Stereo Facial Image Matching to Aid in Fetal Alcohol Syndrome Screening

BY

Rex Grobbelaar

Thesis submitted in partial fulfillment of the requirements for the degree of MSc(Med) in Biomedical Engineering

Department of Human Biology
Faculty of Health Sciences
University of Cape Town
Cape Town, South Africa
November 2004

Declaration

Stereo Facial Image Matching to Aid in Fetal Alcohol Syndrome Screening

I, Rex Grobbelaar, hereby declare that:

(i) This thesis with the above title is my own unaided work, and that apart from the normal guidance from my supervisor, I have received no assistance except as stated below.

(ii) Except where indicated to the contrary, neither the substance nor any part of this thesis with the above title has been submitted in the past, or is being, or is to be submitted for a degree in this university or any other university.

This thesis has been presented by myself and is being submitted for examination for the degree of Master of Science in Medicine in Biomedical Engineering at the University of Cape Town.

_____

Signature of Author

_____

Date

Acknowledgements

I thank my supervisor, Dr. Tania Douglas, for her guidance, leadership and advice throughout my MSc(Med) thesis.

My appreciation to those who contributed to the project:
Andre Bester for his help with the Smart Cameras.
Megan Watson for helping me with the camera calibration and its Matlab coding.
Bruce Spottiswoode for also helping me with Matlab.
Funding by the National Research Foundation (NRF) to make this thesis possible.

Special thanks to my fiancée, Rita Marais for her everlasting love and support.

Thanks to my mother, father and brothers (and their families) for their encouragement and prayers. Thanks for giving me the opportunity to study and believing in me.

Thanks to all my friends for their support.

I would like to give praise to my heavenly Father for giving me the opportunity, determination and talent to complete this project.

Abstract

The thesis project involved the development of software for the purpose of stereo image matching to obtain three-dimensional facial information. This was achieved with stereophotogrammetry, which makes use of two pictures obtained from digital cameras of the same object taken from different angles. With these two pictures, three-dimensional information of the object can be obtained through stereo matching. This information is used to obtain facial measurements that will aid in Fetal Alcohol Syndrome (FAS) screening.

A review of previously developed stereophotogrammetric techniques was performed to establish whether stereophotogrammetry is commonly used for 3D facial reconstruction and FAS diagnosis. It was found that although stereophotogrammetric systems have been developed, they have rarely been applied in the diagnosis of FAS and that the developed systems are expensive and complicated. However, an easily operated and cost-effective method to screen accurately for FAS on a large scale is required in South Africa.

FAS results from excessive maternal alcohol intake during pregnancy, leading to pre- and post-natal growth retardation in the fetus. A characteristic facial phenotype is associated with FAS, and measurements of these associated facial features are compared to population norms in order to identify subjects with FAS.

The MRC/UCT Medical Imaging Research Unit has developed a tool requiring minimum equipment and specialist participation for diagnosing FAS. This tool is based on stereophotogrammetry, using two pictures taken of a child's face from different angles using digital cameras. Corresponding points are manually identified on these two pictures and three-dimensional information of the face is obtained through camera calibration, from which reliable measurements can be made for diagnosing FAS.

The project presented here was developed with the aim of improving the above screening tool. Stereo matching software was developed to identify and accurately match corresponding areas containing the same facial features in both images for three-dimensional reconstruction. During stereo matching the user locates an image

feature of interest in one image by selecting certain points. The other image of the stereo pair is examined by the software to identify the unique image feature corresponding to the projection of the same surface points. The intensity of the pixels plays a big part in the matching process, and matching might not be accurate on images without distinct, contrasting features such as clear edges. Therefore image enhancement techniques were applied in order to improve matching accuracy, where matching is achieved without a person needing to identify points manually on both images. Texture projection and an infrared flash were also applied during the matching process and proved to improve the matching accuracy.

3D coordinates of the matched corresponding points were obtained through camera calibration, and these coordinates were used for feature measurement and for three-dimensional reconstruction of the matched features. The 3D reconstruction was achieved by connecting the 3D points through Delaunay triangulation to create a three-dimensional mesh that covers the relevant facial area. This mesh was interpolated to produce a reconstructed dense three-dimensional surface representing the facial area.

Although good matching accuracy results were obtained, there is still room for improvement in the developed algorithm. Three-dimensional surface reconstruction is successfully achieved, but is dependent on the accuracy of the matching – false matches will lead to an incorrect 3D surface. Therefore a few recommendations are given to help improve the matching accuracy. These include the use of certain matching constraints in the matching algorithm. The developed software enhances the existing stereophotogrammetric FAS screening tool by reducing user interaction and has a sufficient runtime, thus reducing the time and the cost of screening.

Table of Contents

List of Figures

List of Tables

Chapter 1

Introduction

This thesis explores the use of stereophotogrammetry as a medical imaging tool. It further describes the development of stereo matching and three-dimensional reconstruction software to aid in Fetal Alcohol Syndrome screening.

Three-dimensional (3D) imaging and measurement systems can be used for inspection of the human body and its components. These systems provide 3D data as discrete points, surfaces or volumes, and are used in a variety of fields. Once acquired, 3D image data can be visualized, measured and compared to perform the required tasks. The evaluation of 3D imaging system performance depends mainly on the specific application and body region. This thesis is concerned with taking measurements from a human face.

The principles of stereophotogrammetry are applied to diverse problems in the field of medicine. This branch of stereophotogrammetry is known as biostereometrics. Problems such as measuring body surfaces, body motion, contours, change in body shape, posture and movement of teeth are only a few that can be solved by this method (Mikhail *et al.*, 2001).

Observation of a scene or object from two different points allows the determination of depth or distance of the object. This can be achieved with a setup using two imaging sensors – a stereo system. Stereo systems are used by many biological visual systems to perform depth perception (Jähne, 1993), and one of the major areas in computer vision is the recovery of 3D shape information (depth) using stereo vision analysis. Passive stereo vision approaches are widely known. They attempt to imitate the depth extraction ability of the human visual system with the use of two cameras and a computer. In this case, obtaining 3D information involves the identification of the corresponding 2D points between the left and right images that are projections of the same physical point in the 3D scene. This is called the stereo matching problem. The stereo matching problem remains one of the most difficult

problems in computer vision and has stimulated a great deal of literature (Dipanda *et al.*, 2003).

1.1) Motivation for the Project

Fetal Alcohol Syndrome (FAS) is one of the most common preventable forms of mental retardation. FAS prevalence of 40.5 to 46.4 per 1000 children aged 5 to 9 years has been reported in one disadvantaged community in the Western Cape, while the rate for the developed world has been estimated to range from 0.33 per 1000 to 2.2 per 1000 (May *et al.*, 2000). FAS is the result of excessive alcohol intake during pregnancy, which affects the fetus. It leads to pre- and post-natal growth retardation, central nervous system abnormalities, a characteristic facial dysmorphology and other malformations. Characteristics specific to FAS are contraction of the middle third of the face with resultant shorter palpebral fissures, a flattened nasal bridge, upturned and shortened nose, a smooth philtrum and a thin upper lip (Astley & Clarren, 1995). The palpebral fissure lengths (PFL), inner canthal distances (ICD) and interpupillary distances (IPD) are some of the distances that can be measured in order to diagnose FAS, (see figure 1.1). Circularity measurements of the upper lip can also be used in diagnosing FAS (Astley *et al.*, 2002).

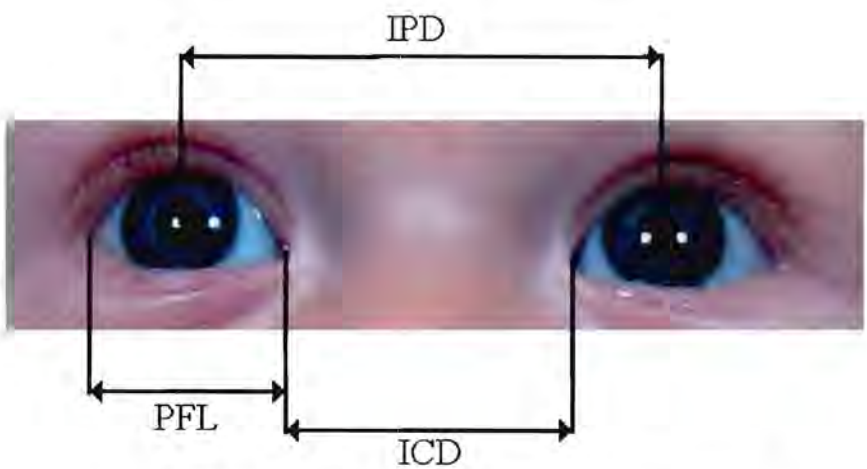


Figure 1.1: Facial features used in FAS diagnosis

Measurements of these facial phenotypic features of FAS help in identifying the syndrome. Accurate diagnoses of the facial phenotype are normally achieved by trained dysmorphologists who perform anthropometric measurements of the face,

which are compared to standardized ranges. Because FAS is widespread, large-scale studies are required in South Africa to determine which communities are at risk. Direct measurement is not practical if large numbers of children are examined, due to the time and cost involved.

An easily executed, cost effective stereophotogrammetric method to measure the facial dysmorphology of children in the diagnosis of FAS has been developed in the MRC/UCT Medical Imaging Research Unit (Meintjies *et al.*, 2002). It is an accurate, timesaving and easily controlled method that can be used in remote areas if necessary. Points are identified individually on each of a pair of stereo images by clicking with the mouse on the point in both images (as described in the literature review), and therefore it is possible that the person doesn't accurately mark exactly the same point in each image. This may result in less accurate three-dimensional coordinates of the points. Furthermore, it is time-consuming to select points on both images, and it would be quicker to select the points on a single image. Selecting points on one image of a stereo image pair and matching these points automatically on the second image is a step towards a fully three-dimensional solution.

The main motivation of the proposed project was to improve the above method for the effective screening and diagnosis of FAS. Stereo matching software was developed in an attempt to solve the above-mentioned inaccuracies and the reconstruction of features and faces in three dimensions was investigated. The stereophotogrammetric FAS screening instrumentation is available and thus it is possible to capture left and right images of a persons face. A database containing such images is also already available and was used in the research described here.

1.2) Objectives of the Study

The first objective was the development of stereo matching software that would identify and match the areas containing the same features in an image pair, without a person needing to mark a point on both images.

The second objective was to obtain three-dimensional coordinates of the matched areas and to determine their accuracy.

The third objective was to investigate the feasibility of constructing three-dimensional images of features and faces from which facial measurements can be made.

Matlab (Mathworks, 2004) was used to develop the necessary software, and available images and instrumentation were used to aid in the software development. Features of the eyes and mouth were focused on initially, but the whole face (including cheeks etc.) was also looked at. Since areas such as the cheeks don't have distinct features, a type of speckle texture projection was used to create features on these areas. These features made it easier to match areas in the different images for three-dimensional reconstruction. Other methods of image enhancement and the use of infrared light to improve matching were also investigated.

The developed software was evaluated in order to determine its effectiveness.

1.3) Outline of Thesis

Chapter 2 gives a literature review of stereophotogrammetry and stereo matching. Previously developed stereophotogrammetric systems are investigated, and the concept of stereo matching is broken down into its components. Literature on image enhancement techniques is also included.

Chapter 3 gives a literature review of three-dimensional reconstruction. Camera calibration and the theory around it are outlined. Different methods of obtaining 3D coordinates and reconstructing a 3D surface are also covered.

Chapter 4 describes the resources used in the project, including hardware and software, while chapter 5 explains what image enhancement techniques were applied.

Chapter 6 and 7 cover the methodology and results of the image matching and 3D reconstruction process developed. The components of the matching method are explained thoroughly in chapter 6, while the matching accuracy and runtime obtained from different sets of images are discussed. Chapter 7 describes how the 3D

coordinates were obtained and how they were used for three-dimensional surface reconstruction.

Chapters 8 and 9 conclude the thesis with a thorough discussion of the methods developed. The conclusions are presented and some recommendations for future work are outlined.

1.4) Ethics Approval

The photographs that were used in the project were obtained during a related project entitled "A new population screening method for Fetal Alcohol Syndrome". For this Prof. Denis Viljoen, head of Human Genetics at the University of the Witwatersrand and the National Health Laboratory Service, obtained an ethical clearance certificate from the University of the Witwatersrand's Committee for Research on Human Subjects (protocol number M990516).

An ethics approval from the Research Ethics Committee, Faculty of Health Sciences, University of Cape Town (UCT) was also obtained on 21 May 2003 (Rec. reference number 142/2003).

Chapter 2

Stereophotogrammetry and Matching

2.1) Introduction to Stereophotogrammetry

Conventionally, facial measurements have been obtained with direct methods, using standard anthropometric equipment including calipers, measuring tape and protractors. These methods are however intrusive and have several sources of error such as distortion induced by pressure on soft tissues. It also has difficulties where certain soft tissues e.g. around the eyes are very sensitive (Burke, 1971).

As an alternative to direct measurements, indirect measurements such as photogrammetry can also be used to obtain facial measurements. Photogrammetry involves the use of standard photographs to take measurements from objects such as the face (Farkas *et al.*, 1980). This overcomes many of the difficulties of measuring the face directly, since there is less interaction with the subject. Photogrammetry is not as invasive and is less time consuming. Stereophotogrammetry makes use of two pictures of the same object taken from different angles. With these two pictures, three-dimensional information (coordinates of points visible on both images) of the object can be obtained (Ras *et al.*, 1996). Linear measurements obtained from stereophotogrammetric techniques are comparable to direct anthropometrical measurements.

The human body, including the face, is a three-dimensional structure. Therefore it can be seen that measurements from two-dimensional images may be inaccurate and it would be cumbersome to adjust these measurements to obtain accurate results. Astley and Clarren performed measurements of facial features on two-dimensional photographs and had to modify the results of measurements of the palpebral fissure lengths to adjust for the effect of measuring a facial feature that is off the midline of a planar photograph (Astley & Clarren, 2001). This indicates a need for stereophotogrammetric systems to obtain effective three-dimensional measurements.

Applications of three-dimensional reconstruction of features of the face include the measurement of facial volume changes during human growth and development (Ferrario *et al.*, 1998), three-dimensional prisoner face capture to replace standard police station "mug shots" (Siebert & Marshall, 2000), and also evaluation of cleft-lip patients in order to assess the influence of surgical lip repair (Ayoub *et al.*, 2003).

2.2) Working Stereophotogrammetric Systems

Stereophotogrammetry methods developed so far for the purpose of obtaining three-dimensional information of the human body (and especially the face) include C3D (Siebert & Marshall, 2000), the ELITE (ELaboratore di Immagini Televisive) system (Ferrario *et al.*, 1996) and 3DFM (Three-Dimensional Facial Morphometry) (Ferrario *et al.*, 1998). A stereophotogrammetric method to measure the facial dysmorphology of children in the diagnosis of FAS has also been developed in the MRC/UCT Medical Imaging Research Unit (Meintjies *et al.*, 2002).

2.2.1) The ELITE System

The ELITE system consists of two charged-coupled device (CCD) cameras (Ferrario *et al.*, 1996) that record the subject (a human face), real time hardware for the recognition of markers (infrared stroboscope that illuminates markers) on the subject, and software for the three-dimensional reconstruction of the *(X,Y,Z)* coordinates of landmarks relative to the reference system. Landmarks are certain features on the face that are clearly visible (e.g. the corner of the eye) or that can be located by palpation (examination by touch). The landmarks that aren't visible on the recorded images are marked with markers. Before each acquisition session, the system needs to be carefully calibrated to correct the optical and electronic distortions of the images, and to obtain actual metrical data. During acquisition, the subject needs to sit in an upright position on a stool so that stereo views of the subject's face can be obtained with the positioned cameras. A left- and right-side image is recorded, with a 90°-angle difference between the two, the reflective markers placed on the subject's face are lit up with the aid of an infrared stroboscope. These images are then used

to obtain three-dimensional coordinates of landmark points, which are then used to obtain three-dimensional measurements of the face.

2.2.2) 3DFM

"Three-dimensional facial morphometry" (3DFM) is a three-dimensional system, which provides real three-dimensional data that is independent of head posture (Ferrario *et al.*, 1998). It makes use of two high-resolution infrared sensitive CCD video cameras coupled with a video processor, which provides the three-dimensional coordinates of the centre of gravity of landmarks marked on a subject's face. Image acquisition and three-dimensional data is obtained in a similar manner as described above (with the ELITE system) and the results are used to calculate facial volume, so that facial volume changes during human growth and development can be recorded.

2.2.3) C3D

The C3D system was developed by the collaboration between the Turing Institute and Glasgow University and is a 3D image acquisition system based on white light speckle texture projection photogrammetry. It is used for human body imaging (Siebert & Marshall, 2000), and relies on placing cameras at different angles to one another (camera-camera base line triangulation), to perform depth detection. C3D consists of 2 pods, and each pod consists of 3 cameras (figure 2.1).

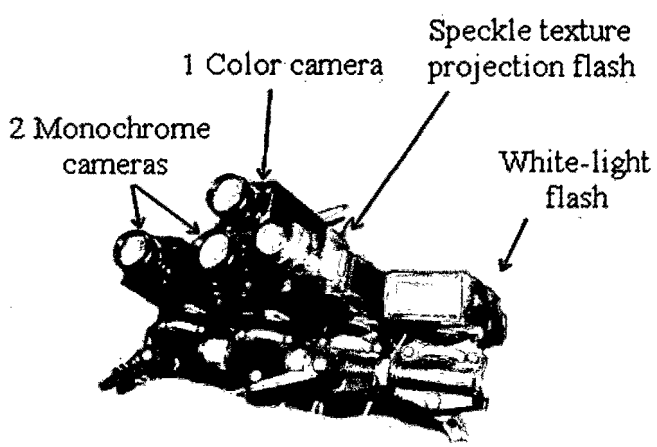


Figure 2.1: Components of a capture pod used in C3D, (Hajeer *et al.*, 2002)

Two monochrome cameras serve to form a stereo baseline and are synchronized to capture images illuminated by special texture flash projectors. A third central colour camera is synchronized to capture the natural photographic appearance of the subject under normal white-light flash, just 20 milliseconds after the flash texture stereo-capture. This texture appearance is eventually "covered" over the constructed 3D model of the face (Hajeer et al., 2002).

C3D also adopts a patented algorithm based on multi-resolution image correlation-based image matching. This algorithm processes stereo image pairs to produce metrically accurate 3D computer graphics models (Ayoub *et al.*, 1998).

In order to determine the detailed geometric configuration of all the cameras, a calibration process based on photogrammetric techniques is used. A calibration target with known dimensions is presented and captured by the cameras from different target poses. The images are processed to find desired coordinates on the target. These are used to fit an approximate geometric model of each camera and its respective relative orientation on the target. The Direct Linear Transform (DLT) (Abdel-Aziz & Karara, 1971) is used for this purpose.

The DLT estimates the three-dimensional coordinates (X,Y,Z), of landmarks by a linear transformation of their two-dimensional coordinates (x,y) in multiple images (Abdel-Aziz & Karara, 1971; Stevens, 1997; Wong, 1975). Variables relating (x,y) to (X,Y,Z) are estimated and with these the three-dimensional coordinates are determined.

C3D is used in advanced morphometric evaluation of cleft-lip patients in order to assess the influence of surgical lip repair (morphometry has to do with measurement of shape and structure in biology). This 3D imaging system is noted to be reliable in recording facial deformity (Ayoub *et al.*, 2003).

C3D has also been used for the planning of maxillofacial operations (Ayoub *et al.*, 1998) and can further be applied to capture 3D models of the human body for applications in law enforcement, medical and commercial applications.

2.2.4) Screening for FAS

In the stereophotogrammetric method developed in the MRC/UCT Medical Imaging Research Unit (Meintjies *et al.*, 2002), a pair of stereo photographs is taken of children's faces and the characteristic facial features of FAS are measured from these photographs. A pair of high-resolution digital cameras obtains left and right images simultaneously (see figure 2.2), while a calibration frame with eleven markers (with known three-dimensional coordinates) is visible on both images.

