawai, N., and Faruk, B. M., "Overview and Comparative Study of Different Microcontrollers," *International Journal for Research in Applied Science & Engineering Technology (IJRASET)*, vol. 2, no. 12, pp. 311-315, 2014.
- [221] Güven, Y., Coşgun, E., Kocaoğlu, S., Gezici, H., and Yılmazlar, E., *Understanding the Concept of Microcontroller-Based Systems to Choose the Best Hardware for Applications*, 2017.

-
- [222] Tolmacheva, A. E., Ogurtsov, D. A., and Dorrer, M. G., "Justification for Choosing A Single-Board Hardware Computing Platform for A Neural Network Performing Image Processing," in *IOP Conference Series: Materials Science and Engineering*, vol. 734, no. 1, p. 012130, IOP Publishing, 2020.
- [223] Kondaveeti, H. K., Bandi, D., Mathe, S. E., Vappangi, S., and Subramanian, M., "A Review of Image Processing Applications based on Raspberry-Pi," in *2022 8th International Conference on Advanced Computing and Communication Systems (ICACCS)*, vol. 1, pp. 22-28, IEEE, March 2022.
- [224] Mathe, S. E., Pamarthy, A. C., Kondaveeti, H. K., and Vappangi, S., "A Review on Raspberry Pi and Its Robotic Applications," in *2022 2nd International Conference on Artificial Intelligence and Signal Processing (AISP)*, pp. 1-6, IEEE, February 2022.
- [225] Mittal, S., "A Survey on Optimized Implementation of Deep Learning Models on The NVIDIA Jetson Platform," *Journal of Systems Architecture*, vol. 97, pp. 428-442, 2019.
- [226] "HiKey970 Parameters," [Online]. Available: <https://www.96boards.org/product/hikey970/>.
- [227] "Neural-Compute Stick Details," [Online]. Available: <https://www.intel.com/content/www/us/en/developer/articles/tool/neural-compute-stick.html>.
- [228] Baloga, V. I., Kholosha, V. I., Evdin, O. M., and Perepelyatnykova, L. V., "Twenty-Five Years After Chornobyl Accident: Safety for the Future. National Report of Ukraine (Russian Edition), Ministry of Ukraine of Emergencies and Affairs of Population Protection from The Consequences of Chornobyl Catastrophe (Ukraine)", 2011.
- [229] Scott, R. J., "Preventing Midair Collisions: An Analysis of the Traffic Collision Avoidance System (TCAS)," Doctoral dissertation, 1998.

-
- [230] Parasuraman, R., and Manzey, D. H., "Complacency and Bias in Human use of Automation: An Attentional Integration," *Human factors*, vol. 52, no. 3, pp. 381-410, 2010.
- [231] Fagnant, D. J., & Kockelman, K., "Preparing A Nation for Autonomous Vehicles: Opportunities, Barriers and Policy Recommendations," *Transportation Research Part A: Policy and Practice*, vol. 77, pp. 167-181, 2015.
- [232] Cui, J., & Sabaliauskaite, G., "US: A Unified Safety and Security Analysis Method for Autonomous Vehicles," in *Future of Information and Communication Conference*, pp. 600-611, Springer, Cham, 2018.
- [233] Boy, G., & Gruber, T. R., "Intelligent Assistant Systems: Support for Integrated Human-Machine Systems," *Knowledge Systems Laboratory*, Computer Science Department, Stanford University, pp. 7-9, 1990.
- [234] Brézillon, P., "From Expert Systems to Context-Based Intelligent Assistant Systems: A Testimony," *The Knowledge Engineering Review*, vol. 26, no. 1, pp. 19-24, 2011.
- [235] Chen, L. B., Su, J. P., Chen, M. C., Chang, W. J., Yang, C. H., & Sie, C. Y., "An Implementation of An Intelligent Assistance System for Visually Impaired/Blind People," in *2019 IEEE International Conference on Consumer Electronics (ICCE)*, 2019, pp. 1-2.
- [236] Zeng, J., Sun, Y., & Wang, F., "A Natural Hand Gesture System for Intelligent Human-Computer Interaction and Medical Assistance," in *2012 Third Global Congress on Intelligent Systems*, 2012, pp. 382-385.
- [237] Sloane, G., "Siri Vs. Alexa vs. Cortana vs. Google Assistant: It's Battle of The AI Systems," Accessed: Nov. 7, 2016. [Online]. Available: <https://adage.com/article/digital/ai-web/306639/>.
- [238] Praahas, A., Prithvi, R., Roshni, F., Shivaraj, S., & Sneha, P., "Machine Learning based Face Recognition System for Virtual Assistant," *International Research Journal of Engineering and Technology*, vol. 05, no. 05, May 2018.