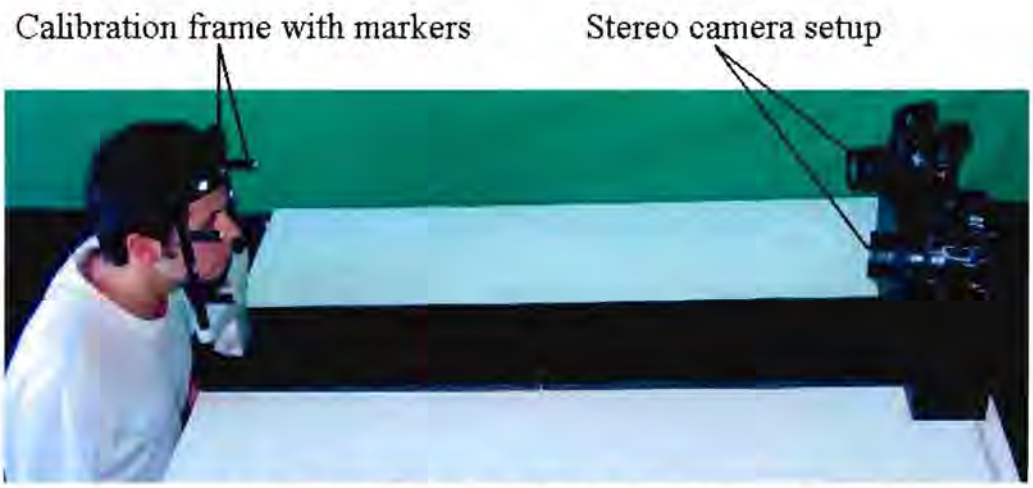


Figure 2.2: The use of stereophotogrammetric equipment

The images are then calibrated and the coordinates of the markers on both images are determined. The DLT is then used to transform two-dimensional image coordinates into three-dimensional object-space coordinates. The relevant points are selected on the images by the user who clicks with a mouse on corresponding points on both images. This way certain features such as the palpebral fissure lengths (PFL) and interpupillary distances (IPD) are measured (refer back to figure 1.1). Software is used to calculate necessary measurements. In order to test the system, two inexperienced investigators executed the measurements, which were first compared to one another, and then with measurements taken by dysmorphologists (specialists in malformations) to determine their accuracy. Measurements by the investigators and dysmorphologists were similar, indicating that the system eliminates the need for specialist participation, thus reducing the time and the cost of screening.

2.2.5) Discussion

The ELITE system and 3DFM system use charged-coupled device (CCD) cameras that capture the subject, real-time hardware for the recognition of markers placed on subjects' faces, and software for the 3D reconstruction of landmark *(X,Y,Z)* coordinates. The process of placing landmarks on the face is time- and labour consuming and cannot always be performed consistently between consecutive sessions. In the FAS screening system, the user needs to identify corresponding points manually (which may result in inaccuracies). The C3D system uses expensive instrumentation to obtain its results. All the above properties are either time consuming or expensive, and these are challenges I hoped to overcome with my project.

2.3) Stereo Matching

Depth can be obtained from two or more photographs of an object with stereo matching. Stereo matching can be seen as a part of the stereophotogrammetric process, and is applied in a variety of fields. Examples in medicine include three-dimensional reconstruction of the actin cytoskeleton from stereo images (Cheng *et al.*, 2000), and the analysis of metaphyses and joints in skeletal collections (Kearfott *et al.*, 1993), which are discussed in paragraph 3.2. An approach to stereo matching where a net of corresponding points between two images is constructed has also been used for three-dimensional reconstruction of a human face from two images, taken at different angles (Kostousov & Molochnikov, 2002).

Stereo matching can be broken down to the following four steps:

1) A stereo pair of images is obtained.
2) An image feature of interest is located in one image.
3) The other image of the stereo pair is examined to identify the corresponding image feature.
4) Three-dimensional information is obtained based on the corresponding features, using disparity (discussed in paragraph 2.3.2) or camera calibration (discussed in paragraph 3.1).

Correspondence and disparity play an important part in the stereo approach.

2.3.1) Correspondence Problem

In this part of stereo matching a point, feature or segment that corresponds (being similar or equivalent according to matching specifications, e.g. pixel intensity) in the two images is matched. This is one of the most significant concerns that arises in stereo matching and is referred to as the correspondence problem. This means that the location of the image part corresponding to the relevant three-dimensional object part must be known exactly in one image plane and located in the other image plane. It is better to match segments or features instead of points, since this reduces the complexity by reducing the number of matches that have to be carried out. Contours also provide stronger matching constraints, increasing the matching accuracy.

There are many computational approaches to solve the correspondence problem, and some prominent constraints in these approaches used are:

1) Uniqueness constraint, which implies that any feature in one image can have no more than one match in the other image (assuming a stereo pair of images).
2) Smoothness (continuity) constraint, which indicates that three-dimensional surface depth changes smoothly (is continuous everywhere) and abrupt changes only occur at boundaries.
3) Epipolar constraint, indicating that for any point in the left image, its matching point in the right image must lie on the corresponding epipolar line.
4) Similarity constraint, which implies that corresponding points are assumed to have similar intensity or colour. This way intensity is the main information used in stereo matching.

These constraints can be applied to improve results by way of feedback (Bokil & Khotanzad, 1995).

2.3.2) Disparity Problem

The difference between the positions of the corresponding points, features or areas on the left and right images must be determined. This difference in position contains

depth information and is called disparity. This implies that a depth map of a scene can be achieved by searching for corresponding points in an image pair. A depth map is a 2D representation of the normalized depth existing between a stereo camera setup and the objects in the overlapping field of view. A depth map is generated from a disparity map, which shows the disparity existing between corresponding points (a depth map and disparity map are related by a scale factor). Disparity can thus be used to calculate the three-dimensional coordinates and create a three-dimensional image through algebraic manipulations. To help improve matching results, a disparity gradient constraint can be applied. This means that for certain kinds of three-dimensional surfaces, the rate of increase or decrease of disparity from one matched area to the next must be within a certain limit (Sun, 1997). For example, if the change in depth in an image occurs smoothly, similar disparity values are expected to occur for adjacent matches. Therefore a candidate point with disparity value differing a lot from the disparity value of its adjacent matching candidates can be rejected as a false match.

2.3.3) Image Matching Techniques

Many approaches have been proposed and implemented for matching of corresponding points and determination of disparity in stereo images. These approaches can roughly be grouped into three categories: feature-based matching, structural matching and area-based matching.

In feature-based matching, defined features such as edges, lines or contours (high-level descriptors) are identified from each image and then matched. The feature-based methods are generally more accurate and less sensitive to noise (Cheng *et al.*, 2000), thus providing more precise positioning for the matching results. However, these features provide only an abstract description of the objects in the image, since the features only exist at certain pixels in the image. This leads to sparsely distributed disparity and in most cases insufficient data for 3D reconstruction. If this is the case, interpolation techniques can be used to obtain disparity values between known disparities obtained from the features. Furthermore, if a feature-based method is used, an extra step is needed for feature detection in the stereo images, which will increase computational cost.

Structural matching is sometimes referred to as relational matching. It establishes a correspondence from the primitives of one structural description to the primitives of a second structural description (Wang, 1998). A structural description is defined by a set of primitives and their interrelationships. For example, the structural description of an image may consist of image features and relationships among the features. Since structural matching techniques utilize not only image features but also topological and geometrical relations among the features to determine the correspondence, the image matching tasks can be fully automated without any *a-priori* information. The problems to be solved when using structural matching are mainly associated with the efficient acquisition of structural descriptions and the operational approach for their matching. The concept of structural matching was originally developed by experts in computer vision.

Area-based methods on the other hand refer to determining the correspondence between image areas and use only the intensity value of pixels as the matching element. Intensity value is a low-level descriptor, and is not necessarily related to the presence of objects in the image, i.e. the structural information is not explicitly considered in the matching. However, these methods have been applied successfully where stereo images have good textures (Sun, 2002), and they tend to give poor results where there is a lack of texture. Intensity values are available for all the pixels in the image, and a dense disparity map can be obtained without interpolation. Least squares matching and other correlation methods such as cross-correlation can also be employed as area-based matching methods, to reduce noise sensitivity and improve matching (see paragraph 2.3.4). Area-based matching is typically done using the normalized correlation between the two windows in the images to be matched. Normalized correlation takes into account differences in brightness and variance between the two images (Mikhail *et al.*, 2001):

$$N = \frac{E(w_1 w_2) - E(w_1)E(w_2)}{\sigma(w_1)\sigma(w_2)} \qquad (1)$$

Where:

N = Normalized correlation

w_1 = Defined window in one image (master) to be matched in second image (slave)

w_2 = Matching window in second image

$E(w)$ = Pixel values in matching window

$E(w_1 w_2)$ = Product of pixel values in two matching windows

$\sigma(w)$ = Standard deviation of matching window

A pixel in the master image is correlated by selecting a matching window around it, then calculating the correlation between this window and a matching window in the slave image. This calculation is repeated as the window in the slave image is moved along through the determined search range (or search window). The matching window with the highest correlation value indicates the best match.

An example of area-based matching is "template matching", in which the intensity profile of a specified entity that must be matched in an image forms a template (Schalkoff, 1989). The selection of an appropriate template size plays a big role in the accuracy of the matching. The process consists of searching for regions in an image where the image intensities and the template intensities regionally coincide (figure 2.3). In other words, the template "slides" over the image until the template and image intensity levels correspond.

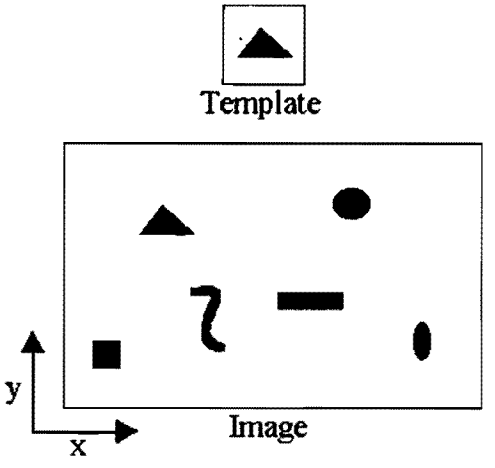


Figure 2.3: Simple template matching concept, (adapted from Schalkoff, 1989)

Other types of stereo matching methods such as pixel-based-, wavelet-based-, phase-based- and filter-based have also been developed (Sun, 2002).

2.3.4) Least Squares- and Cross-Correlation

Least squares correlation and cross-correlation can also be employed as area-based matching methods. In both cases the goal is to estimate the parameters of the transformation between the two images to be matched.

For stereo matching in an image pair, least squares can be described as a method of estimation from a fixed point in one image to obtain the corresponding point in the other image by seeking the solution for which the sum of the squares of the differences between the two corresponding points is least. In least squares correlation, one can assume the relationship between the left and right matching points as affine transformation (Lee *et al.*, 2003). For a given fixed point (x_l, y_l) in the left image, we can estimate the position (x_r, y_r) in the right image with the following affine transformations:

$$x_r = a_{11}x_l + a_{12}y_l + s_1 \qquad (2)$$

$$y_r = a_{21}x_l + a_{22}y_l + s_2 \qquad (3)$$

The left image is transformed to the right image using approximate values for the transformation parameters $(a_{11}, a_{12}, a_{21}, a_{22})$. Through a number of least squares estimation loops, the parameters of the affine transformation are updated and therefore also the position of the corresponding point (x_r, y_r). In this way, the corresponding point is searched for (based on the similarity of gray levels) in a two-dimensional direction. This method can generate very precise matching results. The transformation between the two images during the least squares correlation scheme can become complex due to the great number of observations to be made (one observation per pixel of the reference image). Therefore it is not easy to maintain high precision over large extents of stereo coverage, and it will require a large computation time.

Cross-correlation is also a very efficient tool when it comes to matching images. It is quite robust to noise, and can be normalized to allow pattern matching independently of scale and offset in the images. Cross-correlation (or intensity cross-correlation) is based on similarity of gray levels between two digital images, and takes the information of more than one pixel into account (ESAT, 2004; Mathworks, 2004). The position of the point corresponding to the reference point is given by the position of the maximum of the similarity measure, and the result is only accepted if a certain specified threshold is met. This is achieved by comparing local neighborhoods around the matched node through intensity cross-correlation. As a neighborhood, a small window of $(2P+1)*(2P+1)$ pixels centered around the node is taken. For the points (x, y) and (x', y') in two images, the similarity measure is obtained with the following equation, where all sums are taken over all pixels of the image to be matched:

$$C = \sum_{d=-P}^{P} \sum_{f=-P}^{P} (I(x-d, y-f) - \bar{I})(I'(x'-d, y'-f) - \bar{I'}) \qquad (4)$$

Where:

I, I' = Intensity values at a certain point in the two images respectively

$\bar{I}, \bar{I'}$ = Mean intensity value of the considered neighborhood in the two images respectively

P = Selected integer value to determine the neighborhood size

Ideally, features in the two images must be at the same scale and have the same orientation, since an inaccurate match might be obtained if the images are not similar.

Drawbacks of the least squares correlation and cross-correlation algorithms are that an initial a guess of the corresponding point must be made, which also must be 'close enough' to the true point (typically within 5 pixels), and problems may also arise with repeating patterns in the images, since this will result in different positions with high similarity.

2.3.5) A Previously Developed Stereo Matching Technique

The flexible net approach (Kostousov & Molochnikov, 2002) is a good example of a stereo matching technique. It focuses on finding a set of corresponding points between two images of the same object by creating a net over the area that needs to be matched. To create the net, certain points in one of the images are identified as nodes, with known two-dimensional coordinates. These nodes (lying on distinct features in the image) are then defined on the second image as the points with exactly the same coordinates as the points on the first image. The nodes are now connected to one another with a set of edges, thus forming a net on both images (see figure 2.4).

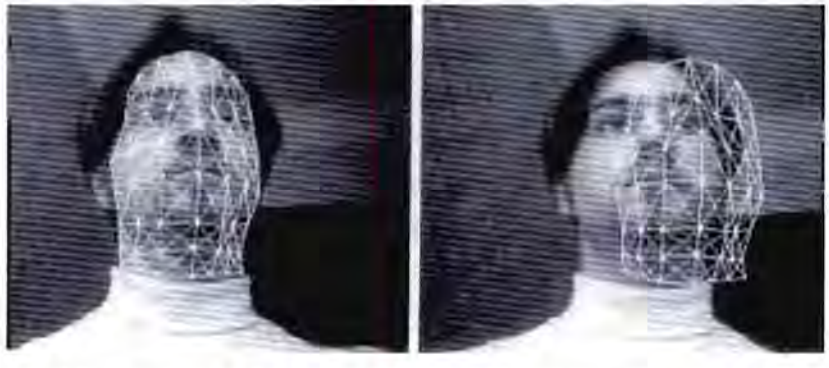


Figure 2.4: Initial position of the net on stereo images, (Kostousov & Molochnikov, 2002)

Now the nets in the two images are compared by comparing the pixel intensity of the pixels that are covered by profiles. These intensity profiles are presented by a pixel sequence along the line between two nodes in the net. The profiles in the second image are shifted and compared with the corresponding profile in the first image with the use of iterative mathematical techniques, until the best match is obtained (see figure 2.5).

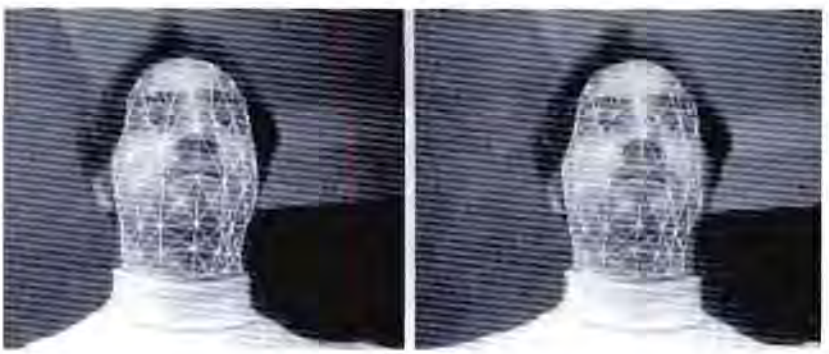


Figure 2.5: End position of the net on stereo images, (Kostousov & Molochnikov, 2002)

Once the best match is obtained, the coordinates of corresponding points in both images are indicated by the corresponding profiles and nodes. This data can then be used to obtain three-dimensional data of the stereo images.

2.3.6) Texture Projection

Many objects have a relatively smooth and featureless surface (e.g. human skin), making the matching process of corresponding points difficult. In order to improve the accuracy of corresponding matches in image pairs, structured lighting or special light patterns (e.g. grids, lines, concentric circles or random speckles) can be projected on the scene being imaged (Schalkoff, 1989; Mikhail *et al.*, 2001).

Structured lighting has been applied before to create active stereo vision systems. These systems, also called structured-light systems, offer an alternative approach to the use of two cameras. An artificial source of energy, such as a laser device, which projects a known pattern on the studied scene, can be used with stereo cameras or can even be used to replace the second stereo camera. Analyzing the deformation of the pattern in the images acquired by the cameras with respect to the projected

pattern provides 3D information. Some systems utilize a coded structured-light pattern, which allows unique codification of certain distinguished patterns in the projected light. Thus, the correspondence task that determines where each distinguished pattern comes from is directly solved. However, this kind of system has a drawback: it imposes constraints on the reflectance of the objects and on the illumination of the measuring scene (Dipanda *et al.*, 2003).

2.4) Image Enhancement Techniques

In order to improve the matching of corresponding points, features or segments, image enhancement techniques can be applied. These include feature enhancement techniques as well as edge detection.

2.4.1) Feature Enhancement

One of the most common defects of digital images is poor contrast, which results from a reduced image intensity amplitude range. Image contrast can often be improved by rescaling the amplitude of each pixel (Pratt, 1991), resulting in enhanced visibility of features. Two useful methods of feature enhancement are described here, namely histogram equalization and contrast stretching.

Histogram Equalization

Every image consists of its own unique set of pixels, each with its own intensity value. This can be represented with a histogram. In an image-processing context, the histogram of an image normally refers to a histogram of the pixel intensity values (figure 2.6). This histogram is a graph showing the number of pixels in an image at each different intensity value found in that image. The general distinctive brightness and contrast characteristics of an image may be determined from this histogram. For a typical grayscale image there are 256 different possible intensities (i.e. 0 – 255), and so the histogram will graphically display 256 numbers showing the distribution of pixels amongst those grayscale values. If nonzero intensity values in a histogram cluster mostly around small intensity values, the image represented by the histogram

will be fairly dark. On the other hand, if intensity values in a histogram are mostly nonzero for a range of large intensity values, the image will be too bright or even "washed out". The cause of either of these effects may be improper scene illumination or incorrect sensor sensitivity levels. Neither of these cases is desirable for subjective viewing, since the range of image intensities is quite narrow, resulting in poor contrast (Schalkoff, 1989).