-
- [239] Wu, B.F., Chen, C.J., Peng, H.Y., Chen, Y.H., and Lin, W.H., "Intelligent Driving Assistant Systems," U.S. Patent 8,244,410, 2012.
- [240] Huda, F. and Preston, D., "Software Qual J," *Software Quality Journal*, vol. 1, no. 1, pp. 9, 1992.
- [241] Dahua Technologies, "How to Choose the Right Resolution," Accessed: Oct. 29, 2019. [Online]. Available: <https://us.dahuasecurity.com/2019/06/26/how-to-choose-the-right-resolution/>.
- [242] Theia Technologies, "The Trade-off between Image Resolution and Field of View: The Influence of Lens Selection," Accessed: Nov. 11, 2019. [Online]. Available: <https://static1.squarespace.com/static/551ecf1ae4b0b101cf72bfa3/t/55413f87e4b0223ca67196d2/1430339463828/Image+Resolution+and+Field+of+View.pdf>.
- [243] "Understanding Focal Length and Field of View Imaging Resource Guide," Edmund Optics, Accessed: Nov. 9, 2019. [Online]. Available: <https://www.edmundoptics.co.uk/resources/industry-expertise/imaging-optics/imaging-resource-guide/>.
- [244] Pontin, M., "Sensor Resolution Versus Lens Resolution," Resolveoptics Articles, IEN Magazine, England, 2016, Accessed: Nov. 1, 2019. [Online]. Available: https://www.iem.eu/uploads/tx_etim/resolveopticsarticle2016a.pdf.
- [245] "What is Video Compression," Accessed: Oct. 21, 2019. [Online]. Available: <https://www.videosurveillance.com/tech/video-compression.asp>.
- [246] Voeller, J.G. (Ed.), *Wiley Handbook of Science and Technology for Homeland Security, 4 Volume Set*, John Wiley & Sons, 2010.
- [247] Simon, L., "CCTV Camera Settings: Here's How to get the Best Picture Possible," IFSEC International, [Online]. Available: <https://www.ifsecglobal.com/video-surveillance/cctv-camera-settings-heres-how-to-get-the-best-picture-possible/>.

-
- [248] Aziz, S., Ain, W.N., Majeed, R., Khan, M.A., Qayum, I., Ahmed, I., & Hosain, K., "Growth Centile Charts (Anthropometric Measurement) of Pakistani Pediatric Population," *JPMA-Journal of the Pakistan Medical Association*, vol. 62, no. 4, p. 367.
- [249] Young, J.W. "Head and Face Anthropology of US Civilians," Office of Aviation Medicine, Washington, DC, 1993.
- [250] Axis Communications, "Pixel Density: Meeting Operational Requirements in Network Video," May 2022, Accessed: Oct. 1, 2019. [Online]. Available: <https://www.axis.com/dam/public/53/47/fd/pixel-density-en-US-364241.pdf>.
- [251] Punjab Safe City Authority, "Introduction to PPIC3 Programme," Accessed: Dec. 21, 2019. [Online]. Available: <https://psca.gop.pk/assets/pdf/files/An-Introduction-to-PPIC3-edited.pdf>.
- [252] Dahlman, E., Parkvall, S., Skold, J. *4G: LTE/LTE-advanced for Mobile Broadband*, Academic press, 2013.
- [253] How fast is 4G? - 4G Speeds and UK Network Performance, 4G. co.uk, UK, Accessed: Nov. 11, 2019. [Online]. Retrieved from: <https://www.4g.co.uk/how-fast-is-4g/>.
- [254] DeCusatis, C. ed., *Handbook of Fiber Optic Data Communication: A Practical Guide to Optical Networking*. Academic Press., 2013.
- [255] Cisco USA, "Cisco Catalyst 2960-X and 2960-XR Series Switches, Datasheet Cisco Catalyst," Accessed: Feb. 15, 2019. [Online]. Available: https://www.cisco.com/c/en/us/products/collateral/switches/catalyst-2960-x-series-switches/datasheet_c78-728232.pdf.
- [256] Reolink, "IP Camera Bandwidth Calculation: Easy Formula & Quick Tips to Reduce Bandwidth Usage," Reolink, Apr. 28, 2019. Accessed: Nov. 14, 2019. [Online]. Available: <https://reolink.com/ip-camera-bandwidth-calculation/>.

-
- [257] H.264+ Bit Rate Chart, Technical Bulletin Hikvision, USA, Accessed: Oct. 2, 2018. [Online]. Available: https://us.hikvision.com/sites/default/files/tb/tb_bit_rate_chart_120115us_0.pdf.
- [258] "Overview of Lahore," Metropolitan Corporation Lahore, Pakistan, Accessed: Aug. 11, 2018. [Online]. Available: <http://lahore.gop.pk/OverViewLahore.php>.
- [259] "Census Report 2017," Pakistan Bureau of Statistics Government of Pakistan, 2017. Accessed: Aug. 11, 2018. [Online]. Available: http://www.pbs.gov.pk/sites/default/files/PAKISTAN%20TEHSIL%20WISE%20FOR%20WEB%20CENSUS_2017.pdf.
- [260] Richardson, I.E., *The H. 264 Advanced Video Compression Standard*, John Wiley & Sons, 2011.
- [261] Munir, K., "Development or Disparity: The Case of Lahore, Pakistan," Accessed: Dec. 1, 2019. [Online]. Available: <https://due-parsons.github.io/methods3-fall2018/projects/development-or-disparity-the-case-of-lahore-pakistan/>.
- [262] Bhutekar, S.J. and Manjaramkar, A.K., "Parallel Face Detection and Recognition on GPU," *International Journal of Computer Science and Information Technologies*, vol. 5, no. 2, pp. 2013-2018, 2014.
- [263] Oro, D., Fernández, C., Saeta, J.R., Martorell, X., and Hernando, J., "Real-Time GPU-based Face Detection in HD Video Sequences," in *2011 IEEE International Conference on Computer Vision Workshops (ICCV Workshops)*, pp. 530-537, IEEE, 2011.
- [264] V. Jain and D. Patel, "A GPU based Implementation of a Robust Face Detection System," *Procedia Computer Science*, vol. 87, pp. 156-163, 2016.
- [265] "GPU Database," TECHPOWERUP, W1zzard Germany Accessed: Dec. 7, 2019. [Online]. Available: <https://www.techpowerup.com/gpu-specs/>.
- [266] Mughal, T.A., Abbas, M., Abrar, A., and Abbas, A., "Investigating the Temporal Variation in the Frequency of Smog Precursors in the Ambient Air of Lahore."

-
- [267] Ahsan, T., Chaudhry, M.A., Huda, N.u., Mukhtar, N., Ahmad, R., and Ali, M., "Knowledge and Perception of Smog in Families of employees of CMH Lahore, Pakistan," *Rawal Medical Journal*, vol. 45, no. 1.
- [268] Crete, F., Dolmiere, T., Ladret, P., and Nicolas, M., "The Blur Effect: Perception and Estimation with A New No-Reference Perceptual Blur Metric," in *Human Vision and Electronic Imaging XII*, vol. 6492, p. 64920I, International Society for Optics and Photonics, Feb. 2007.
- [269] Zhang, W., Chen, W., Wang, Z., and Li, H., "An Improved Fog-Removing Method for the Traffic Monitoring Image," in *2014 12th International Conference on Signal Processing (ICSP)*, pp. 938-941, IEEE, Oct. 2014.
- [270] Rahman, Z.U., Jobson, D.J., and Woodell, G.A., "Multi-Scale Retinex for Color Image Enhancement," in *Proceedings of 3rd IEEE International Conference on Image Processing*, vol. 3, pp. 1003-1006, IEEE, Sep. 1996.
- [271] Lee, S., Yun, S., Nam, J.H., Won, C.S., and Jung, S.W., "A Review on Dark Channel Prior Based Image Dehazing Algorithms," *EURASIP Journal on Image and Video Processing*, vol. 2016, no. 1, pp. 1
- [272] Belhassen, H., Fresse, V., and Bourennane, E.B., "Comparative Study of Face and Person Detection Algorithms: Case Study of Tramway in Lyon," in *2019 International Conference on Advanced Systems and Emergent Technologies (IC_ASET)*, Mar. 2019, pp. 154-159.
- [273] Ullah, R., Hayat, H., Siddiqui, A.A., Siddiqui, U.A., Khan, J., Ullah, F., Hassan, S., Hasan, L., Albattah, W., Islam, M., & Karami, G. M. (2022). "A Real-Time Framework for Human Face Detection and Recognition in CCTV images.", *Mathematical Problems in Engineering*, 2022.
- [274] Halawa, L.J., Wibowo, A. and Ernawan, F., "Face Recognition using Faster R-CNN with Inception-V2 Architecture for CCTV Camera," in *2019 3rd International Conference on Informatics and Computational Sciences (ICICoS)*, October 2019, pp. 1-6, IEEE.