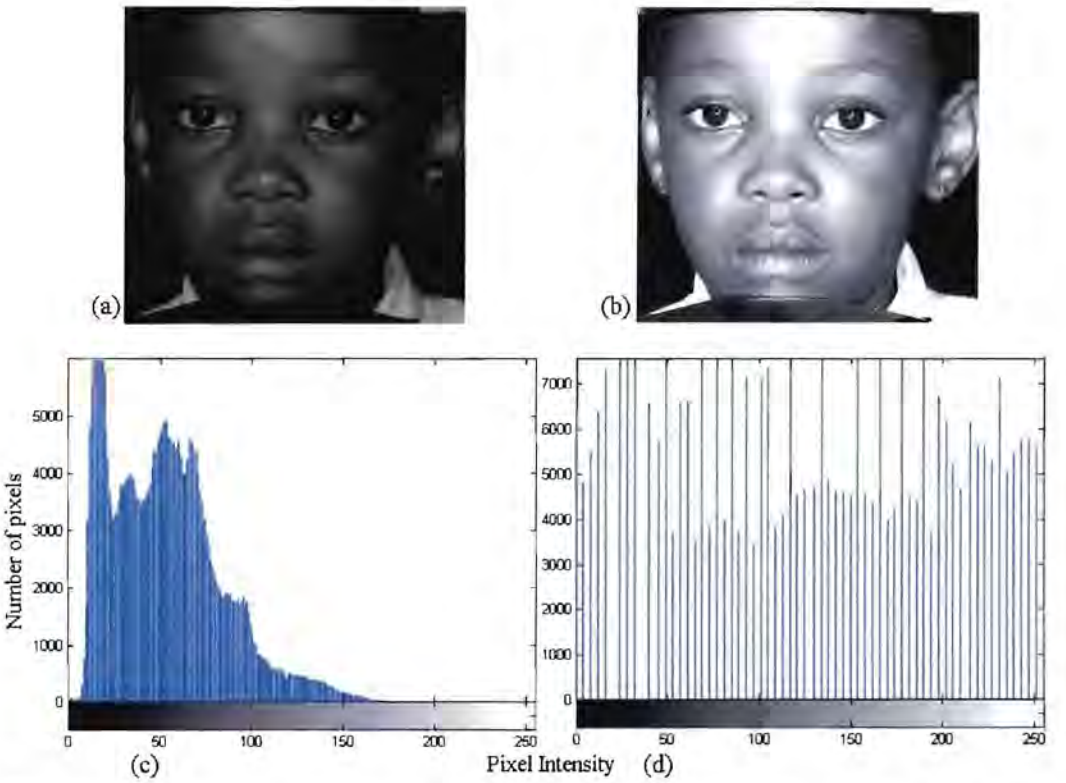


Figure 2.6: Image enhancement through histogram equalization: (a) Left facial image without enhancement. (b) Enhancement through histogram equalization. (c) Intensity histograms of image without enhancement and (d) of image after histogram equalization

Histogram modeling techniques such as histogram equalization provide an effective method for modifying the contrast of an image, by altering that image such that its intensity histogram has a desired shape. Histogram modeling may employ non-linear and non-monotonic transfer functions to map between pixel intensity values in the input and output images. Histogram equalization however, employs a monotonic, non-linear mapping which re-assigns the intensity values of pixels in the input image such that the output image contains a uniform distribution of intensities (a flat

histogram) (Boyle & Thomas, 1988). Histogram modeling techniques, and especially histogram equalization, are used in image comparison processes, since it is effective in detail enhancement (Gonzalez & Woods, 1992).

Contrast Stretching

In addition to histogram equalization, the image intensity values can be adjusted by contrast stretching (Gonzalez & Woods, 1992), resulting in increased contrast in an image. This maps intensity values so that values below a minimum and above a maximum specified intensity are clipped, and those in-between are spread, resulting in a better feature-enhanced image.

Referring to figure 2.7(a), it is shown that a transformation function is used for contrast stretching, where r and s denote the input and output gray levels respectively. The locations of points (r_1, s_1) and (r_2, s_2) control the shape of the transformation function, and contrast stretching is done by mapping values between (r_1, s_1) and (r_2, s_2). For example, if $r_1 = s_1 = 0$ and $r_2 = s_2 = 1$, the transformation is a linear function producing no change in gray levels. Intermediate values produce various degrees of spread in the gray levels of the output image, thus affecting its contrast. Values below r_1 and above r_2 are clipped, and values between r_1 and r_2 map to values between s_1 and s_2. If $s_2 < s_1$, the output image is reversed, like a photographic negative. The values of (r_1, s_1) and (r_2, s_2) can therefore be changed and adapted until an image with the best contrast is obtained (figure 2.7). This operation can also map the gray values of the background to a constant value and is therefore useful to suppress background noise while leaving the darker gray values in the image unchanged (Jähne, 1993).

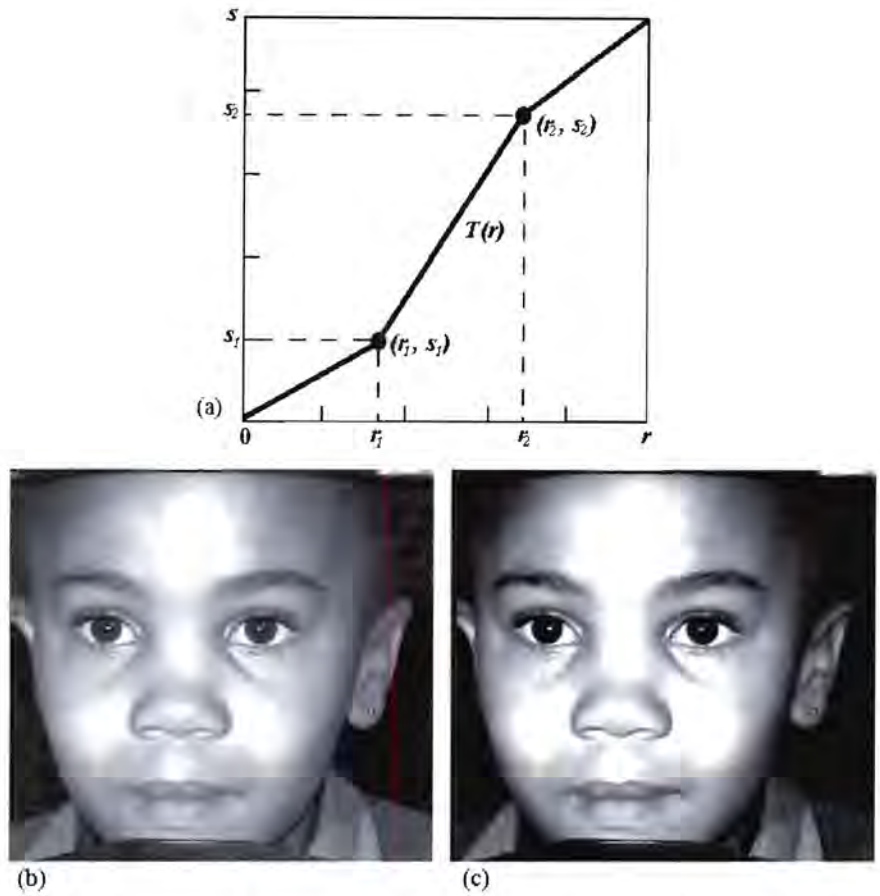


Figure 2.7: Image enhancement through contrast stretching: (a) Form of transformation function. (b) Original facial image. (c) Facial image after contrast stretching

2.4.2) Edge Detection

Changes or discontinuities in an image intensity value are important characteristics of an image since they often provide an indication of the physical range of objects within the image. These intensity discontinuities from one level to another (or between reasonably smooth regions) in images are known as edges.

There are two major classes of differential edge detection in images, namely first order derivative- and second order derivative edge detection.

First Order Derivative Edge Detection

First order derivative edge detection methods include Roberts-, Sobel-, Prewitt- and the Frei-Chen edge detectors.

The Roberts cross-difference operator works on the basis of obtaining diagonal edge gradients by forming running differences of diagonal pairs of pixels (Pratt, 1991), and is defined as:

$$G(x,y) = \left\{[G_1(x,y)]^2 + [G_2(x,y)]^2\right\}^{1/2} \quad (5)$$

$$G_1(x,y) = F(x,y) - F(x+1,y+1) \quad (6)$$

$$G_2(x,y) = F(x,y+1) - F(x+1,y) \quad (7)$$

Where:

F(x, y) = Original image

G(x, y) = Differential image

This method is however highly sensitive to small intensity changes in an image, and this problem can be reduced by using two-dimensional gradient formation operators, e.g. the Prewitt edge detector.

Prewitt has introduced a 3-by-3 pixel edge gradient operator (Pratt, 1991; PCI Geomatics, 2004) to be applied on an image *F(j,k)* described by the pixel numbering arrangement of figure 2.8:

A_0	A_1	A_2
A_7	F(j,k)	A_3
A_6	A_5	A_4

Figure 2.8: Numbering arrangement for 3-by-3 edge detection operators, (adapted from Pratt, 1991)

The template shown in figure 2.8 is applied in Prewitt edge detection as two separate masks (two 3-by-3 templates – figure 2.9), one for detecting image derivatives in *x* (image rows) and one for detecting image derivatives in *y* (image columns).

-1	0	+1
-1	0	+1
-1	0	+1

x

+1	+1	+1
0	0	0
-1	-1	-1

y

Figure 2.9: Prewitt convolution masks

An image is convolved with both masks, producing two derivative images (*dx* and *dy*). The strength of the edge at any given image location is then the square root of the sum of the squares of these two derivatives.

The Prewitt square root edge gradient is defined as:

$$G(x,y) = \left\{ [G_R(x,y)]^2 + [G_C(x,y)]^2 \right\}^{1/2} \qquad (8)$$

$$G_R(x,y) = \frac{1}{K+2}\left[(A_2 + KA_3 + A_4) - (A_0 + KA_7 + A_6) \right] \qquad (9)$$

$$G_C(x,y) = \frac{1}{K+2}\left[\left(A_0 + KA_1 + A_2\right) - \left(A_6 + KA_5 + A_4\right)\right] \qquad (10)$$

The Prewitt edge detector applies K = 1 in above equations, and in this formulation the row (*R*) and column (*C*) gradients are normalized. A discrete set of edges is produced when you threshold this result, and all edges are ignored that are not stronger than this specified sensitivity threshold. The Prewitt operator is more sensitive to horizontal and vertical edges than diagonal edges, while the reverse is true for the Sobel operator. The Sobel edge detector differs from the Prewitt edge detector in that the values of the north-, south-, east- and west pixels are doubled, i.e. K = 2 in equations *(9)-(10)*. The motivation for this weighting is to give equal importance to each pixel in terms of its contribution to the spatial gradient. Frei and Chen have proposed to use $K = \sqrt{2}$, so that the gradient is the same for horizontal, vertical and diagonal edges. The Frei-Chen-, Prewitt- and Sobel operators are better than the Roberts operator at identifying object edges since they're larger in size, providing averaging of small intensity changes (Pratt, 1991).

All the above edge enhancement operators have been derived heuristically. Canny on the other hand took an analytical approach to design such an operator. The Canny operator is based on a one-dimensional, continuous domain model of a step edge of amplitude h_E plus a specified σ (sigma) value as the standard deviation of an added Gaussian filter. Increasing the sigma value reduces the detector's sensitivity to noise, at the expense of losing some of the finer detail in the image.

Canny edge detection is performed by convolving a continuous domain-, one-dimensional edge signal *f(x)* with an antisymmetric impulse response function *h(x)*, which is of zero amplitude outside a range [*-W, W*]. An edge is then marked at the local maximum of the convolved gradient between *f(x)* and *h(x)*. For *h(x)*, the distance between peaks (denoted as *p)* is set to some fraction of the operator width factor *W* (see figure 2.10).

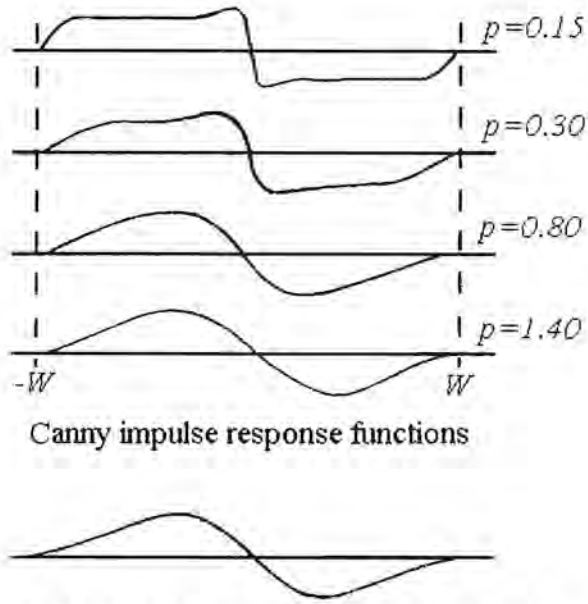


Figure 2.10: Comparison of Canny and derivative of Gaussian impulse response functions, (adapted from Pratt, 1991)

Figure 2.10 shows plots of the Canny impulse response functions in terms of *p*. It can be seen that for low functions of *p* the Canny function resembles a boxcar function, while larger values leads to a closer approximation of a derivative of Gaussian impulse response function. The Canny method is less likely than the other edge detection methods to be influenced by noise, and is more likely to detect true, weak edges.

Second Order Derivative Edge Detection

Second order derivative edge detection techniques employ a form of spatial second order differentiation to intensify edges. Edges are located if a significant spatial change occurs in the second derivative. The Laplacian second order derivative is briefly discussed.

The edge Laplacian of an image function *F(x,y)* in the continuous domain is defined as:

$$G(x,y) = -\nabla^2 \{F(x,y)\} \qquad (11)$$

Where the Laplacian is:

$$\nabla^2 = \frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2} \qquad (12)$$

The Laplacian *G(x,y)* is zero if *F(x,y)* is constant or changing linearly in amplitude (intensity). If the rate of change of *F(x,y)* is greater than linear, *G(x,y)* displays a sign change at the point where *F(x,y)* is emphasized. The zero crossing of *G(x,y)* indicates a presence of an edge. The negative sign in eq. *(11)* is present so that the zero crossing of *G(x,y)* has a positive slope for an edge whose amplitude increases from left to right or from the bottom to the top of an image.

The Laplacian edge detector can be adapted to form the Laplacian of Gaussian (LoG) edge detection operator, in which Gaussian-shaped smoothing is performed prior to applying the Laplacian. The continuous domain LoG gradient is:

$$G(x,y) = -\nabla^2 \{F(x,y) \otimes H_s(x,y)\} \qquad (13)$$

Where the impulse response of the Gaussian smoothing function is defined as:

$$H_s(x,y) = g(x,s)g(y,s) \qquad (14)$$

Here $g(x,s)$ and $g(y,s)$ denote continuous domain Gaussian functions with standard deviation s.

In many applications where edge detection is performed in order to outline objects in an image, the only performance measure of real importance is how well edge detector markings match with the visual perception of object boundaries. A human observer is usually able to recognize object boundaries in a scene quite accurately in

a perceptual sense. Thus, in the evaluation of edge detectors it is useful to assess them in terms of how well they produce outline drawings of an object that are meaningful to the human observer.

Chapter 3

Three-Dimensional Reconstruction from Stereo Image Pairs

Once matching is complete and a set of corresponding points in the stereo image pair is determined, the information obtained from these corresponding points can be used for three-dimensional image representation. The first step is obtaining three-dimensional coordinates and then the three-dimensional surface reconstruction follows. Obtaining 3D coordinates can be achieved with or without the aid of camera calibration.

3.1) Camera Calibration

The transformation process from real object space (3D) to image space (2D) by means of a camera is based upon the basic concept of perspective projection. Perspective projection is the transformation of data from a higher dimensional space to a lower dimensional space, and can be described using the concept of a pinhole camera model.

The camera lens is often referred to as a single point, called the centre of projection (Schalkoff, 1989), or the perspective centre (see figure 3.1), defining the pinhole camera model. Here the optical axis of the camera passes through the perspective centre (also sometimes referred to as the optical centre) and intersects the image plane at the principal point (PP). The distance from the perspective centre to the image plane (along the optical axis) is known as the principal distance, which is also often referred to as the focal length (Mikhail *et al.*, 2001). Furthermore, each point in object space reflects a ray of light that passes through the pinhole aperture (or perspective centre) of the camera to form a point on the image space. Thus the image is made up of a bundle of light rays reflected off the object and its surroundings. Each light ray can be defined by three collinear points (lying on the

same straight line), namely the object point, the image point and the perspective centre.

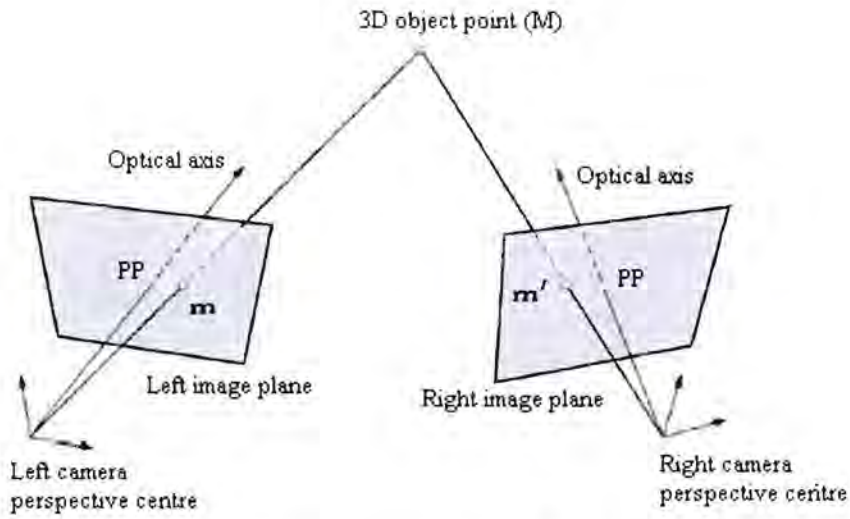


Figure 3.1: Geometry of the stereo camera setup, with m and m' the projections of the object point M onto the left and right image planes, (adapted from Garcia *et al.*, 2002)

The method to obtain the transformation from 3D object space to 2D image space is known as camera calibration (Byun & Nagata, 1996), and can be seen as the first step towards computational computer vision. Calibration can be used to reconstruct the 3D structure of objects from a stereo pair of images, after the correspondence problem is solved, and the process can be divided into two phases. Firstly, camera modeling deals with the mathematical approximation of the physical and optical behavior of the camera by using a set of parameters. Secondly, direct or iterative methods are used to estimate the value of these parameters (Salvi *et al.*, 2002). This consists of establishing the intrinsic- and extrinsic parameters, which enables one to exploit the 2D information captured in the images and which can be used to establish the epipolar geometry of the system. Intrinsic parameters determine how light is projected through the lens onto the image plane, i.e. the internal geometry and optical characteristics of the camera is determined. These parameters include focal length, the coordinates of the image centre (principal point), the first order radial lens distortion coefficient and tangential distortion. The extrinsic parameters measure the position and orientation of the camera with respect to a world coordinate system, thus describing the rotational and translational components of the transformation

between the world coordinate system and the camera coordinate system (Izquierdo & Ohm, 2000). For a comparative review of some of the most commonly used calibrating techniques, one is referred to Salvi *et al.* (2002).

The intrinsic and extrinsic parameters can be found using resection and intersection techniques. Resection is the determination of an image's position and orientation in space, while intersection refers to the determination of a point's object space coordinates from its coordinates in two or more images (this is done by intersecting the image rays from two or more images). These two operations can be combined to calculate the object coordinates and image orientation parameters through a calibration method called bundle adjustment, which is very accurate and flexible (Mikhail *et al.*, 2001). Another camera calibration algorithm for 3D point reconstruction is the Direct Linear Transformation (DLT) (Abdel-Aziz & Karara, 1971), which models the transformation between an image space coordinate system and the object space coordinate system as a linear function. The main distinction between the bundle adjustment and the DLT is that the DLT requires a 3D network of independently established control points, and if these control points are confined to a common plane, the solution becomes inaccurate. However, the DLT and bundle adjustment are two of the most widely used algorithms employed in photogrammetry.

3.1.1) Bundle Adjustment

As mentioned before, image rays connect an object space point, the perspective centre of the camera and the projection of the point on the image. A single image can be thought of as a bundle of these image rays converging at the perspective centre, where the rays have an unknown position in space. Bundle adjustment establishes the position and orientation of each bundle of image rays converging at the perspective centre, using the rays as well as the given ground control information (Mikhail *et al.*, 2001). Bundle Adjustment has a high degree of freedom, which increases the reliability of the solution, and is based on the following collinearity equations, showing the relationship between 2D image space and 3D object space:

$$x - x_o = f\left[\frac{m_{11}(X - X_L) + m_{12}(Y - Y_L) + m_{13}(Z - Z_L)}{m_{31}(X - X_L) + m_{32}(Y - Y_L) + m_{33}(Z - Z_L)}\right] \qquad (15)$$

$$y - y_o = f\left[\frac{m_{21}(X - X_L) + m_{22}(Y - Y_L) + m_{23}(Z - Z_L)}{m_{31}(X - X_L) + m_{32}(Y - Y_L) + m_{33}(Z - Z_L)}\right] \quad (16)$$

Where:

(x, y)	= 2D image coordinates of projected point in the image
(x_0, y_0)	= Principal point coordinates
(X, Y, Z)	= 3D coordinates of point in real object space
(X_L, Y_L, Z_L)	= 3D coordinates of the perspective centre
$m_{11} - m_{33}$	= Elements of the rotation matrix
f	= Focal length

Bundle adjustment has the advantage that external control points are not required. A more reliable evaluation of camera calibration and interior orientation can be made than with straight resection and intersection techniques. This is because bundle adjustment reduces the control requirements and therefore also the cost of control point surveying, and individual resection solutions. Furthermore, individual resection solutions are sensitive to errors in the control information used for each solution and may not be consistent (Mikhail *et al.*, 2001).