-
- [275] Enriquez, K., "Faster Face Detection using Convolutional Neural Networks & the Viola-Jones Algorithm," *no. May* 2018.
- [276] Lee, W., Park, J., Jeong, J., Jeon, E., Kim, H.E., Park, S., and You, H., "Analysis of the Facial Anthropometric Data of Korean Pilots for Oxygen Mask Design," in *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*, vol. 56, no. 1, Sep. 2012, pp. 1927-1931.
- [277] Shabab, A., "Marketing the Beard: The use of Propaganda in the Attempt to Talibanize Pakistan.", Doctoral dissertation, University of Southern California). 2012.
- [278] LL, D., Wang, L.M., and Zhuang, Z., "Measurement and Analysis of Human Head-Face Dimensions," *Chinese journal of industrial hygiene and occupational diseases*, vol. 26, no. 5, pp. 266-270, 2008.
- [279] Lin, Y.C., and Chen, C.P., "Characterization of Small-To-Medium Head-And-Face Dimensions for Developing Respirator Fit Test Panels and Evaluating Fit of Filtering Facepiece Respirators with Different Face Seal Design," *PLoS One*, vol. 12, no. 11, p. e0188638, 2017.
- [280] "Perfect Pixel Count Meeting your Operational Requirements," Axis Communications. Accessed: Oct. 1, 2022. [Online]. Available: https://www.axis.com/files/feature_articles/ar_perfect_pixel_count_55971_en_1402_lo.pdf.
- [281] Terino, E.O., "Alloplastic Facial Contouring by Zonal Principles of Skeletal Anatomy," *Clinics in plastic surgery*, vol. 19, no. 2, pp. 487-510, 1992.
- [282] Farkas, L.G., and Kolar, J.C., "Anthropometrics and Art in the Aesthetics of Women's Faces," *Clinics in plastic surgery*, vol. 14, no. 4, pp. 599-616, 1987.
- [283] "Census Report 2017," Pakistan Bureau of Statistics, Government of Pakistan, 2017. Accessed: Dec. 24, 2019. [Online]. Available: http://www.pbs.gov.pk/sites/default/files/PAKISTAN%20TEHSIL%20WISE%20FOR%20WEB%20CENSUS_2017.pdf.

-
- [284] Haider, K.Z., Malik, K.R., Khalid, S., Nawaz, T. and Jabbar, S., "Deepgender: Real-Time Gender Classification using Deep Learning for Smartphones," *Journal of Real-Time Image Processing*, vol. 16, pp. 15-29, 2019.
- [285] Kumar, S., Singh, S., & Kumar, J. "Gender Classification using Machine Learning with Multi-Feature Method.", in *2019 IEEE 9th Annual Computing and Communication Workshop and Conference (CCWC)* (pp. 0648-0653). IEEE., 2019
- [286] Thangaraj, R., Pandiyan, P., Pavithra, T., Manavalasundaram, V.K., Sivaramakrishnan, R. and Kaliappan, V.K., "Deep Learning based Real-Time Face Detection and Gender Classification using OpenCV and Inception v3," in *2021 International Conference on Advancements in Electrical, Electronics, Communication, Computing and Automation (ICAECA)*, pp. 1-5, October 2021. IEEE.
- [287] Claes, P., Roosenboom, J., White, J.D., Swigut, T., Sero, D., Li, J., Lee, M.K., Zaidi, A., Mattern, B.C., Liebowitz, C., and Pearson, L., "Genome-Wide Mapping of Global-to-Local Genetic Effects on Human Facial Shape," *Nature genetics*, vol. 50, no. 3, pp. 414-423, 2018.
- [288] Claes, P., Liberton, D.K., Daniels, K., Rosana, K.M., Quillen, E.E., Pearson, L.N., McEvoy, B., Bauchet, M., Zaidi, A.A., Yao, W., and Tang, H., "Modeling 3D Facial Shape from DNA," *PLoS Genet*, vol. 10, no. 3, p. e1004224, 2014.
- [289] Qamar, R., Ayub, Q., Mohyuddin, A., Helgason, A., Mazhar, K., Mansoor, A., Zerjal, T., Tyler-Smith, C., and Mehdi, S.Q., "Y-Chromosomal DNA Variation in Pakistan," *The American Journal of Human Genetics*, vol. 70, no. 5, pp. 1107-1124, 2002.
- [290] Central Intelligence Agency, "The World Factbook 2009," Government Printing Office, 2003.
- [291] Gazdar, H., "Class, Caste, or Race: Veils over Social Oppression in Pakistan," *Economic and Political Weekly*, pp. 86-88, 2007.
- [292] Ahmed, F., "Africa on the Coast of Pakistan," *New Directions*, vol. 16, no. 4, 1989, p. 5.

-
- [293] Gao, W., Cao, B., Shan, S., Chen, X., Zhou, D., et. al., "The CAS-PEAL Large-Scale Chinese Face Database and Baseline Evaluations," *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 38, no. 1, pp. 149–161, 2008.
- [294] Khan, K., Khan, R.U., Ali, J., Uddin, I., Khan, S., and Roh, B.H., Race Classification using Deep Learning," *Computer Mater. Contin.*, vol. 68, no. 3, pp. 3483-3498, 2021.
- [295] Salah, S.H., Du, H., and Al-Jawad, N., "Fusing Local Binary Patterns with Wavelet Features for Ethnicity Identification," in *Proceedings of world academy of science, engineering and technology (Vol. 79)*, World Academy of Science, Engineering and Technology (WASET), Jul. 2013, p. 471.
- [296] Roomi, S., Virasundarii, S., Selvamegala, S., Jeevanandham, S., Hariharasudhan, D., et. al., "Race Classification Based on Facial Features," in *Proc. Third National Conf. on Computer Vision, Pattern Recognition, Image Processing, and Graphics*, Massachusetts Ave., NW Washington, DC United States, 2011, pp. 54–57.
- [297] Antonio, G., Percannella, G., Vento, M., and Vigilante, V., "Benchmarking Deep Network Architectures for Ethnicity Recognition using a New Large Face Dataset," *Mach. Vis. Appl.*, vol. 31, no. 7, pp. 1–13, 2020.
- [298] Yiting, X., Luu, K., and Savvides, M., "A Robust Approach to Facial Ethnicity Classification on Large Scale Face Databases," in *Proc. IEEE Fifth Int. Conf. on Biometrics: Theory, Applications and Systems*, Arlington, VA, USA, 2012, pp. 143–149.
- [299] Chen, H., Gallagher, A., and Girod, B., "Face Modeling with First Name Attributes," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 36, no. 9, pp. 1860–1873, 2014.
- [300] Geng, X., Zhou, Z.H., and Miles, K., "Automatic Age Estimation Based on Facial Aging Patterns," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 29, no. 12, pp. 2234–2240, 2007.

-
- [301] Sarfaraz, M., Gupta, S., Wajid, A., Gupta, S., and Ahmed, M., "Prediction of Human Ethnicity from Facial Images using Neural Networks," in *Proc. Data Engineering and Intelligent Computing*, Singapore, 2018, pp. 217–226.
- [302] Qiong, C., Shen, L., Xie, W., Parkhi, O.M., and Zisserman, A., "VGGFace2: A Dataset for Recognizing Faces Across Pose and Age," in *Proc. 13th IEEE Int. Conf. on Automatic Face & Gesture Recognition*, Xi'an, China, 2021, pp. 67–74.
- [303] Ubaid, M.T., Khalil, M., Khan, M.U.G., Saba, T. and Rehman, A., "Beard and Hair Detection, Segmentation and Changing Color Using Mask R-CNN," *Proceedings of International Conference on Information Technology and Applications: ICITA 2021*, pp. 63-73, April 2022. Singapore: Springer Nature Singapore.
- [304] Mushtaq, D.G., *Scientific & Islamic Reasons for Growing a Beard*.
- [305] Khan, N., *Muslim becoming: Aspiration and Skepticism in Pakistan*, Duke University Press, 2012.
- [306] Hassner, T., Harel, S., Paz, E., and Enbar, R., "Effective Face Frontalization in Unconstrained Images," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015, pp. 4295-4304.
- [307] "Raspberry-Pi-4-Product-Brief," RaspberryPi.org, US, Accessed: Nov. 11, 2019. [Online]. Retrieved from: <https://static.raspberrypi.org/files/product-briefs/Raspberry-Pi-4-Product-Brief.pdf>.
- [308] "OPTEX Passive Infrared CX-702 Datasheet," Accessed: Sep. 20, 2019. [Online]. Retrieved from: <https://www.optex.co.jp/e/support/download-intrusion-detection/passivesensor/cx-702.html>.
- [309] Grandini, M., Bagli, E., and Visani, G., "Metrics for Multi-Class Classification: An Overview," *arXiv preprint arXiv:2008.05756*, 2020.
- [310] Gibaja, E., and Ventura, S., "A Tutorial on Multilabel Learning," *ACM Computer. Surveys*, vol. 47, no. 3, pp. 1-38, 2015.

-
- [311] Hoo, Z.H., Candlish, J., and Teare, D., "What is an ROC Curve?" *Emerg. Med. J.*, vol. 34, no. 6, pp. 357-359, 2017.
- [312] Powers, D.M., "Evaluation: from Precision, Recall and F-Measure to ROC, Informedness, Markedness and Correlation," *arXiv preprint arXiv:2010.16061*, 2020.
- [313] Cook, J., and Ramadas, V., "When to Consult Precision-Recall Curves," *The Stata J.*, vol. 20, no. 1, pp. 131-148, 2020.
- [314] Henderson, P., and Ferrari, V., "End-to-End Training of Object Class Detectors for Mean Average Precision," in *Proc. Computer. Vision. –ACCV 2016: 13th Asian Conf. on Computer. Vis.*, Taipei, Taiwan, 2016, pp. 198-213.
- [315] Kalpathy-Cramer, J., Patel, J.B., Bridge, C., and Chang, K., "Basic Artificial Intelligence Techniques: Evaluation of Artificial Intelligence Performance," *Radiologic Clinics*, vol. 59, no. 6, pp. 941-954, 2021.
- [316] Charu, C.A., *Neural Networks, and Deep Learning: A Textbook*.
- [317] Zeiler, M.D., and Fergus, R., "Visualizing and Understanding Convolutional Networks," in *Proc. Comput. Vis.–ECCV 2014: 13th European Conf.*, Zurich, Switzerland, 2014, pp. 818-83
- [318] Sara, U., Akter, M., and Uddin, M.S., "Image Quality Assessment through FSIM, SSIM, MSE and PSNR—A Comparative Study," *J. Computer. Communication.*, vol. 7, no. 3, pp. 8-18, 2019.
- [319] Feng, H., Mu, G., Zhong, S., Zhang, P., and Yuan, T., "Benchmark Analysis of Yolo Performance on Edge Intelligence Devices," *Cryptography*, vol. 6, no. 2, p. 16, 2022.