Bundle adjustment can be seen as the process of iteratively adjusting the camera poses and point positions in order to move toward the optimal least squares answer. However, bundle adjustment does need a good estimation value to ensure convergence and the computation of the solution is generally a non-linear procedure, which is expensive in terms of time and memory (due to large data sets and computations). Second-order non-linear least squares algorithms are usually employed, such as the Gauss-Newton and Levenberg-Marquardt methods (Bartoli, 2003). These methods iteratively improve sub-optimal parameter estimates by solving "normal equations" (equations that describe the best least square approximation for the distance of a vector from a set of *n* vectors).

3.1.2) The Direct Linear Transform

The DLT as first proposed by Abdel-Aziz and Karara (1971), models the transformation between an image space coordinate system and the object space coordinate system. Thus it transforms the 2D coordinates of a point on a number of images into the 3D coordinates of that point in real or object space. This transformation is modeled by a linear transformation, which is a 3-by-4 matrix called the perspective matrix of the camera (Ayache, 1991). The perspective matrix is also sometimes referred to as the transformation matrix (Salvi *et al.*, 2002), and is determined with the use of an image of a calibration frame (see figure 3.2). The positions in this image of the intersection points are known with precision, supplying the extrinsic- and intrinsic parameters.

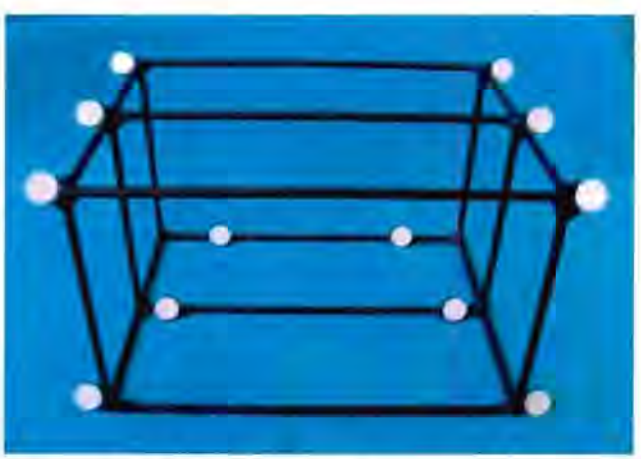


Figure 3.2: An example of a calibration frame used for calibration with the Direct Linear Transform

Image refinement parameters for lens distortion and film deformation were initially not included, but the DLT was later expanded to include these parameters (Mikhail *et al.*, 2001) and can be defined by the following equations:

$$x + \delta x = \frac{L_1 X + L_2 Y + L_3 Z + L_4}{L_9 X + L_{10} Y + L_{11} Z + 1} \qquad (17)$$

$$y + \delta y = \frac{L_5 X + L_6 Y + L_7 Z + L_8}{L_9 X + L_{10} Y + L_{11} Z + 1} \quad (18)$$

$$\delta x = (x - x_0)(K_1 r^2 + K_2 r^4 + \ldots) \quad (19)$$

$$\delta y = (y - y_0)(K_1 r^2 + K_2 r^4 + \ldots) \quad (20)$$

$$r^2 = (x - x_0)^2 + (y - y_0)^2 \quad (21)$$

Where:

(x, y) = 2D image coordinates of the projected point in the image

(x_0, y_0) = Principal point coordinates

(X, Y, Z) = 3D coordinates of point in real object space

$L_1 - L_{11}$ = Transformation parameters

$(\delta x, \delta y)$ = Total lens distortions in *x*- and *y*-directions

$K_1, K_2 \ldots$ = Coefficients of lens distortion at infinity focus

To achieve the transformation from 2D image space into 3D object space, the DLT requires a minimum of six 3D object space control points with known 3D coordinates (the bright circular reflectors on the frame in figure 3.2), which must be well distributed in 3D space. These control points are needed to solve the transformation parameters required for the transformation. These parameters are solved using a least squares adjustment, and details are described in Appendix B.

The DLT has the advantage that it's simple and it requires no initial approximations for the unknowns. A solution can still be obtained where a simultaneous bundle adjustment solution fails to converge caused by a lack of reasonable initial approximations. However, the solution is less rigorous and can be of a lower accuracy compared with the bundle adjustment.

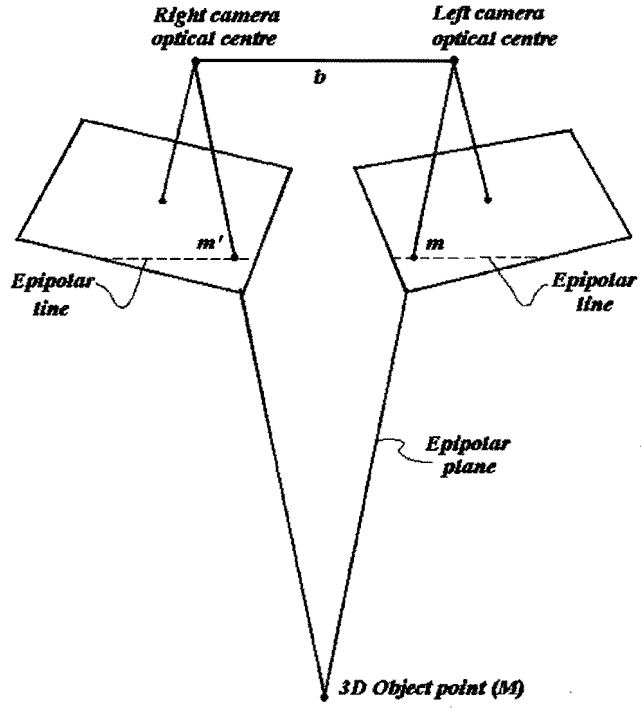


Figure 3.3: The epipolar plane and corresponding epipolar lines, (adapted from Mikhail *et al.*, 2001)

3.2) Obtaining 3D Coordinates Without Camera Calibration

A few possible methods to obtain 3D coordinates by using disparity were investigated. These include work done by Kearfott *et al.* (1993), Cumani & Guiducci (1997), and Cheng *et al.* (2000).

3.2.1) Stereo Imaging for Analysis of Metaphyses and Joints in Skeletal Collections

The metaphysis is the growing region at the end of long bones. This is where articulation with adjacent bone takes place, resulting in the occurrence of resistance to the linear and rotational shearing forces during motion. These forces vary as a function of stress in different species, and it's these differences that are of interest in the analysis of fossils by anthropologists.

The implementation of digital stereo imaging for analysis of metaphyses and joints in skeletal collections is done by implementing a digital stereo imaging technique for capturing bone surface contours by collecting corresponding points in the two images obtained from different angles (Kearfott *et al.*, 1993). This is achieved using an area-based correlation matching method. Depths of the matched points are then computed from the difference in the location of the point in the two images (disparity).

Kearfott *et al.* (1993) considered the following camera arrangement as shown in figure 3.4:

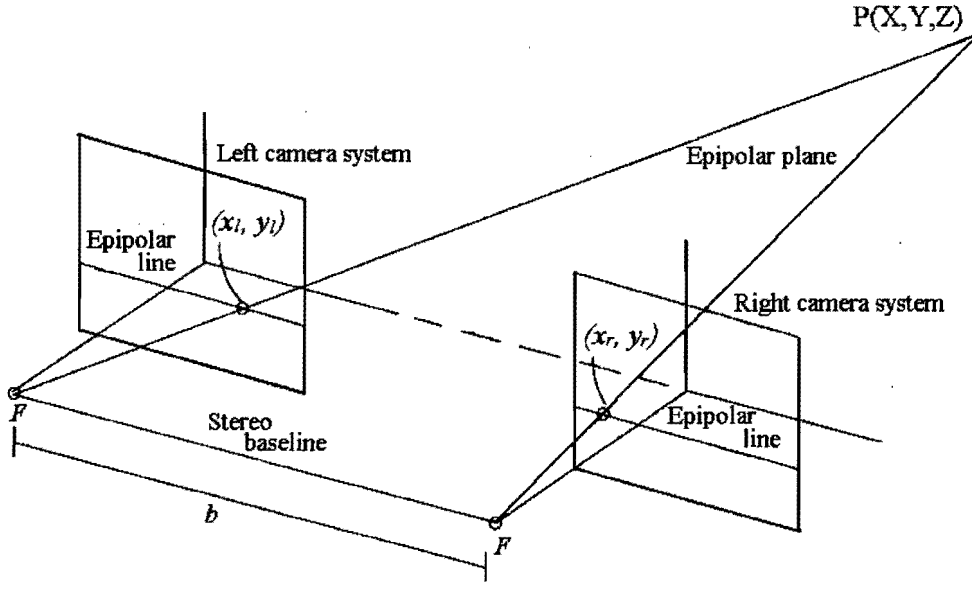


Figure 3.4: Camera arrangement, (adapted from Kearfott *et al*, 1993)

In figure 3.4, *F* is the focal point of each camera (left and right), and the line connecting the focal points (stereo baseline) forms a unique plane when connected to an imaged point $P(X,Y,Z)$. This plane is known as the epipolar plane, and the projection of $P(X,Y,Z)$ in the left image (x_l, y_l), and in the right image (x_r, y_r) must lie along an epipolar line. The epipolar line indicates the intersection of the epipolar plane and image planes. It must be noted that epipolar lines aren't always horizontal lines as indicated in figure 3.4, and that this is only the case when stereo cameras are oriented such that there's only a horizontal displacement difference between them.

Based on figure 3.4, the location of any point $P(X,Y,Z)$ can be calculated from:

$$X = \frac{b(x_l + x_r)}{2d} \quad (22)$$

$$Y = \frac{b(y_l + y_r)}{2d} \quad (23)$$

$$Z = \frac{bf}{d} \quad (24)$$

Where:

(x_l, y_l) = Left image corresponding 2D coordinates

(x_r, y_r) = Right image corresponding 2D coordinates

(X,Y,Z) = 3D coordinates

d = Disparity

b = Camera separation (baseline distance)

f = Camera focal length

Area-based matching is chosen for this application over feature-based matching, due to its simplicity. Object boundaries are contoured, and the object is divided into several one-dimensional search areas. Corresponding points in each search area are found and the depth of all the matched points is calculated. In this application, depth is calculated with eq. *(24)*, and the horizontal and vertical positions of a point are computed from its average position in both images, i.e.

$$X = \frac{(x_l + x_r)}{2} \quad (25)$$

$$Y = \frac{(y_l + y_r)}{2} \quad (26)$$

This does not account for image perspective changes, as equations *(22)* and *(23)* do (Kearfott *et al.*, 1993), but is adequate in this case since the surface of the bone metaphyses doesn't have significant depth variation. Furthermore, the imaging device is held close to the object and doesn't create a significant amount of perspective distortion. It is noted that the accuracy of depth computation decreased

with increasing object-camera distance. Tests performed in this case indicated that an object-camera distance of 45 cm gave depth variation within 0.76% accuracy, provided that the object was placed at a position where the cameras were calibrated. At 150 cm the inaccuracy increased to as high as 12 per cent.

The density of the matched points determines whether surface interpolation, applying spline-fitting or using *a priori* geometric models, is necessary. In this case it was decided to generate the surface of the object by drawing straight lines between each coordinate point on every row, thus creating a mesh. However, the main purpose of this work is to make measurements of specific areas, rather than visualizing exactly the three-dimensional surface.

3.2.2) Recovering the 3D Structure of Tubular Objects from Stereo Silhouettes

Cumani & Guiducci (1997) looked at recovering three-dimensional shape from a stereo pair of silhouette representations of an observed scene. More particularly, almost vertical tubular objects (AVTO's) are considered (i.e. generalized cylinders with some restrictions on their axis' shape and pose). This can practically be used in applications such as the realization of a vision system of an autonomous robot for harvesting asparagus in an open field (*Grattoni et al.*, 1993).

Cumani & Guiducci (1997) noted that, assuming unit focal length, the location of an imaged point *P(X,Y,Z)* is seen at:

$$(x_l, y_l) = (X/Z, Y/Z) \quad (27)$$

$$(x_r, y_r) = ((X-b)/Z, Y/Z) \quad (28)$$

$$d = x_l - x_r \quad (29)$$

$$Z = \frac{b}{d} \quad (30)$$

In equations *(27)-(30)* the variables relate to those defined for equations *(22)-(24)*. However, equations *(27)-(30)* hold for the ideal case. With real cameras, accurate calibration is needed and the search for corresponding points in the two images can

be restricted to epipolar lines. From eq. *(29)* it is also clear that the left and right images are aligned and that disparity only occurs on a horizontal level (along the x-axis).

By focusing on the particular case of AVTO's, it was possible in this case to devise a procedure that recovers a good estimate of object shape and location, even when mutual occlusions occurred.

3.2.3) 3D Reconstruction of Actin Cytoskeleton from Stereo Images

Cytoskeleton can be described as the internal framework of a cell, composed largely of actin filaments and microtubules. Actin filaments constitute a big part in cytoskeleton, where they form an interwoven 3D structural network, providing shape and form, and playing a big role in mechanical properties.

The three-dimensional reconstruction of the actin cytoskeleton from stereo images is necessary in order to understand the structural and mechanical properties of these actin filaments. It is done with a reconstruction approach that automatically reconstructs the three-dimensional structures of cytoskeletal polymers from images of the same subject taken at different angles (Cheng *et al.*, 2000). Corresponding points are found between these images and used to recover depth information about the structures. This process consists of feature representation, stereo matching and disparity refinement.

Three-dimensional information is extracted from photographs (in this case micrographs taken from different angles) via stereo vision algorithms by detecting corresponding points in the images and obtaining depth information from the disparity. Cheng *et al.* (2000) noted the following set of equations to obtain 3D information, under the assumption that the images are aligned:

$$\begin{bmatrix} x_l \\ y_l \end{bmatrix} = \begin{bmatrix} X\cos\theta + Z\sin\theta \\ Y \end{bmatrix} \qquad (31)$$

$$\begin{bmatrix} x_r \\ y_r \end{bmatrix} = \begin{bmatrix} X\cos\theta - Z\sin\theta \\ Y \end{bmatrix} \quad (32)$$

From equations *(29)* and *(30)*, the depth is computed as:

$$Z = (x_l - x_r)/(2\sin\theta) = d/(2\sin\theta) \quad (33)$$

$$d = (x_l - x_r) \quad (34)$$

In equations *(31)-(34)* the variables relate to those defined for equations *(22)-(24)*, and θ is the tilt angle between the 2 cameras.

The matching process is performed on a point-to-point basis, and reconstruction is achieved with a combination of feature- and area-based methods. After the 3D reconstruction, new structural information becomes available such as intersection points of the filaments, thickness and filament geometry (figure 3.5). Analysis of the measurements can aid in the understanding and modeling of cell mechanical properties and cytoskeletal dynamics.

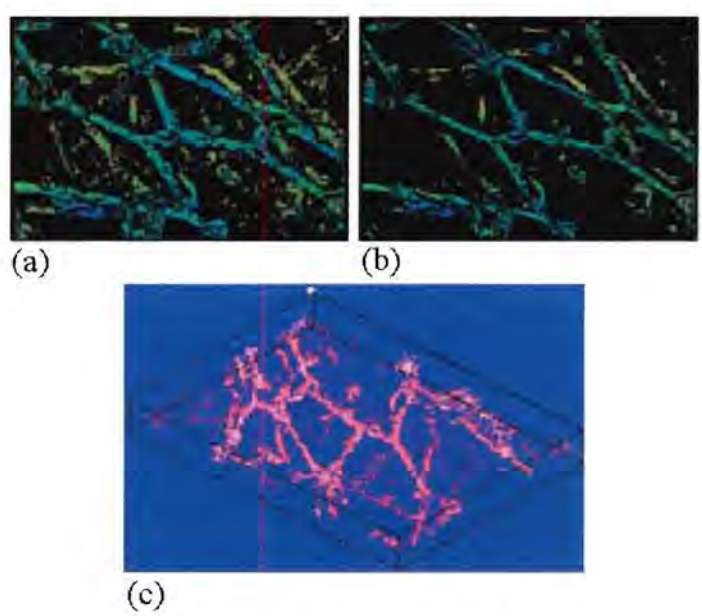


Figure 3.5: 3D reconstruction of cytoskeleton: (a)-(b) Example of stereo images of the cell cytoskeleton. (c) Reconstructed 3D cytoskeleton structure, (ICMIT, 2004)

3.3) Three-Dimensional Surface Reconstruction: Delaunay Triangulation and Voronoi Diagrams

Delaunay triangulation is well known in the geosciences and has been applied in a variety of other fields as well. Voronoi diagrams and its application are somewhat less known, although they are considered by many scientists as "a fundamental geometric data structure" (Mostafavi *et al.*, 2003). The Delaunay and Voronoi structures can be seen as the duals of each other, and therefore the construction of one automatically creates the structure of the other.

The Delaunay triangulation (DT) method originated from the study of structures in computational geometry. It was introduced by Voronoi (1908) and was extended by Delaunay (1934) for irregularly placed sites by means of empty circle methods. It is a very effective method to generate unstructured meshes, since it constructs a triangle's mesh from a defined set of points in 2D, by using all the points as vertices. DT has been applied in a variety of scientific and engineering fields, including the examination of alloys in metallurgy, mesh generation of finite element methods and cartography for town planning (Jin *et al.*, 2003).

The DT method is defined by an empty circle condition, meaning that one triangle is valid only if its circumcircle encloses none of the other defined points. The circumcircle of a triangle is a circle that can be drawn through all three the vertices (the meeting point of two lines, forming an angle) of that triangle. The steps to construct the DT in two dimensions is explained by the following steps (compare with figure 3.6):

1) Any two points are selected, and connected with a line (e.g. points 1 and 2 in figure 3.6(a)). Now the rest of the points are connected with these 2 points, and a triangle is established when there's no other point within the circumcircle of this Delaunay triangle (Delaunay circle). From figure 3.6(a) it can be seen that two triangles can be built (triangle A and B). If there are no points on the one side of the line connecting points 1 and 2, then only one triangle on the other side can be established. Triangles A and B satisfy the empty circle condition and are thus retained.
2) In order to continue generating the mesh, triangle A is selected. One of the sides of this triangle that is not connected to triangle B, is selected, and now plays the same role as the line that connects points 1 and 2 (see step 1). A

new triangle is obtained when there is no other point in the Delaunay circle (see fig. 3.6(b)). In this case, the determined Delaunay circle contains points 5 and 9 and the triangle is not accepted. Now the points inside (points 5 and 9) are used to construct new possible triangles. Triangle C is created via point 9 and is accepted since there are no other points inside its Delaunay circle.

3) Step 2 is repeated on the other side of triangle A that is not connected to another triangle. Triangle D is obtained as show in figure 3.6(d).
4) Steps 2 and 3 are repeated to obtain the rest of the triangles in order to connect all the points and create the complete mesh (see figure 3.6(d)).

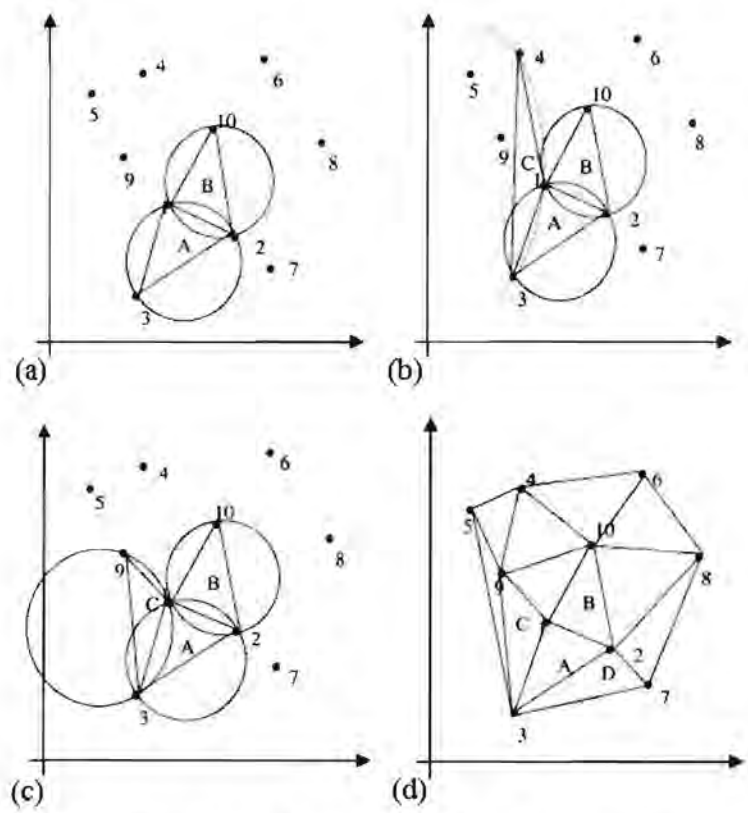


Figure 3.6: Delaunay triangulation algorithm, (Jin *et al*, 2003)

If one would connect the centres of the Delaunay circles mentioned above (i.e. between pairs of adjacent triangles), Voronoi diagrams are obtained, where one Voronoi edge will be associated with each Delaunay edge (figure 3.7(c)).

Consider an image/surface with a set of nodes, and that the surface is partitioned by assigning every node in it to its nearest neighbour. All points sharing the same

mutual nearest neighbour form a region of the surface, which is termed a Voronoi region (Mulchrone, 2002). A Voronoi diagram consists of these adjacent straight-sided regions (polygons) around the nodes, dividing the surface into regions of closeness to a given node (implying that any location in a particular polygon is closer to that polygon's generating point than any other).

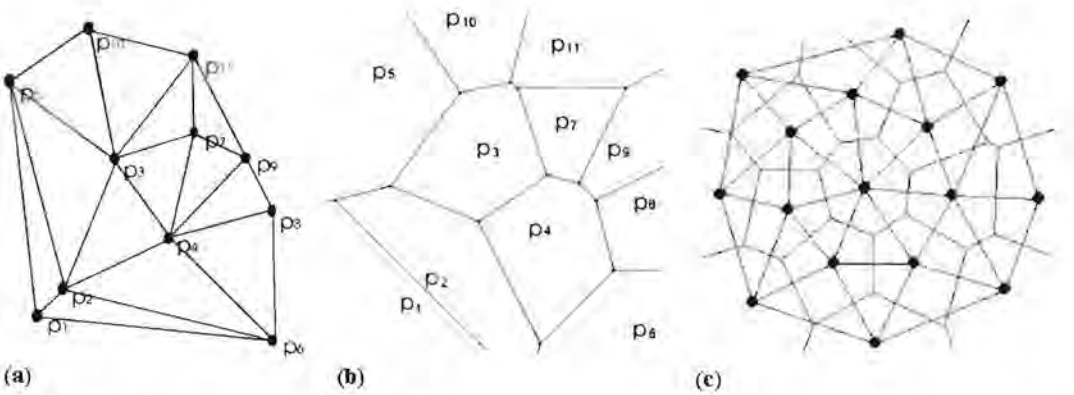


Figure 3.7(a): Example of Delaunay triangulation. (b) The corresponding Voronoi diagram. (c) Delaunay triangulation and dual Voronoi diagram illustrating the association, (adapted from Mulchrone, 2002 & Mostafafi *et al.*, 2003)

The computation time can be very high if there are a large number of vertices present in a mesh. However, too few will lead to poor quality reconstruction. This problem can be overcome if the number of vertices is reduced in featureless areas.

3.4) Three-Dimensional Surface Reconstruction: NURBS Curves and Surface Skinning

Three-dimensional surface reconstruction can also be achieved with the help of splines. Splines are piecewise polynomials with pieces that are smoothly connected together, with joining points on the polynomials called knots (Unser, 1999). B-splines are symmetrical, bell-shaped functions and were so named because they form the basis for the set of all splines. In this case, the B-splines also form the basis of NURBS curves, which can be applied for surface reconstruction.

Non-Uniform Rational B-Splines (NURBS) curves and surfaces are parametric functions which can represent any type of curve or surface. More specifically,

NURBS surfaces are objects created with curves instead of with polygons (where Voronoi diagrams are based on polygons), and facial animation and deformation have been achieved in the past by applying the NURBS curve (Huang & Yan, 2003). A NURBS curve of degree *p* is determined by control points, their weights and knots, and B-spline basis functions defined on the knots. The NURBS curve can be modified by repositioning its control points or changing its weights.

Using the NURBS representation, a process called Surface skinning can be performed (Piegl & Tiller, 1996). This is a process where a smooth surface is passed through a set of cross-sectional curves representing an object. In other words, for a given set of cross-sectional curves, a NURBS surface is sought that interpolates these curves at given parameter values. Figure 3.8 illustrates the skinning process.

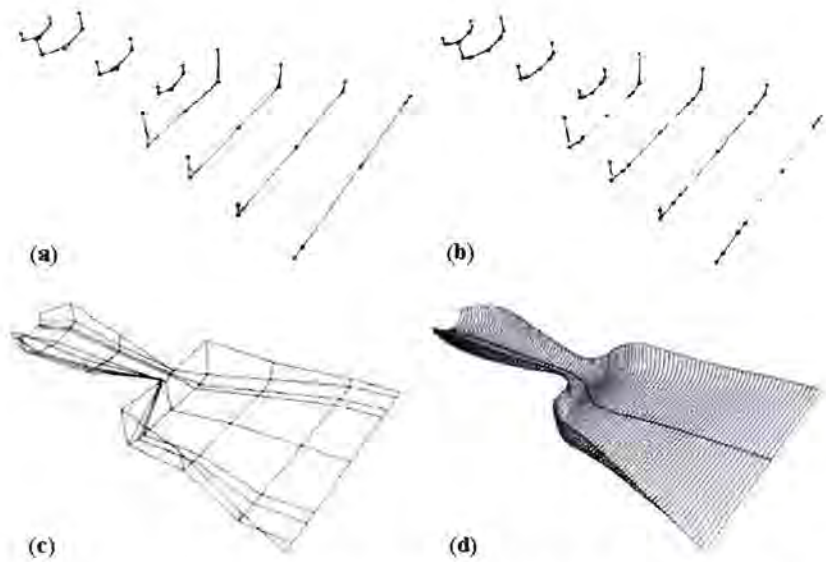


Figure 3.8: The process of skinning: (a) Cross-sectional curves. (b) Cross-sectional curves made compatible. (c) Control net of skinned surface. (d) Skinned surface, (Piegl & Tiller, 1996)

In figure 3.8(a) eight cross-sectional curves of a second degree are shown. All of them have the same number of control points, however, they are defined over different knot vectors. Figure 3.8(b) shows the curves made compatible by increasing the number of control points to nine. In figure 3.8(c) the net of the skinned surface is seen, while the surface itself is shown in figure 3.8(d).

This method admits great generality in that the cross-sectional curves can be of any degree and can be defined over different knot vectors. This flexibility however, comes at a high price since it results in an astonishing number of control points, e.g. it's likely that 50 cross-sections of various types require as many as 500,000 surface control points to interpolate. This problem can be solved if the cross-sectional curves are replaced by compatible approximations, and if approximation is used instead of interpolation. However, skinning requires lots of computer memory, and applying the approximations doesn't eliminate this problem.

3.5) Interpolation Techniques to Smooth the 3D Surface

It is not usually possible to obtain corresponding matches for all points in an image pair, and an interpolation step is necessary to fill the gaps. When a plane of discrete data points exists, two-dimensional interpolation can be applied to predict the value of intermediate datapoints between the known points within the plane. Interpolation between the points in an initial set of widely spaced matching points can give a complete depth map of the object (or a complete three-dimensional reconstruction of the object).

One of the most common interpolation methods implemented is bilinear interpolation, which presents a very low algorithmic complexity, and gives good results for scenes with low complexity (Izquierdo & Ohm, 2000).

Chapter 4

Resources

This chapter describes the hardware and software used to obtain digital images and to perform matching and three-dimensional reconstruction tasks on these images.

4.1) Hardware

Two different camera sets were used to obtain images for testing the developed software.

The first set of images was obtained during a FAS screening project conducted previously (Meintjies *et al.* (2002) - discussed in paragraph 2.2.4). The images were obtained by simultaneously photographing children's faces in a control frame with a pair of high-resolution digital cameras. The cameras were mounted 1.04 meters from the control frame, with a baseline distance of 0.26 m.

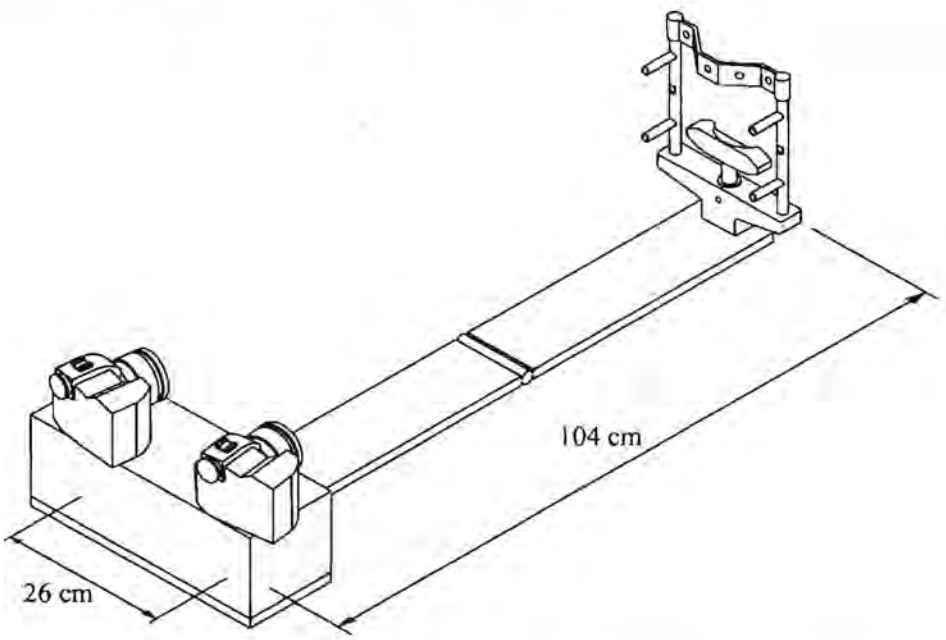


Figure 4.1: Apparatus designed for the Meintjies *et al.*, (2002) study

The cameras used are Sony DKC-FP3 Digital Still cameras, which were triggered simultaneously by remote control (figure 4.2). All the captured images have a resolution of 1344-by-1024 pixels and are saved as JPEG images. They were initially saved on two 8 MB memory sticks and from there downloaded to a notebook computer via an IEEE 1394 digital interface, where they were stored in a database. This equipment and related software were designed for the FAS screening project, and the images obtained were used for testing the software algorithms developed in the current project.

Figure 4.2: Sony DKC-FP3 Digital Still cameras and Sony PCG-Z600RE notebook computer

The second set of images was obtained with two Digital Smart Cameras. These cameras were designed by Electronic Development House (EDH) Stellenbosch, South Africa, and the current version is still a prototype (figure 4.3). Each of these cameras contains a StrongArm SA1110 206MHz processor that runs a version of Debian Linux, which is adapted for the cameras. Each camera contains a Sony CCD area imager that provides 256 level grayscale images with a resolution of 512-by-492 pixels. Lenses used on the camera are Yamano 16 mm manual iris lenses.

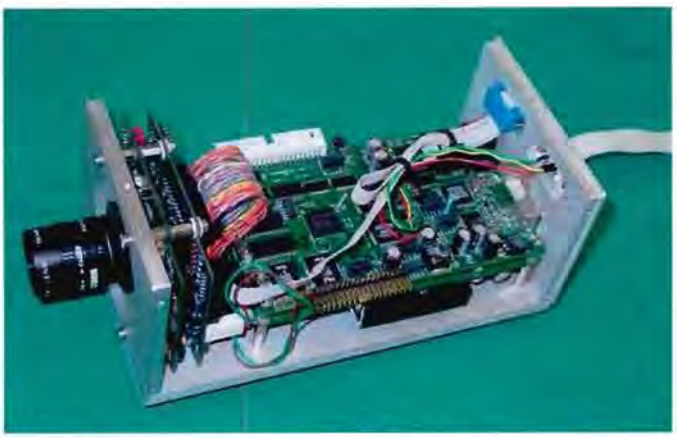


Figure 4.3: The Digital Smart Camera

The cameras are kept in a camera enclosure (figure 4.4), which contains a power supply unit and the hub. The hub is a 10MBps 8-port Ethernet hub that links the two cameras and a notebook computer across a network. A board of infrared LEDs is mounted on the front external part of the enclosure between the two lenses. The cameras are triggered from the connected notebook computer. Images were initially saved on the notebook computer.

Figure 4.4: The camera enclosure showing the lenses and infrared LEDs

With both camera sets, the images obtained were copied to the same desktop computer in order to run tests and develop the stereo matching software. Tests were performed on a 433 MHz, 127.0 MB RAM Intel Celeron Processor.

Texture projection was used in some of the image sets obtained from the Digital Smart Cameras, with the aid of a data projector that projected a texture pattern on to the face while the images were acquired.

4.2) Facial Image Pairs Used

The 1024-by-1344 colour images (figure 4.5) were read into Matlab where they were saved as three-dimensional matrices (M-by-N-by-3, where 3 represents red, green and blue). The images were then converted to grayscale images, which are two-dimensional images (M-by-N), with pixel intensities varying between 0 – 255 (with the former being black and the latter white).

(a) (b)

Figure 4.5: Example of colour image pair: (a) Left facial image. (b) Right facial image

It was decided to convert the images to grayscale since this simplified the matching process significantly. Colour images might be used in the future to see whether this might influence the matching. In order not to display any unnecessary features in the images, the images were then cropped so that only the face is visible in the images (figure 4.6).

(a) (b)

Figure 4.6: Cropping the facial images: (a) Original facial image. (b) Cropped facial image

The 512-by-492 pixel resolution grayscale images were photographed with an infrared flash, in order to determine whether this might influence the matching accuracy. The infrared light gives the images a different appearance, i.e. the face is brighter and the image brightness is more consistent (figure 4.7).

(a) (b)

Figure 4.7: Example of grayscale image pair with infrared flash applied: (a) Left facial image. (b) Right facial image

A second set of 512-by-492 pixel resolution grayscale images was later photographed with texture projection, in an attempt to improve the matching accuracy. The infrared flash diminished the intensity of the projection, and therefore the flash was blocked out when the images were taken with the texture projection

(figure 4.8). However, when the infrared flash was removed, the images appeared too dark and histogram equalization was applied (refer to paragraph 2.4.1). In this case the cameras were mounted roughly 1.1 meters from the object, with a baseline distance of 0.4 m.

Figure 4.8: Example of grayscale image pair with applied pattern projection: (a)-(b) Facial image pair obtained with infrared flash. (c)-(d) Facial image pair obtained without infrared flash and with histogram equalization

4.3) Software

All of the stereo matching software developed and used to obtain corresponding matches in an image pair and for three-dimensional reconstruction of the matched area, was developed in Matlab. Matlab version 6.5 Release 13 was used for this purpose, with extensive use of the image processing toolbox (Image Processing

Toolbox version 3.2). A descriptive list of all the programs developed is given in Appendix D.

Microsoft Excel was used for a statistical comparison between matched nodes and manually marked nodes to determine the matching accuracy of nodes around the eyes and mouth.

Finally, Microsoft Power Point was used in conjunction with the data projector in order to project texture onto the face during image acquisition.

5.1.2) Contrast Stretching

Contrast stretching also adjusts the image intensity values, as explained in paragraph 2.4.1. Different threshold values for the contrast stretching were applied, resulting in different intensity values for the images. This means that the images' intensity values were adjusted in order to emphasize desired features, until the best intensity image was created in order to help improve the matching accuracy.

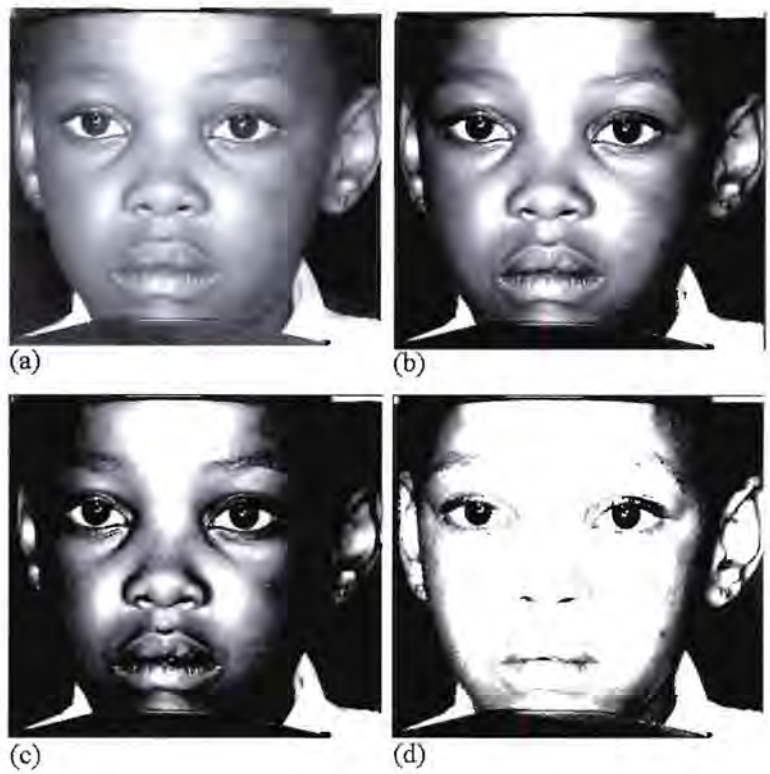


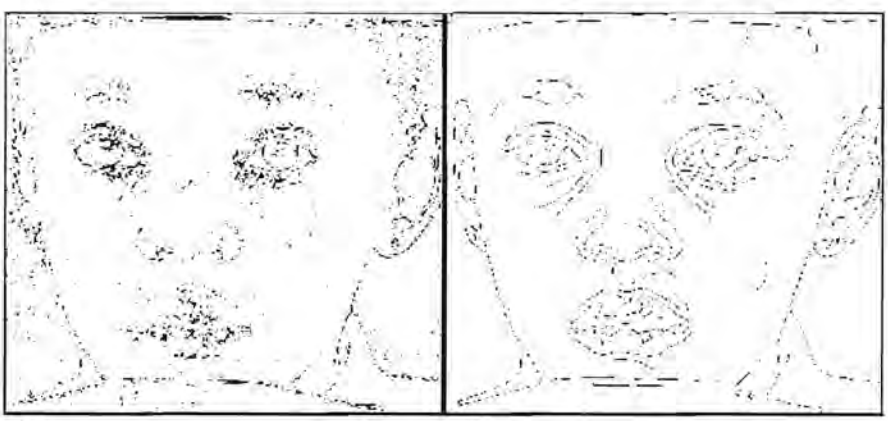
Figure 5.2: Image enhancement through contrast stretching: (a) Image after histogram equalization but with no contrast stretching. (b) Applying average threshold values. (c) Applying lower threshold values. (d) Applying higher threshold values

5.2) Applying Edge Detection Methods

Sobel-, Prewitt-, Roberts-, Laplacian of Gaussian- and Canny edge detection methods were considered. In each case the detected edges were displayed as a binary image with 1's where the function found edges and 0's elsewhere. The

Canny- and Prewitt edge detection functions gave the best results. Both methods were further tested by applying different sensitivity threshold values in order to see how the resulting set of edges influenced the enhancement of the image. It was found that the Canny edge detection method was the most effective. The optimum threshold values were determined and Canny was applied for enhancement.

Canny employs a derivative of a Gaussian filter to compute the gradient field of the image (Canny, 1986). It then interpolates the gradient field to identify edges as points that are locally maximum in the gradient direction of the image. Two thresholds are specified to detect strong and weak edges, and the weak edges are included in the output only if they are connected to strong edges (figure 5.3).