Annexure A

علاقہ: _____ Area: _____

QUESTIONNAIRE

استفسار نامہ

1	Age عمر	Less than 20 کم از کم 20	20-35	36-45	46-55	More than 56 56 سے زیادہ
2	Gender جنس	Male مرد	Female عورت			
3	For how long you are living/working at this location? آپ کب سے اس مقام پر رہے ہیں / کام کر رہے ہیں؟	Less than 3 years کم از کم 3 سال	4-6 years 4-6 سال	7-9 years 7-9 سال	10-12 years 10-12 سال	More than 12 years 12 سال سے زیادہ
4	How many hours a day do you stay at this location? آپ دن میں کتنے گھنٹے اس مقام پر قیام کرتے ہیں؟	Less than 1 hour 1 گھنٹے سے بھی کم	1-2 hours 1-2 گھنٹے	2-3 hours 2-3 گھنٹے	3-4 hours 3-4 گھنٹے	More than 4 hours 4 گھنٹے سے زیادہ
5	How do you travel? آپ کیسے سفر کرتے ہیں؟	Bus بس	Car گاڑی	Rickshaw/ Chingchi رکشہ / چنگچی	Motorcycle موٹر سائیکل	cycle/by foot سائیکل / بائی فٹ
6	How many signals you cross in a day? آپ ایک دن میں کتنے سگنل پار کرتے ہیں؟	None کوئی نہیں	1-2	3-5	6-9	9+
7	Do you feel safe after the installation of CCTV cameras? کیا آپ سی سی ٹی وی کیمرے کی تنصیب کے بعد محفوظ محسوس کرتے ہیں؟	Yes ہاں	No نہیں			
8	Does these cameras on main roads threat to your privacy? کیا اہم سڑکوں پر یہ کیمرے آپ کی رازداری کے لئے خطرہ ہیں؟	Yes ہاں	No نہیں			
9	Have you ever got any help from PSCA for any case? کیا آپ کو کبھی کسی بھی معاملے میں پی ایس سی اے سے کوئی مدد ملی ہے؟	Yes ہاں	No نہیں			
10	Have you ever heard about the PSCA helping someone in solving any case?	Yes ہاں	No نہیں			

	کیا آپ نے کبھی پی ایس سی اے کے بارے میں سنا ہے کہ وہ کسی بھی معاملے کو حل کرنے میں کسی کی مدد کرتا ہے؟		
11	Do you think that Humans are manually keeping an eye on you? کیا آپ کو لگتا ہے کہ انسان آپ پر نظر رکھے ہوئے ہیں؟	Yes ہاں	No نہیں
12	Do you want the government to add more cameras on the roads? کیا آپ چاہتے ہیں کہ حکومت سڑکوں پر مزید کیمرے شامل کرے؟	Yes ہاں	No نہیں
13	Do you think cameras in the streets is a privacy concern? کیا آپ کو لگتا ہے کہ گلیوں میں کیمرے رازداری کی تشویش ہے؟	Yes ہاں	No نہیں
14	Have you ever been a victim of street crime? کیا آپ کبھی اسٹریٹ کرائم کا شکار ہوئے ہیں؟	Yes ہاں	No نہیں
15	Have you ever experienced terror attack in radius of 1km? کیا آپ نے کبھی ایک کلومیٹر کے دائرے میں دہشت گردانہ حملے کا تجربہ کیا ہے؟	Yes ہاں	No نہیں
16	Do you think that there should be an automated system for surveillance minimizing the human involvement? کیا آپ کو لگتا ہے کہ انسانی شمولیت کو کم سے کم کرنے کے لئے نگرانی کے لئے ایک خودکار نظام ہونا چاہئے؟	Yes ہاں	No نہیں

Note: This questionnaire is generated only for PhD research for development of *Automated Person of Interest Tracking System*. Your personal data/identity will not be shared or published. Cumulative feedback will be shown in tabular form only. Please read all the questions first and leave this feedback blank if NOT willing to participate.