(a) (b)

Figure 5.3: Edge detection from the facial image displayed in figure 5.2(a): (a) Prewitt edge detection with a specified threshold of 0.05. (b) Canny edge detection with a maximum specified threshold of 0.15 and minimum threshold of 0.06

The resulting edges were copied (or redrawn) onto the facial images for matching, and this was done on separate occasions with an intensity of 0 (black) and 255 (white) to see which intensity produced the best matching results. Canny edge detection was applied in various combinations together with histogram equalization and contrast stretching, in order to enhance the image pairs before matching took place.

5.3) Texture Projection

In order to further improve the accuracy of corresponding matches in image pairs, a variety of texture patterns were created as slides in Microsoft PowerPoint and projected with a data projector onto the face as the photographs were taken. A few examples of the texture patterns developed and projected onto the face are displayed in figure 5.4. This projection especially helped in matching on smooth features such as the cheeks and forehead.

Figure 5.4: Examples of different texture projections applied

Chapter 6

Image Matching

After image enhancement, the corresponding points must be obtained through matching. This is achieved by selecting certain points in one of the images, and creating nodes as small square areas with the selected pixels as midpoints. These nodes are then copied to the second image and shifted to obtain the best match iteratively. Once the best match is obtained, the nodes indicate the coordinates of corresponding points in both images. The matching process and the results obtained are discussed in detail through the remainder of this chapter.

6.1) Methods: The Matching Process

After image enhancement techniques were applied, the matching process could begin. The matching method developed during the project was inspired by the flexible net approach presented by Kostousov & Molochnikov (2002), although the process differs significantly. In the algorithm described in this document, nodes are matched individually and from this a three-dimensional mesh is constructed. Image enhancement techniques are also applied to aid the matching process. The flexible net approach (Kostousov & Molochnikov, 2002) constructs a net on each image and matches intensity profiles presented by a pixel sequence along the line between two nodes in the net (refer to paragraph 2.3.5). The authors do not mention whether image enhancement techniques were applied in this approach.

In a pair of images, taken from the left and right at an angle, the distances between certain features appear to be different in the two images, due to parallax. Therefore, when creating a net covering certain features in the left image and matching the profiles represented by two connected nodes, or even matching the whole net simultaneously on the right image, certain areas will not be matched correctly, due to this error of parallax. Because of this inaccuracy, a modified technique of matching these image pairs accurately was implemented. This was achieved by creating only

the nodes on the left image, and matching these nodes one at a time on the right image. Since each node is a small square, matching isn't affected by the error of parallax. After all the nodes are matched correctly, lines are drawn between them to create a net. This net will then cover the same features in both images accurately, indicating the area of the object that will be reconstructed three-dimensionally.

6.1.1) Creating the Nodes

The matching process began by specifying the number of nodes, which would later determine the size and shape of the net. The left image from a facial image pair was displayed and the user manually selected the desired points where the nodes should be, and these node positions were then stored. Each node was defined as a small square with intensity of 255 (see figure 6.1). The midpoint of the square was located at the marked coordinate (the pixel on the image where the user selected the node to be). The node therefore consisted of white pixels surrounding the central selected point.

The pair of images was also modified in preparation for matching of the nodes, in that all pixels with intensity of 255 (white pixels) were changed to an intensity of 254 on the grayscale images. This was done in order to avoid errors when identifying the nodes on these images, since each node was identified for matching by its intensity of 255.

Tests were performed by matching nodes of size 9-by-9 pixels, 7-by-7 pixels and also 5-by-5 pixels in order to determine how the size of the node would influence the matching accuracy (figure 6.1). For the 1024-by-1344 pixel resolution colour images, tests were also performed with 11-by-11 nodes to compensate for the higher resolution of the images.

Figure 6.1: Illustration of nodes and the different node sizes. Nodes from left to right: 11-by-11 node, 9-by-9 node, 7-by-7 node and 5-by-5 node

6.1.2) Using a Search Window

By limiting the search space to an *m*-by-*n* sized search window instead of searching the whole image for a match, the runtime is reduced and the chance of obtaining false matches is also reduced.

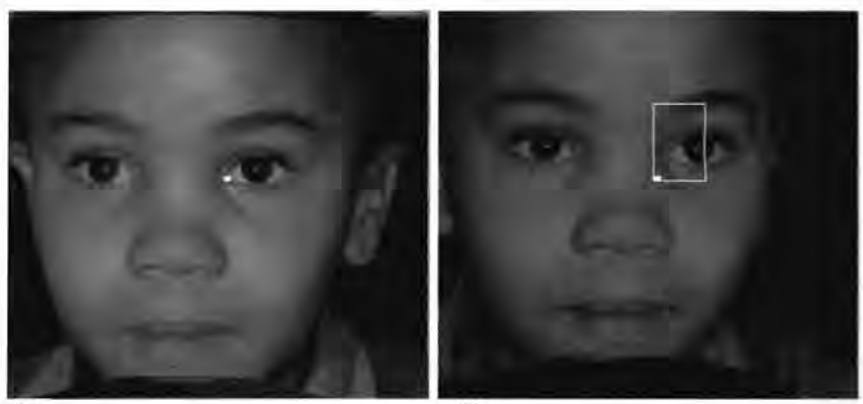
The size of the search window is determined by the image resolution and the angle between the cameras. The bigger the angle between the camera pair, the bigger the disparity between corresponding points on the stereo image pair will be. Therefore the size of the search window needs to be adapted to ensure that the corresponding point will be inside the search window. It is important to note that the smaller the search window, the smaller the chances of obtaining a false match, and that a correct match will be found more quickly with a smaller search window. However, the search window must be large enough to ensure that it contains the correct match.

The camera setup (Digital Smart Cameras developed by EDH) for the infrared images without texture projection was of such nature that the image pairs were almost aligned horizontally (small vertical disparity), but had a big horizontal disparity. Therefore, in order to find the corresponding point in the two images, a search window with a long base in relation to the height had to be used as indicated in figure 6.2.

(a) (b)

Figure 6.2: Infrared image pair to indicate search window: (a) Left image with marked node. (b) Right image with copied node and search window (size 115-by-15) inside which the best match was searched for

With the images obtained with the Sony DKC-FP3 Digital Still cameras (those without applied infrared flash), an almost square search window could be used (figure 6.3), since the image pairs had a similar horizontal and vertical disparity.



(a) (b)

Figure 6.3: Image pair to indicate search window: (a) Left image with marked node. (b) Right image with copied node and search window (size 40-by-60) inside which the best match was searched for

The head posture was not identical in all images of an image set. The search windows had to be large enough to cover the correct region in all images of each set.

6.1.3) Obtaining a Match

Once an image pair was enhanced, the search for corresponding points began. The nodes were then created on the enhanced left image and copied to the same coordinates on the enhanced right image. After a node was drawn on both the images, a vector for each image was created to store the position of every pixel that formed the node on each image:

$$\{Lp_n\}_{n=1}^{N} \quad (35)$$

$$\{Rp_n\}_{n=1}^{N} \quad (36)$$

Where:

$\{Lp\}$ = Left node position vector

$\{Rp\}$ = Right node position vector

n = Pixel number

N = Total number of pixels covered by the node in the image

The position vectors were used to obtain the intensity of all the pixels that were covered by the node, and these intensity values were then stored in different intensity vectors:

$$\{Li_n\}_{n=1}^{N} \quad (37)$$

$$\{Ri_n\}_{n=1}^{N} \quad (38)$$

Where:

$\{Li\}$ = Left node intensity vector

$\{Ri\}$ = Right node intensity vector

Although these nodes were drawn at the same coordinates on the left and right image, the node on the right image didn't cover the same features as on the left image, due to the different camera viewpoints from which the pair of photographs was taken. The matching of the left and right nodes entailed moving the right node

vertically and horizontally, one pixel at a time, until the best match was obtained. The number of steps that the right node was moved was determined by the size of the search window.

The pixel intensities of the right node were saved in a new right node intensity vector after every step:

$$\left\{Ri_{n,m}\right\}_{n=1,m=1}^{N,M} \qquad (39)$$

Where:

$\{Ri\}$ = Right node intensity vector

m = Step taken to shift the node

M = Total number of steps that the node must be shifted (total size of the search window)

n = Pixel number

N = Total number of pixels covered by the node in the image

Now the intensity difference between the right node pixels and the left node pixels had to be obtained after each shift of the right node:

$$\left\{Di_{n,m}\right\}_{n=1,m=1}^{N,M} = \left|\left\{Li_n\right\}_{n=1}^{N} - \left\{Ri_{n,m}\right\}_{n=1,m=1}^{N,M}\right| \qquad (40)$$

Where:

$\{Di\}$ = Absolute difference vector created after each step

Thus after each step an "absolute difference vector" was created to compare the left node with the relocated right node. This absolute difference vector was further used (after each step) to compute the average intensity difference between the left node and right node pixels. The average intensity difference was then stored in a "matching matrix" $\{A_{r,c}\}$ from which the coordinates of the best match would ultimately be obtained. Two different methods to obtain the average intensity difference between the left node and right node pixels were tested on the same data and compared in order to see which method provided the most accurate results.

These two methods were: Applying the sum of absolute differences, i.e. using eq. *(41)*, and applying the sum of squared differences, i.e. using eq. *(42)*.

$$A_{r,c} = \frac{1}{N}\sum_{n=1}^{N} Di_{n,m} \qquad (41)$$

$$A_{r,c} = \sqrt{\frac{1}{N}\sum_{n=1}^{N}\left(Di_{n,m}\right)^2} \qquad (42)$$

$$m = r * c \qquad (43)$$

Where:

$A_{r,c}$ = Average intensity difference between the two nodes after each shift

r = Row position of the node centre inside the search window

c = Column position of the node centre inside the search window

m = Step taken to shift the node inside the matching matrix

Normalized correlation (eq. *(1)*, chapter 2) was also applied on the same data as the two methods mentioned above for comparison to determine which method provides the best match. The results obtained with eq. *(41)*, eq. *(42)* and eq. *(1)* respectively were then stored in the matching matrix.

The matching matrix had the same dimensions as the search window (same shape and size), and thus each time the node shifted to a new position in the search window, a single value (the average intensity difference between the two nodes) was saved in the corresponding position in the matching matrix.

When the sum of absolute differences or the sum of squared differences was applied, the smallest difference in the matching matrix indicated the best match. With normalized correlation the highest correlation value indicated the best match. The position of this best match in the matching matrix indicated the new row- and column position of the best matching right node position. The right node was redrawn accordingly (figure 6.4 – here the node is drawn as a black square to

increase its visibility). Appendix A shows a diagrammatic illustration of the matching process for clarification purposes.

(a) (b) (c)

Figure 6.4: Matching a node: (a) Left image with marked node. (b) Right image with copied node. (c) Right image with matched node

6.1.4) Applying Cross-Correlation

After the nodes were matched, cross-correlation (refer to paragraph 2.3.4 and eq. *(4)*) was used to attempt an improved and more accurate match. The matched points could be used as an initial guess of the corresponding points. An 11-by-11 neighbourhood around each node centre was defined in the right image and a similar region also defined around the node centre in the left image. The correlation between the values at each pixel in these defined regions and neighbourhoods was then calculated, and the position of maximum similarity (highest correlation value) of gray levels was searched for. This was then used as the optimal position of the node. Nodes were moved up to 4 pixels only, based on the results of the cross-correlation. If some of the nodes couldn't be correlated, their position values were returned unmodified.

6.1.5) Determining the Matching Accuracy

In order to determine how accurate the matching results were, tests were performed on a collection of image pairs obtained from the Sony DKC-FP3 Digital Still cameras,

and on the image pairs obtained from the Digital Smart Cameras developed by EDH. The image pair collections used were:

- Five different image pairs obtained from the Sony cameras.
- Four image pairs that were obtained with the Digital Smart Cameras with the use of an infrared flash.
- Two image pairs that were obtained with the Digital Smart Cameras with the use of an infrared flash and texture projection.
- Five image pairs that were obtained with the Digital Smart Cameras with texture projection and without the infrared flash.

Tests were performed on these image pairs by marking 19 nodes on specific features of the left image (figure 6.5(a)), matching these nodes on the right image, displaying the results and determining the accuracy. The accuracy was determined visually with the help of four figures that showed the results of matching a stereo image pair, as displayed in figure 6.5. Figure 6.5(a) displays the left image with the marked nodes, figure 6.5(b) displays the right image with the copied nodes and figure 6.5(c) displays the right image with the matched nodes. Figure 6.5(d) is the right image with the nodes manually marked on the same features that were originally marked on the left image. The four figures can now be compared and the accuracy is determined by comparing figures 6.5(a) and 6.5(c), as well as figures 6.5(c) and 6.5(d) to see which nodes matched correctly and which didn't. In other words, comparisons were made with manually marked nodes, based on visual inspection.

The correctly matched nodes were counted for each image pair and the total out of 19 nodes matched were recorded. The average number of matched nodes for all the image pairs used in the set was determined and converted to a percentage. These tests were then repeated for accuracy.

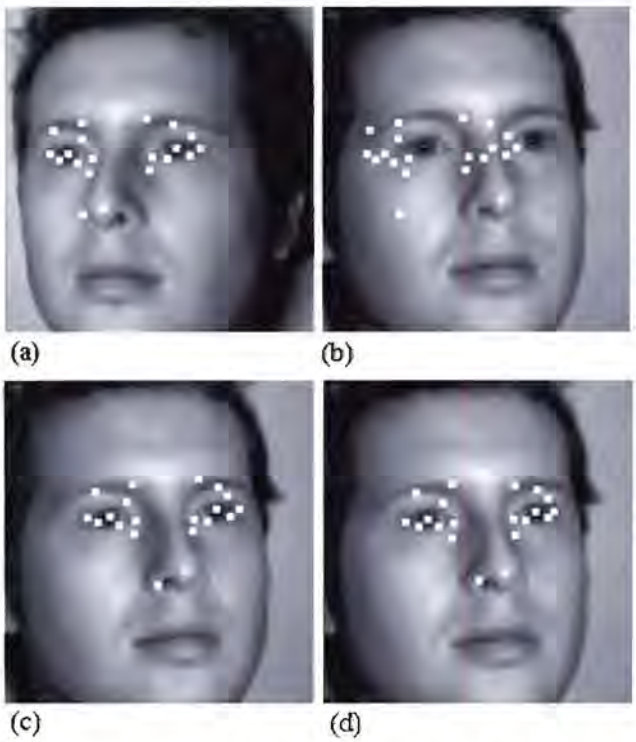


Figure 6.5: Comparing nodes on infrared images to determine matching accuracy: (a) Left image with marked nodes. (b) Right image with copied nodes. (c) Right image with matched nodes. (d) Right image with manually marked nodes for comparison

6.1.6) Statistical Comparison

A statistical comparison was performed to further determine the efficiency of the matching algorithm. A collection of image pairs of children's eyes was provided (Meintjies *et al.*, 2002), which were cropped from the images obtained with the Sony DKC-FP3 Digital Still cameras. A dataset of manually recorded 2D coordinates of six points covering the eyes on these images was also provided (Douglas *et al.*, 2003). These points (see figure 6.6) had been marked manually to obtain measurements of the palpebral fissure lengths (PFL), inner canthal distances (ICD) and the interpupillary distances (IPD) between the eyes.

Figure 6.6: Illustration of the six marked points around the eyes

A comparison study was performed between the coordinates obtained manually and coordinates obtained with the matching technique to determine the accuracy of the matching technique. The coordinates from the available dataset of manually obtained coordinates were used to create nodes on the left image. These nodes were then matched, using the algorithms on the second image of the image pair (right image). The new coordinates of the matched nodes were recorded and compared to the manual coordinates. A statistical comparison of manual and matched *x*- and *y*-coordinates was performed on 48 image pairs. A search window with size of 25-by-15 pixels was used. The tests were performed by matching nodes with size of 9-by-9 pixels and 11-by-11 pixels, and image enhancement was used to improve the matching.

Another collection of cropped image pairs of children's mouths (also cropped from the images obtained with the Sony DKC-FP3 Digital Still cameras) was created to determine the matching efficiency around the mouth. A dataset of 2D coordinates of four points manually marked around the mouth was also created. Four nodes were matched around the mouth (figure 6.7) and the matched coordinates were compared to the manually marked 2D coordinates.

Figure 6.7: Illustration of the four marked points around the mouth

A statistical comparison study was performed between the manually marked coordinates and the matched coordinates around the mouth. This was done in the same way as with the nodes around the eyes, using 48 image pairs. A 25-by-15 search window was once again used in which to search for a match, and the tests were performed by matching nodes with size of 9-by-9 pixels and 11-by-11 pixels.

6.1.7) Ellipse Fitting to Measure Upper Lip Circularity

Astley *et al.* (2002) used image analysis software to measure the magnitude of upper lip thinness from digital images. A frontal facial image is presented on a computer monitor and upper lip thinness is measured by outlining the upper lip with a computer mouse to generate the quantitative measure of thinness (circularity). It is then ranked on the 5-point Likert scale depicted on the Lip-Philtrum Guide by using circularity as a guide, where the circularity of the upper lip on the Lip-Philtrum Guide is: Rank 5 = 178, Rank 4 = 85, Rank 3 = 65, Rank 2 = 50, and Rank 1 = 35. For example, if the vermilion border of the upper lip is thin (circularity $\geq$ 75), then it can contribute to positive screening of FAS.

As an alternative to manually outlining the upper lip, fitting a semi-ellipse to points selected on the upper lip could approximate the shape of the upper lip and also save time. Circularity measurements obtained from the semi-ellipse would differ from those obtained using the Astley *et al.* (2002) method, and therefore a new scale for ranking circularity would have to be found.

An ellipse is commonly defined by its centre, the length of its major *(a)* and minor *(b)* axis and its angle of orientation *(θ)*, as illustrated in figure 6.8.

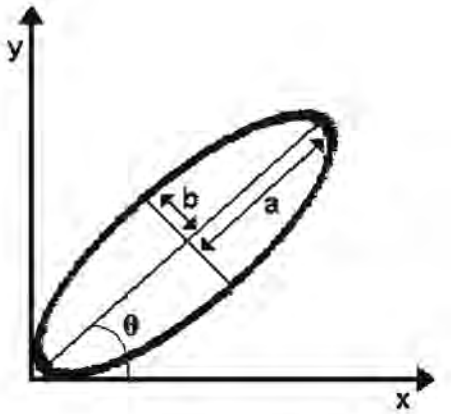


Figure 6.8: Illustration of an ellipse

Circularity is used as the continuous measure of upper lip thinness, which can be used in diagnosing FAS. It increases as the ellipse becomes thinner and varies in values between 12.80 and infinity. Circularity is defined in terms of the perimeter and area (Astley *et al.*, 2002), as shown in eq. *(44)*.

$$Circularity = \frac{(Perimeter)^2}{Area} \qquad (44)$$

The area of an ellipse is obtained with eq. *(45)* and the perimeter of an ellipse with eq. *(46)* (Spiegel, 1968):

$$Area = \pi ab \qquad (45)$$

$$Perimeter = 2\pi\sqrt{\left(\frac{a^2 + b^2}{2}\right)} \qquad (46)$$

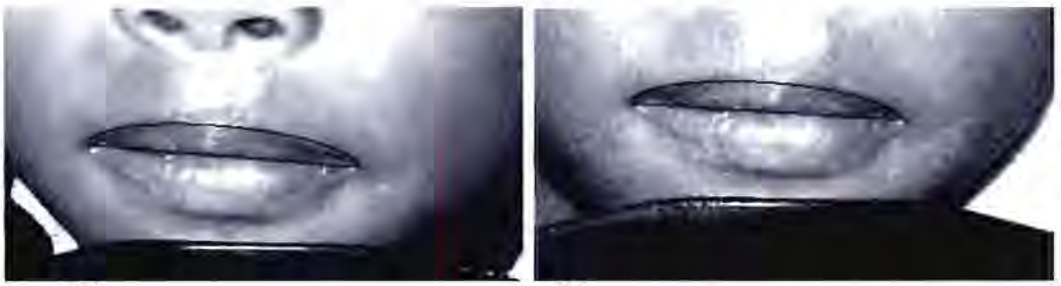
Where:

a = Length of major axis

b = Length of minor axis

Four nodes marked around the upper lip were used to fit a semi-ellipse around the upper lip. The semi-ellipse consisted of half an ellipse closed with a straight line at its bottom (figure 6.9), and was used to obtain approximate measurements of the

circularity of the upper lip (i.e. to determine the upper lip thinness). For example, the circularity of the semi-ellipse in figure 6.9(a) is 52.87 and in figure 6.9(b) is 54.22.



(a) (b)

Figure 6.9: A semi-ellipse fitted to the upper lip on an image pair after manual marking of nodes (for illustration, as no frontal photos were available): (a) Left image with fitted semi-ellipse. (b) Right image with fitted semi-ellipse

Since half an ellipse together with a straight line was used for fitting around the upper lip, equations *(45)-(46)* were modified before use. This entailed using half the area and also half the perimeter plus the length of the line, where the line's length is double the length of the major axis (refer to figure 6.8). Equations *(45)-(46)* were rewritten and used as:

$$Area = \frac{\pi ab}{2} \qquad (47)$$

$$Perimeter = \pi \sqrt{\left(\frac{a^2 + b^2}{2} \right)} + 2a \qquad (48)$$

In equations *(47)-(48)* the variables are the same as those defined for equations *(45)-(46)*. The semi-ellipse was fitted onto three-dimensional coordinates obtained from the marked and the matched nodes around the upper lip to determine the efficiency of the matching algorithm. In each case the circularity of the semi-ellipse was determined and compared. This was performed on 48 image pairs. The semi-ellipse has a shape that can be used to easily approximate the circularity of the vermilion border of the upper lip, and was therefore used to measure the upper lip circularity.

Four points were selected around the upper lip in the order as indicated in figure 6.10. The second point was used as the centre of the ellipse, and the distance between points 1 and 2 represented the length of the minor axis of the ellipse (*b* in figure 6.8). With the selected points, a semi-ellipse could be fitted to determine the circularity of the upper lip. Half of the distance between points 3 and 4 represented the length of the major axis (*a* in figure 6.8).

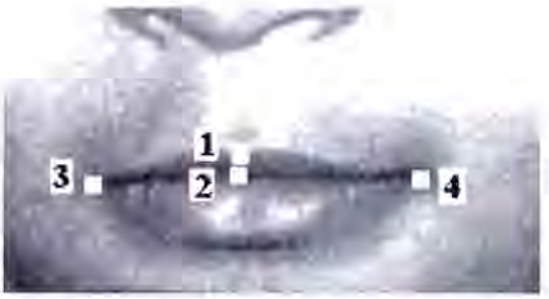


Figure 6.10: Illustration of four marked points around the upper lip for semi-ellipse fitting: Left image with selected four points

Three-dimensional coordinates of the selected points around the upper lip were obtained using the DLT (refer to paragraph 3.1.2). The sets of 3D coordinates were used to determine the centre coordinates, the tilt angle, the minor and major axis lengths of the ellipse and ultimately the ellipse equation. This was then used to determine the upper lip circularity.

6.2) The Matching Results: Accuracy and Efficiency

All the results obtained from the matching process are discussed through the rest of the chapter and the different methods of obtaining these results are also compared.

6.2.1) Accuracy of the Matching Process

During the matching of the nodes, different approaches were compared to obtain the best matching results. These included using nodes of different sizes. Nodes consisting of a square of 11-by-11 pixels, 9-by-9 pixels, 7-by-7 pixels and 5-by-5 pixels were matched on the same image pairs and compared. It was found that

smaller nodes give less accurate matching results. The most accurate matching results were obtained with the 9-by-9 nodes on the 512-by-492 pixel resolution grayscale images. For the 1024-by-1344 pixel resolution images, the best matching results were obtained when using 11-by-11 nodes. Therefore it was decided to use nodes the size of a 9-by-9 square for the smaller resolution images and a node size of 11-by-11 pixels for the higher resolution images.

Three different methods to obtain the average intensity difference between the left node and right node pixels were also tested on the same data in order to see which method provided the most accurate results. The methods tested included the sum of absolute differences, the sum of squared differences and normalized correlation. All three methods provided very similar results, but the sum of absolute differences provided slightly more accurate matching results (i.e. a few nodes were slightly more accurately matched). Since the sum of absolute differences proved to be the more effective equation in this case, it was used in the matching process.

Cross-correlation (as discussed in paragraph 6.1.4) was applied in an attempt to further improve the matching results (i.e. after the nodes were matched on the right image). This hardly influenced the node positions, and when it did, nodes were slightly less well matched. It was therefore decided not to include cross-correlation in the matching process.

6.2.2) Using Images Taken with the Sony DKC-FP3 Digital Still Cameras

Tests were performed on 5 different image pairs, and repeated for accuracy. The following tables give a description of the applied image enhancement and also the percentage matching accuracy. Although various enhancement techniques were applied and tested, only the most accurate results are displayed in the tables. The image pairs used have a resolution of 1024-by-1344 pixels. A search window with dimensions 40-by-60 (i.e. columns-by-rows) was used, and a node size of 11-by-11 pixels. The results are shown in table 6.1.

The best average matching result (81.06%) was obtained with test #6, table 6.1, and is illustrated in figure 6.11. In this test, histogram equalization was applied to the

image pair, after which Canny edge detection was applied. The resulting set of detected edges (edges were marked with an intensity of 0 (black)) was then copied on to another image to which had been applied histogram equalization and contrast stretching, and the matching was performed.

Test #	Image enhancement techniques applied	Accuracy % (min, max)	Accuracy % (mean)
1	No image enhancement techniques applied	35.80 – 39.98	37.89
2	Only histogram equalization (*histeq.)*	77.89 – 80.01	78.95
3	First applying *histeq* and then contrast stretching	77.89 – 78.95	78.42
4	Apply Prewitt edge detection on image after *histeq*. Create new copy of image by first applying *histeq*, then contrast stretching and then copying the set of detected edges (with intensity of 255 - white) to this image	71.58 – 73.68	72.11
5	Apply Canny edge detection on image after *histeq*. Create new copy of image by first applying *histeq*, then contrast stretching and then copying set of detected edges (white) to this image	80.01 – 81.05	80.53
6	Apply Canny edge detection on image after *histeq*. Create new copy of image by first applying *histeq*, then contrast stretching and then copying set of detected edges (with intensity of 0 - black) to this image	80.01 – 82.11	81.06

Table 6.1: Tests and results from 1024-by-1344 resolution images (using 5 images pairs, matching 19 nodes)

(a) (b)

Figure 6.11: Matching accuracy obtained with test #6, table 6.1: (a) Left image with marked nodes. (b) Right image with matched nodes

For both the two most accurate tests (test #5 and #6), the Canny thresholds applied were a maximum threshold of 0.2 and a minimum of 0.08, while a sigma value of 2.5 was applied. In each case, contrast stretching was applied with mapping values between $(0.3,0)$ and $(1,1)$ (referring to (r_1,s_1) and (r_2,s_2) in paragraph 2.4.1). Figure 6.12 shows the enhancement techniques applied during tests #5 and #6 from table 6.1.

(a) (b)

Figure 6.12: The two most effective enhancement techniques: (a) Enhancement applied for test #5, table 6.1. (b) Enhancement applied for test #6, table 6.1

When tests were performed on these images with no enhancement techniques applied (test #1), an average matching accuracy of 37.89 % was obtained, which

shows that image enhancement through histogram equalization, contrast stretching and edge detection is necessary to obtain better matching results.

6.2.3) Using Images Taken with the Digital Smart Cameras

The first collection of image pairs obtained with the Digital Smart Cameras consists of four image pairs photographed with the use of an infrared flash. Tests were performed on these 4 image pairs in exactly the same way as with the images obtained with the Sony Digital Still cameras (paragraph 6.2.2), and the tests were repeated for accuracy. It should however be noted that the use of infrared light enhances visibility of features in the images and plays an important role in obtaining similar brightness in image pairs, therefore reducing the need to apply image enhancement techniques. For example, histogram equalization was not necessary since the infrared flash gave similar brightness in the image pairs. Only contrast stretching was applied followed by edge detection. The following tables give a description of the image enhancement applied and also the percentage matching accuracy. Once again various enhancement techniques were applied and tested, but only the most accurate results are displayed in the tables. A search window of size 115-by-15 (columns-by-rows) was used for these images and a node size of 9-by-9. The image pairs had a resolution of 512-by-492 pixels.

The best matching result (78.3%) was obtained with test #2, table 6.2. In this test, only contrast stretching was applied to the image pair (figure 6.13). Contrast stretching was once again applied with mapping values between $(0.3,0)$ and $(1,1)$ (referring to (r_1,s_1) and (r_2,s_2) in paragraph 2.4.1).

(a) (b)

Figure 6.13: Applying contrast stretching for test #2, table 6.2: (a) Original facial image obtained with infrared flash. (b) Enhanced facial image

Test #	Image enhancement techniques applied	Accuracy % (min, max)	Accuracy % (mean)
1	No image enhancement techniques applied	76.33 – 77.63	76.98
2	Only contrast stretching	77.63 – 78.95	78.30
3	Apply Prewitt edge detection on image after contrast stretching. Create new copy of image by applying contrast stretching and then copying white set of detected edges to image	67.11 – 68.42	67.77
4	Apply Canny edge detection on image after contrast stretching. Create new copy of image by applying contrast stretching and then copying white set of detected edges to image	68.42 – 76.32	72.37
5	Apply Prewitt edge detection on image after contrast stretching. Create new copy of image by applying contrast stretching and then copying black set of detected edges to image	63.16 – 67.11	65.14
6	Apply Canny edge detection on image after contrast stretching. Create new copy of image by applying contrast stretching and then copying black set of detected edges to image	76.32 – 77.63	76.98

Table 6.2: Tests and results on infrared images (using 4 images pairs, matching 19 nodes)

Figure 6.14 shows the results of matching a stereo image pair taken with an infrared flash.

(a) (b)

Figure 6.14: Matching accuracy obtained using an infrared flash with test #2, table 6.2: (a) Left image with marked nodes. (b) Right image with matched nodes

Two image pairs were taken with the use of an infrared flash and texture projection, and five image pairs with applied texture projection and no infrared flash. Tests were performed on all these image pairs in exactly the same way as with the previous image pairs, and repeated for accuracy. Enhancement techniques applied were exactly the same as the enhancement techniques performed on the other 512-by-492 resolution images and the same node size was used for matching, but histogram equalization was additionally applied to the five image pairs with texture projection and no infrared flash. A search window size with dimensions 40-by-40 (columns-by-rows) was used. Tables 6.3 and 6.4 give a description of the most effective image enhancement and also the percentage matching accuracy.

The best matching result (82.90%) was obtained with test #5, table 6.3. In this test contrast stretching was applied with mapping values between $(0.3,0)$ and $(1,1)$, after which Canny edge detection was applied. The resulting set of detected edges (edges were marked with an intensity of 0 (black)) was copied on to the image, and the matching was performed. The Canny thresholds applied were a maximum threshold of 0.2 and a minimum of 0.08, while a sigma value of 2.5 was applied. In test #2, only contrast stretching was applied with mapping values between $(0,0.2)$ and $(1,0.8)$, while test #3 only used contrast stretching with mapping values between

$(0.3,0)$ and $(1,1)$, (referring to (r_1, s_1) and (r_2, s_2) in paragraph 2.4.1). Figure 6.15 shows an example of the facial images and enhancement used in these tests.

Test #	Image enhancement techniques applied	Accuracy % (min, max)	Accuracy % (mean)
1	No image enhancement techniques applied	75.44 – 78.54	76.99
2	Only contrast stretching with low mapping values	78.95 – 81.58	80.27
3	Only contrast stretching with high mapping values	80.71 – 84.21	82.46
4	Apply Canny edge detection on image after contrast stretching. Create new copy of image by applying contrast stretching and then copying white set of detected edges to image	76.32 – 81.58	78.95
5	Apply Canny edge detection on image after contrast stretching. Create new copy of image by applying contrast stretching and then copying black set of detected edges to image	81.58 – 84.22	82.90

Table 6.3: Tests and results on images with infrared flash and texture projection (using 2 images pairs, matching 19 nodes)

(a) (b) (c)

Figure 6.15: Facial images with contrast stretching: (a) Original image – test #1, table 6.3. (b) Contrast stretching with low mapping values – test #2. (c) Contrast stretching with high mapping values – test #3

Table 6.4 shows the results from the 5 image pairs used with texture projection and no infrared flash. Because the infrared flash wasn't used, images without histogram

equalization applied were too dark (almost completely black – images were obtained in a dark room), and therefore no tests were performed on the images without enhancement applied.

In table 6.4 the best matching accuracy is with test #4. In this test histogram equalization was applied to the image pair, after which contrast stretching was applied. Canny edge detection was then applied and the resulting set of detected edges (edges were marked with an intensity of 255 (white)) was then copied on to the histogram equalized and contrast stretched image. The matching was then performed on that image. For both the two most accurate tests (test #4 and #5), the Canny thresholds applied were a maximum threshold of 0.2 and a minimum of 0.08, while a sigma value of 2.5 was applied. In the tests shown in table 6.4, mapping values for contrast stretching were the same as for table 6.3. From these results it is clear that texture projection plays a big role in improving the matching accuracy.

Test #	Image enhancement techniques applied	Accuracy % (min, max)	Accuracy % (mean)
1	Only histogram equalization	78.95 – 83.16	81.06
2	First applying histogram equalization and then contrast stretching with low mapping values	78.95 – 84.21	81.58
3	First applying histogram equalization and then contrast stretching with high mapping values	78.95 – 81.10	80.03
4	Apply Canny edge detection on image after histogram equalization and contrast stretching. Create new copy of image by applying contrast stretching and then copying white set of detected edges to image	83.16 – 84.21	83.69
5	Apply Canny edge detection on image after histogram equalization and contrast stretching. Create new copy of image by applying contrast stretching and then copying black set of detected edges to image	81.05 – 83.16	82.10

Table 6.4: Tests and results on images with only texture projection and no infrared flash (using 5 images pairs, matching 19 nodes)

Figure 6.16 shows an example of the matched nodes (the nodes are drawn as black squares to increase their visibility).

Figure 6.16: Matching accuracy obtained with test #4, table 6.4: (a) Left image with marked nodes. (b) Right image with matched nodes

6.2.4) Results: Statistical Comparison and Ellipse Fitting

For the statistical comparison performed on the 48 image pairs obtained with the Sony DKC-FP3 Digital Still Cameras, the most successful image enhancement techniques were applied (refer to test #6 in table 6.1).

The results obtained from comparing coordinates around the eyes of the right images obtained manually and via matching respectively, are shown in table 6.5 and illustrated in figure 6.17.

An approximate conversion from pixels to mm for the used 1024-by-1344 images is 1 mm ≡ 4.34 pixels. Typical pixel differences between the corresponding nodes in figure 6.17(c) and (d) vary from 0 to 5 pixels (0 – 1.15 mm). The maximum difference between the nodes in figure 6.17 is a difference of 5 pixels in the x-direction (horizontally) for the node on the pupil of the left eye (node #5).

Node pos. (fig. 1)		Mean abs. differences between coordinates	Standard deviation	Max. absolute difference	P-value (Paired Student's t-test)	Statistical significance (Confidence level of 5%)	Mean abs. differences between coordinates (mm)	Max. absolute difference (mm)
1	X	3.01	3.46	13.13	0.46	No	0.69	3.03
	Y	1.72	1.55	5.74	0.79	No	0.40	0.36
2	X	3.36	3.08	13.38	0.68×10^{-3}	Yes	0.77	3.08
	Y	1.31	2.04	7.45	0.29	No	0.30	1.72
3	X	3.26	3.31	13.36	0.99×10^{-4}	Yes	0.75	3.08
	Y	2.23	1.89	7.31	0.71	No	0.51	1.68
4	X	2.60	3.22	13.41	0.29	No	0.60	3.09
	Y	2.20	1.84	7.42	0.71	No	0.51	1.71
5	X	3.28	1.72	6.97	0.25×10^{-9}	Yes	0.76	1.61
	Y	0.97	0.95	6.09	0.49	No	0.22	1.40
6	X	2.74	2.39	12.20	0.15	No	0.63	2.81
	Y	2.10	2.39	7.11	0.08	No	0.48	1.64

Table 6.5: Results obtained for comparison of manually marked coordinates and matched coordinates around the eyes (all measured in pixel values except for last two columns)

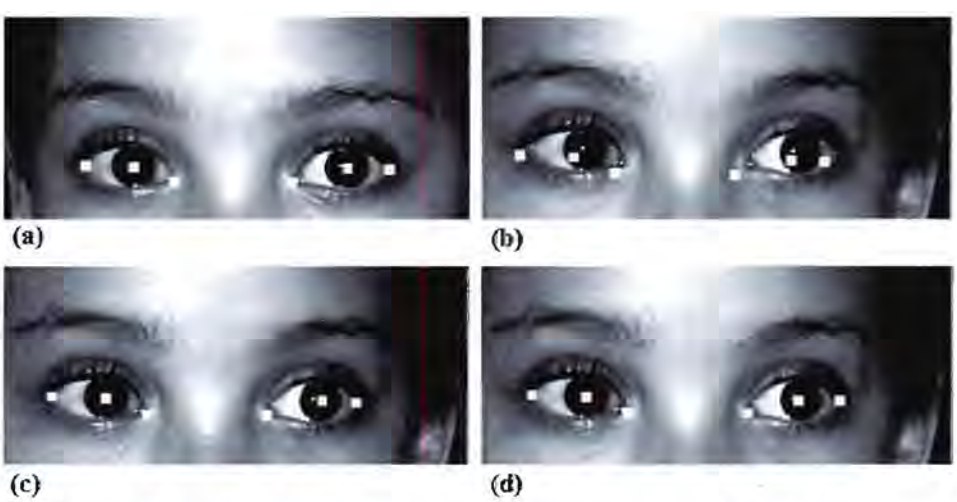


Figure 6.17(a): Left image with manually marked nodes. (b) Right image with copied nodes before matching. (c) Right image with manually marked nodes. (d) Right image with matched nodes

The results obtained from comparing coordinates around the mouth (refer to figure 6.7) of the right images obtained manually and via matching respectively, are shown in table 6.6.

Node pos. (fig. 2)		Mean abs. differences between coordinates	Standard deviation	Max. absolute difference	P-value (Paired Student's t-test)	Statistical significance (Confidence level of 5%)	Mean abs. differences between coordinates (mm)	Max. absolute difference (mm)
1	x	3.67	3.15	13.00	0.23	No	0.85	3.00
	y	1.54	1.44	7.00	0.17	No	0.35	1.61
2	x	6.63	3.94	13.00	0.56×10^{-2}	Yes	1.52	3.00
	y	1.73	1.40	7.00	0.65	No	0.40	1.61
3	x	3.38	3.40	13.00	0.02	Yes	0.78	3.00
	y	2.10	1.61	7.00	0.42	No	0.48	1.61
4	x	2.79	1.98	8.00	0.45	No	0.64	1.84
	y	1.69	1.11	4.00	0.83	No	0.39	0.92

Table 6.6: Results obtained for comparison of manually marked coordinates and matched coordinates around the mouth (all measured in pixel values except for last two columns)

The circularity of the upper lip was determined from a semi-ellipse created with the use of 3D coordinates obtained with the DLT. Three-dimensional coordinates were obtained from manually marked nodes around the upper lip in the left image together with manually marked and matched nodes in the right image for semi-ellipse fitting. The circularity results obtained were compared as shown in table 6.7.

From the Paired Student's t-test performed between the marked circularity and the matched circularity, a P-value of 0.99 was obtained. This indicates that the difference in circularity is not statistically significant at the 5% level.

Circularity	Manually marked nodes	Matched nodes
Maximum	80.27	80.27
Minimum	28.12	26.85
Mean	42.95	42.96
Mean absolute difference between circularities obtained from marked and matched nodes	2.46	
Standard deviation	3.51	
P-value (Paired Student's t-test)	0.99	

Table 6.7: Results obtained for comparison of circularity obtained from manually marked coordinates and matched coordinates around the mouth (using 48 image pairs)

6.3) Runtime for the Matching Process

The runtime for the matching tests depended mainly on the resolution of the images. It took 25-35 seconds to load a 1024-by-1344 resolution image pair, while it took 5-10 second to load a 512-by-492 resolution image pair in order to mark the nodes on the left image. Once the nodes were marked on the left facial image, it took 30-45 seconds to obtain the matching results of 19 nodes with 1024-by-1344 resolution images, while it took 10-25 seconds with the 512-by-492 resolution images. Whether 5 nodes or 50 nodes were matched didn't affect the runtime too much.