یہ سوالنامہ صرف دلچسپی سے باخبر رہنے کے نظام کے خود کار طریقے سے شخص کی ترقی کے لئے پی ایچ ڈی تحقیق کے لئے تیار کیا جاتا ہے۔ آپ کا ذاتی ڈیٹا / شناخت اشتراک یا شائع نہیں کیا جائے گا۔ مجموعی آراء صرف جدول کی شکل میں دکھائی جائیں گی۔ براہ کرم پہلے تمام سوالات پڑھیں اور اس رائے کو خالی چھوڑ دیں اگر حصہ لینے کے لئے تیار نہیں ہیں۔

Annexure B

Title:	H.264+ Bit Rate Chart	Version:	---	Date:	12/01/2015
Product:	Hikvision Cameras			Page:	1 of 1
Action Required:	None, Information Only				

About H.264+

H.264+ is Hikvision's optimized, fully compatible version of the industry-standard H.264 video compression protocol. H.264+ saves 50 percent bandwidth and storage space, on average, over H.264. For static scenes with few moving objects, H.264+ can provide up to 75 percent savings.

The following table lists recommended H.264+ bit rates for various resolutions.

H.264+ Recommended Bit Rate at General Resolutions (Approximate Value)															
Resolution and Rate		Frames	30 fps	25 fps	22 fps	20 fps	18 fps	15 fps	12.5 fps	10 fps	8 fps	6 fps	4 fps	2 fps	1 fps
12 MP (4000 × 3000)	Max Bitrate (Kbps)		27033	22528	21504	20480	18432	16384	14336	12288	10240	8192	6144	3584	2560
	Average Bitrate (Kbps)		10823	9020	8610	8200	7380	6560	6104	5232	4620	3696	2928	2165	1800
4K/8 MP (4096 × 2160)	Max Bitrate (Kbps)		19660	16384	15360	14336	13312	11264	10240	9216	8192	6656	5120	3072	2048
	Average Bitrate (Kbps)		7871	6560	6540	6104	5668	5082	4620	4158	3696	3172	2560	1856	1440
3840 × 2160	Max Bitrate (Kbps)		18432	15360	13312	12160	11264	10240	9216	8192	6656	5632	4608	2560	2048
	Average Bitrate (Kbps)		7380	6540	5668	5486	5082	4620	4158	3696	3172	2816	2304	1800	1440
6 MP (3072 × 2048)	Max Bitrate (Kbps)		13516	11264	10240	9216	8192	7680	6656	6144	5120	4352	3584	2048	1536
	Average Bitrate (Kbps)		5754	5082	4620	4158	3696	3660	3172	2928	2560	2176	2165	1440	1248
2560 × 1920	Max Bitrate (Kbps)		11059	9216	8192	8192	7680	6656	6656	5120	4096	3584	2560	1536	1024
	Average Bitrate (Kbps)		4989	4158	3696	3696	3660	3172	3172	2560	2048	2165	1800	1248	832
2048 × 2048	Max Bitrate (Kbps)		9216	7680	6144	5632	5120	4608	4352	4096	3584	2560	2048	1536	1024
	Average Bitrate (Kbps)		4158	3660	2928	2816	2560	2304	2176	2048	2165	1800	1440	1248	832
4 MP (2560 × 1440)	Max Bitrate (Kbps)		7987	6656	5632	5120	4608	4352	4096	3584	3072	2560	1920	1280	1024
	Average Bitrate (Kbps)		3806	3172	2816	2560	2304	2176	2048	2165	1856	1800	1560	1040	832
QXGA (2048 × 1536)	Max Bitrate (Kbps)		6144	5120	4608	4352	4096	3584	3200	3072	2560	2048	1536	1024	768
	Average Bitrate (Kbps)		2928	2560	2304	2176	2048	2165	1933	1856	1800	1440	1248	832	696
1536 × 1536	Max Bitrate (Kbps)		5529	4608	4096	3584	3200	3072	2560	2048	1536	1024	896	768	640
	Average Bitrate (Kbps)		2764	2304	2048	2165	1933	1856	1800	1440	1248	832	812	696	580
1080p (1920 × 1080)	Max Bitrate (Kbps)		4915	4096	3584	3072	2560	2240	2048	1536	1024	896	768	640	512
	Average Bitrate (Kbps)		2457	2048	2165	1856	1800	1575	1440	1248	832	812	696	580	464
UXGA (1600 × 1200)	Max Bitrate (Kbps)		4915	4096	3584	3072	2560	2240	2048	1536	1024	896	768	640	512
	Average Bitrate (Kbps)		2457	2048	2165	1856	1800	1575	1440	1248	832	812	696	580	464
2048 × 768	Max Bitrate (Kbps)		3686	3072	2560	2240	2048	1792	1536	1024	768	640	640	512	384
	Average Bitrate (Kbps)		2226	1856	1800	1575	1440	1456	1248	832	696	580	580	464	366
1280 × 1280	Max Bitrate (Kbps)		3686	3072	2560	2240	2048	1536	1024	896	768	640	512	448	384
	Average Bitrate (Kbps)		2226	1856	1800	1575	1440	1248	832	812	696	580	464	427	366
1280 × 960	Max Bitrate (Kbps)		2457	2048	1792	1536	1280	1024	896	768	640	512	448	384	256
	Average Bitrate (Kbps)		1727	1440	1456	1248	1040	832	812	696	580	464	427	366	244
720p (1280 × 720)	Max Bitrate (Kbps)		2457	2048	1792	1536	1280	1024	896	768	640	512	448	384	256
	Average Bitrate (Kbps)		1727	1440	1456	1248	1040	832	812	696	580	464	427	366	244
1024 × 1024	Max Bitrate (Kbps)		2150	1792	1536	1280	1024	896	768	640	512	448	384	320	256
	Average Bitrate (Kbps)		1511	1456	1248	1040	832	812	696	580	464	427	366	305	244
960 × 576	Max Bitrate (Kbps)		1680	1400	1232	1120	1008	840	672	560	453	367	297	241	216
	Average Bitrate (Kbps)		1365	1137	1001	910	913	761	609	507	431	349	283	229	205
720 × 720	Max Bitrate (Kbps)		1536	1280	1152	1024	896	768	640	512	448	320	256	192	128
	Average Bitrate (Kbps)		1248	1040	936	832	812	696	580	464	427	305	244	183	122
4CIF (704 × 576)	Max Bitrate (Kbps)		1228	1024	768	640	640	512	448	384	320	320	256	160	80
	Average Bitrate (Kbps)		997	832	696	580	580	464	427	366	305	305	244	152	76
VGA (640 × 480)	Max Bitrate (Kbps)		921	768	640	640	512	448	384	320	320	256	224	128	80
	Average Bitrate (Kbps)		834	696	580	580	464	427	366	305	305	244	213	122	76
528 × 384	Max Bitrate (Kbps)		614	512	450	409	368	307	245	204	165	134	108	88	79
	Average Bitrate (Kbps)		556	464	428	389	350	292	233	194	157	127	102	83	75
704 × 288	Max Bitrate (Kbps)		614	512	450	409	368	307	245	204	165	134	108	88	79
	Average Bitrate (Kbps)		556	464	428	389	350	292	233	194	157	127	102	83	75
352 × 288	Max Bitrate (Kbps)		307	256	225	204	184	153	122	102	82	67	54	44	39
	Average Bitrate (Kbps)		292	244	214	194	175	145	116	97	78	63	51	41	37
176 × 144	Max Bitrate (Kbps)		153	128	112	102	92	76	61	51	41	33	27	22	19
	Average Bitrate (Kbps)		145	122	106	97	87	72	58	48	39	31	25	20	18

NOTE 1: The max bitrate values are used for scenes with relatively more movement.

NOTE 2: For maximum reduced bandwidth and storage, variable bitrate type is recommended when H.264+ is enabled.

NOTE 3: The max bitrate values are suitable for all Hikvision H.264+ products using variable bitrate type. Recommended max bitrate value under constant bitrate type is the same as H.264.

Annexure C

VIDEO/ PHOTO CONSENT FORM

I, _____ grant permission to **Mr. Danish Arif**, PhD candidate at **University of Cape Town** for the use of my photograph(s) and video captured within the premises of **Department of Electrical Engineering, University of the Punjab** for his research project i.e. development of **Person Of Interest Recognition and Optical-based Tracking (POIROT) Intelligence Assistance System**. I understand that I may revoke this authorization at any time by notifying in writing. Images will be stored in a secure location and only authorized researchers will have access to them. They will be kept as long as they are relevant and after that time destroyed or archived. I will be communicated and consent will be required before publishing my image in any article.

Name _____

Address _____

City _____

Phone _____ **Email** _____

Signature _____ **Date** _____

Annexure D

Dear Participant,

Subject: REQUEST FOR USING YOUR IMAGE IN PUBLICATION

I am writing to request permission to use your image in my thesis publication that you have provided during the research, Person of Interest Rapid Identification and Tracking (POIRIT) Assistance System.

I would like to assure you that your image will not disclose your name or address and will not be used in any inappropriate way. I will not use this image for any other purposes without your permission. If you have any concerns or questions about the usage of the image, please let me know. I am happy to provide additional information or clarify any points that you may need.

Thank you for your consideration of this request.

Sincerely,

Danish Arif

Kindly fill the consent form below if your answer is in positive for above mentioned request.

I, _____ grant permission to **Mr. Danish Arif**, *University of Cape Town* for the publish my photograph(s) for his research project i.e., development of Person Of Interest Rapid Identification and Tracking (POIRIT) Assistance System.

Signatures _____