6.4) Summary and Discussion

Tests were performed on different image pairs to determine the matching accuracy of the matching algorithm. Different image enhancement techniques were applied to improve the accuracy, which included feature enhancement, edge detection, texture projection during image acquisition, and the use of an infrared flash.

The best matching result obtained with the 1024-by-1344 resolution images is 81.06%. The best matching accuracy on the 512-by-492 resolution images with

infrared flash and texture projection is 82.90%, while an accuracy of 83.69% is obtained with texture projection only. The use of an infrared flash reduces the need for other image enhancement techniques, but should not be used together with texture projection during image acquisition, since the flash reduces the intensity and therefore the effectiveness of the texture projection. Although the use of an infrared flash and texture projection both contribute to the matching process and its accuracy, texture projection proves to be more effective. It must be noted that different texture patterns were used during the testing, and those that had less repetition in their texture pattern resulted in more accurate matching results. The resolution of an image will also affect the matching, and a higher resolution image has finer detail, which increases the chances of obtaining an accurate match. An even higher matching accuracy will be obtained if texture projection is applied to an 1024-by-1344 resolution image. A higher image resolution will however result in a longer runtime.

From the statistical comparison, the matching results around the eyes and mouth have a good accuracy with mean absolute differences in landmark coordinates less than 1mm for the eyes and less than 2 mm for the mouth. The eye differences in all the y-coordinates are not statistically significant at the 5% level (table 6.5), but half of the x-coordinate differences are significant. These are the x-coordinates of the two pupils and of the right medial canthus (inner corner of the right eye). This can be related to the white reflection on the centre of the eye, which changes position in the image pair, affecting the matching. Some reflection also occurs around the inner canthus of the eye, which can affect the matching accuracy. Reflection off the eyes can be reduced with the use of an infrared flash, therefore reducing the risk of an inaccurate match.

Regarding the matching around the mouth, the differences in all the y-coordinates are not statistically significant at the 5% level (table 6.6). However, two of the x-coordinate differences are statistically significant - those obtained beneath the mouth and at the right corner of the mouth (points 2 and 3 in figure 6.7). This can be related to the fact that certain individuals might have a similar skin complexion to the colour of the lips and that this similarity in pixel intensity might lead to a false match. The matching accuracy around the eyes and mouth can also be affected by manual errors, i.e. inaccuracies that occurred when marking the corresponding nodes manually on the right image.

The circularity obtained from the manually marked nodes and from the matched nodes shows a satisfactory correspondence, and the difference in circularity is not statistically significant at the 5% level. A mean absolute difference of 2.46 was obtained between the circularity obtained from the marked coordinates and the circularity obtained through matching, and a standard deviation of 3.51 was obtained. This further indicates that good matching results can be obtained with the stereo matching algorithm. It must be noted that fitting the semi-ellipse to the upper lip gives only an approximation of the upper lip circularity, due to its shape limitations.

The results are discussed in more detail in chapter 8.

Chapter 7

Three-Dimensional Reconstruction

The final step in the image processing software is the three-dimensional reconstruction of facial features from a stereo image pair. Information from the matched corresponding points is used to obtain 3D coordinates through mathematical techniques adapted from the literature. These coordinates are then used for three-dimensional reconstruction of the features enclosed by the matched points. The methods of determining three-dimensional coordinates and using them for three-dimensional reconstruction are looked at in this chapter, as well as the 3D results obtained.

7.1) Methods: Obtaining the 3D Coordinates

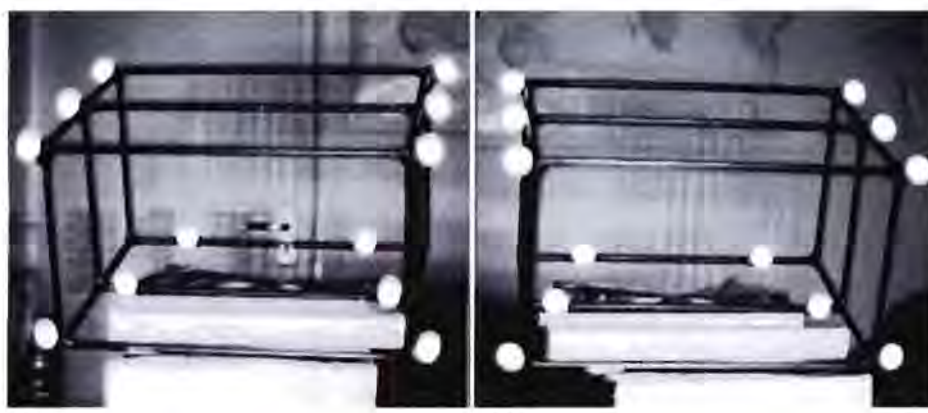
Camera calibration was used to obtain the transformation from 2D image space to 3D object space. This was achieved with the help of the DLT (Abdel-Aziz & Karara, 1971), which transforms the 2D coordinates of a visible, corresponding point on each of the two images into the 3D coordinates of that point.

It was decided to implement the DLT instead of bundle adjustment since the DLT is simple to apply and it requires no initial approximations for the unknowns (refer back to paragraph 3.1.2). A solution can still be obtained where bundle adjustment might fail to obtain an accurate solution due to possible insufficient initial approximations. Bundle adjustment also takes longer and requires more computing power, since it involves large data sets and the computation of the inverses of large matrices.

There are twelve transformation parameters required for the transformation from 3D space to 2D space. These parameters of the DLT were solved using a calibration frame (which requires a minimum of six control points), and least squares adjustment. The calibration frame used with the Digital Smart Cameras consists of a grid of 6 mm steel bars enclosing a volume of 200-by-250-by-150 mm. Attached to

the frame at various points are 12 markers of which the 3D coordinates are accurately known.

The Digital Smart Cameras were used to obtain an image pair of the calibration frame. The DLT was solved for each camera, and the two sets of transformation parameters were obtained (i.e. 12 left parameters and 12 right parameters). With these parameters, 3D coordinates (X,Y,Z) of any point visible in both the left and the right image, with image coordinates (x_l, y_l) and (x_r, y_r), respectively, could then be computed.



(a) (b)

Figure 7.1: Image pair of calibration frame: (a) Left image and (b) Right image

After the image pair of the frame was obtained (figure 7.1), the frame was removed and facial images were taken. The face was placed in the same position as the frame, with the chin resting on the same point as the centre of the bottom part of the frame. Nodes were matched on these facial image pairs, thus obtaining the two sets of 2D coordinates, which were used to obtain the 3D coordinates.

7.2) Methods: Three-Dimensional Surface Reconstruction

The 3D coordinates obtained using the DLT were used for three-dimensional reconstruction of the features enclosed by the matched points. This was achieved by applying Delaunay triangulation to create a mesh that connected the 3D coordinates. Delaunay triangulation (refer to paragraph 3.3) was applied for facial reconstruction

instead of using Voronoi diagrams, since it gave a clearer and better visual appearance of the three-dimensional surface. It was also chosen above surface skinning through splines (e.g. NURBS curves – refer to paragraph 3.4) for 3D reconstruction, since skinning requires a great deal of computer memory and a large number of control points (or nodes).

Delaunay triangulation was applied by determining which of the 3D coordinate points should be connected to each other and then connecting the relevant points (midpoints of the nodes) with straight lines, creating a 3D mesh. These lines were created by using the line equation $y = mx + c$, where m is the slope of the line and c is the intercept on the y-axis (James *et al.*, 2000). This mesh represented the reconstructed three-dimensional surface. This surface was too sparse for facial feature representation, and was therefore interpolated by applying bilinear interpolation to produce a dense three-dimensional surface. The interpolation was performed on single pixel intervals between the known 3D points. Bigger interpolation values will result in less surface detail. Smaller interpolation intervals (e.g. 0.25 pixel intervals) will result in a denser, smoother 3D surface, but the runtime will increase as the interval size decreases, since more data points are used.

Delaunay triangulation was also applied on each of the two images with the 2D coordinates of the matched nodes to create a 2D mesh of each image for illustrating which features were covered by the mesh and thus included in the three-dimensional reconstruction. The midpoints of the nodes were connected with straight lines of intensity 255 on each image of the facial image pair. The resulting 2D mesh on each image of the image pair differs from each other since different coordinates are used. The 3D mesh obtained from Delaunay triangulation (as described in the previous paragraph) also has a different structure since 3D coordinates are used.

7.3) Results: Obtaining 3D Coordinates

The accuracy of the DLT was determined by comparison of 3D coordinates of markers on a calibration frame obtained from the DLT with the known 3D coordinates of the markers (see table in Appendix C). Twelve prominent points were marked on an image pair of the calibration frame. These points were marked in the centres of the markers of the frame (see figure 7.2).

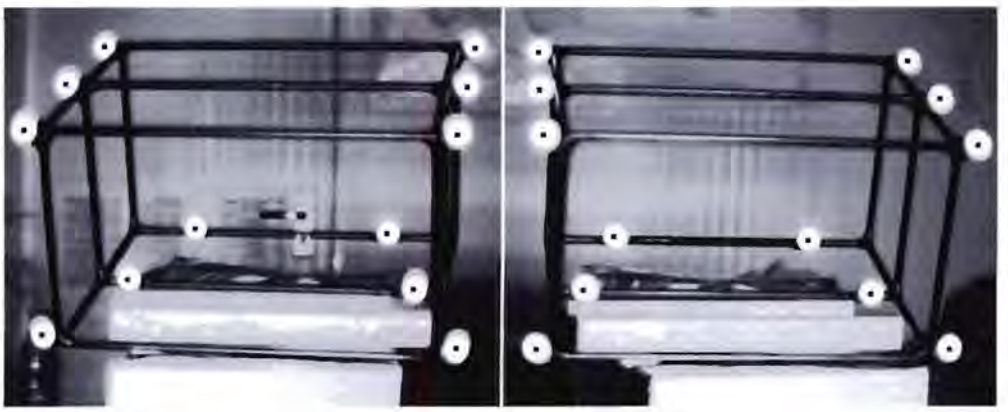


Figure 7.2: Image pair of a calibration frame with marked nodes in the centres of all the markers

The absolute difference between the obtained 3D coordinates and the known 3D coordinates were obtained in order to determine whether the DLT gives accurate 3D coordinates (table 7.1).

ΔX (mm)	ΔY (mm)	ΔZ (mm)
0.66	0.47	0.41
0.35	0.27	0.04
0.61	0.63	0.47
1.44	2.12	0.19
0.32	1.04	0.69
1.01	0.40	0.27
0.23	0.56	0.69
0.22	0.17	0.82
0.39	0.56	0.61
0.17	0.66	2.28
0.47	0.30	2.03
0.30	0.39	0.97

Table 7.1: Difference in the X-, Y- and Z-directions between 3D coordinates obtained from the DLT and known accurate 3D coordinates

From table 7.1 an average difference of 0.51 mm was obtained in the X-direction, while an average difference of 0.63 mm was obtained in the Y-direction and a 0.79 mm difference in the Z-direction. Thus an average difference of less than 1 mm occurs when using the DLT, showing that the DLT gives accurate 3D coordinates.

7.4) Results: Displaying the 3D Image

The Direct Linear Transform was applied to obtain 3D coordinates of an eye area for three-dimensional reconstruction. A collection of 14 nodes was matched (figure 7.3) and a three-dimensional mesh was obtained from these nodes through Delaunay triangulation, as shown in figure 7.4.

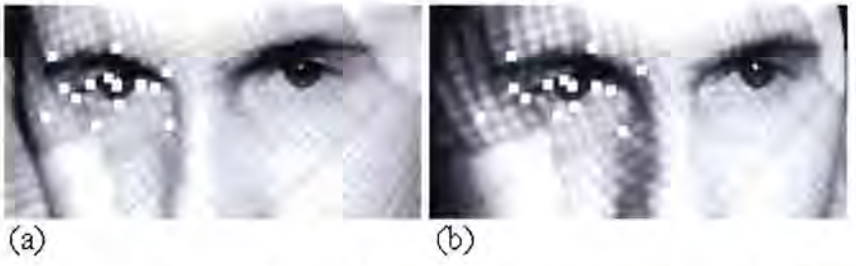


Figure 7.3: Matching 14 nodes for 3D reconstruction: (a) Left image with marked nodes. (b) Right image with matched nodes

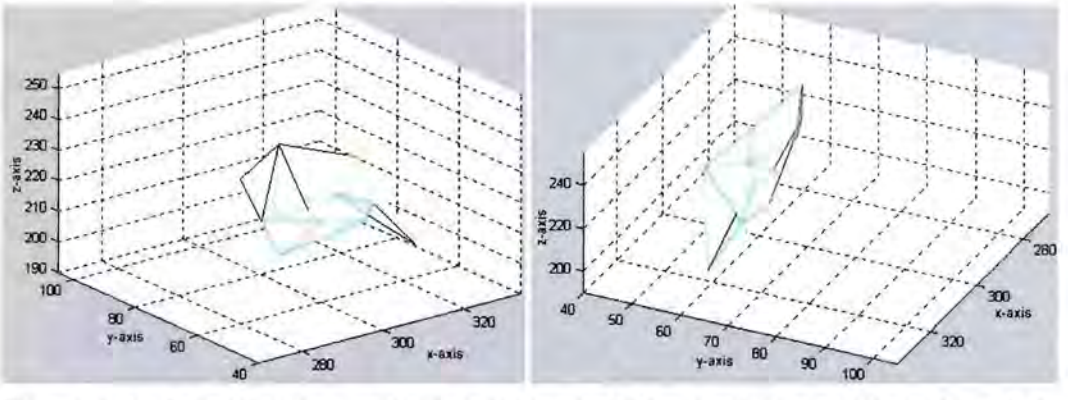


Figure 7.4: Constructing a 3D mesh through Delaunay triangulation: The mesh seen from different angles

From figure 7.4 it is clear that the mesh obtained is too sparse, leading to an unclear 3D surface. Applying bilinear interpolation to the above 3D mesh produced a dense three-dimensional surface as shown in figure 7.5.

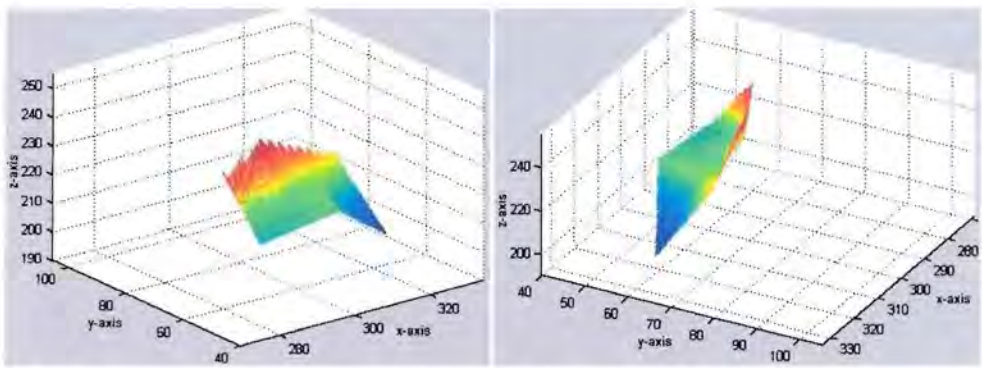


Figure 7.5: Obtaining a dense 3D surface through bilinear interpolation: The 3D surface seen from different angles

It can be seen from figures 7.4 and 7.5 that matching too few nodes results in a three-dimensional surface with too little detail. In order to obtain more detail, a large number of nodes must be matched accurately. However, if some of the nodes aren't matched correctly, the reconstructed 3D surface will be inaccurate. In order to show a three-dimensional reconstructed surface with a large number of nodes matched correctly, 155 nodes were marked manually on a left and right facial image. The results are shown in figures 7.6 to 7.9.

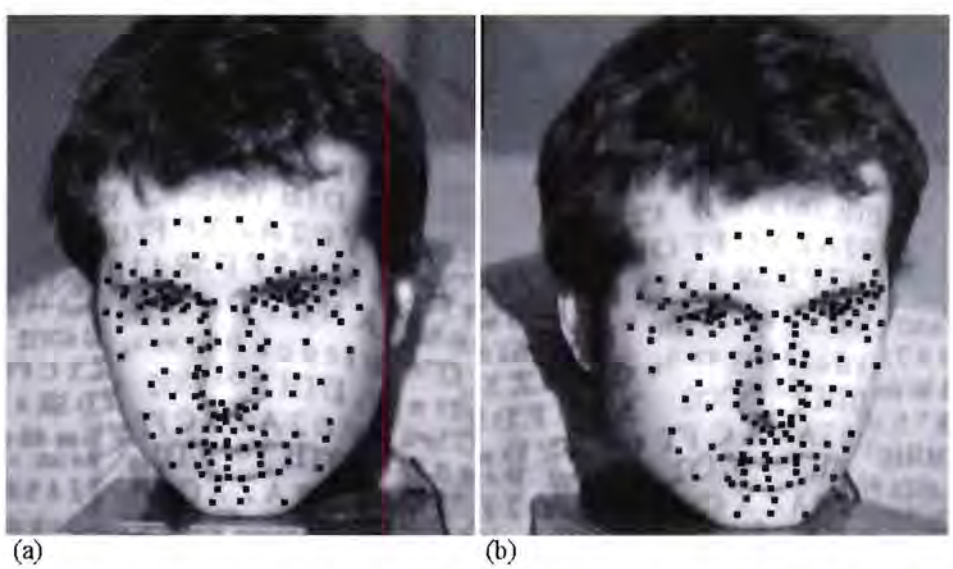
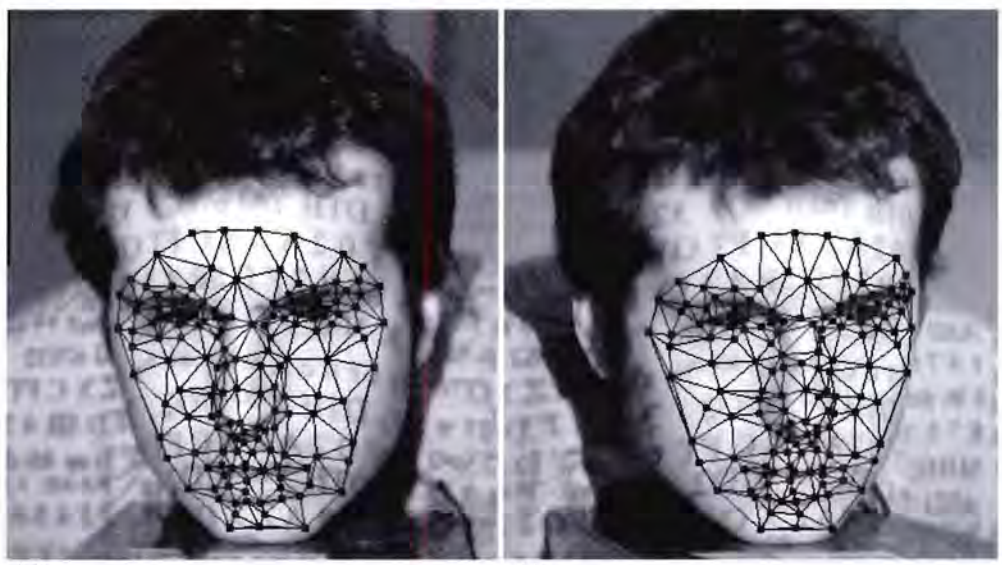


Figure 7.6: Marking 155 nodes for 3D reconstruction of the whole face: (a) Left image with marked nodes. (b) Right image with manually marked nodes

Figure 7.7 shows a 2D mesh created on each of the images through Delaunay triangulation. This indicates which features will be covered by the 3D mesh and thus included in the three-dimensional reconstruction.



(a) (b)

Figure 7.7: Constructing a 2D mesh through Delaunay triangulation: (a) Left image with 2D Delaunay mesh. (b) Right image with 2D Delaunay mesh

The DLT was applied and used the 2D coordinates of the marked nodes to determine its 3D coordinates. Figure 7.8 shows the three-dimensional reconstruction through Delaunay triangulation obtained from the 3D coordinates.

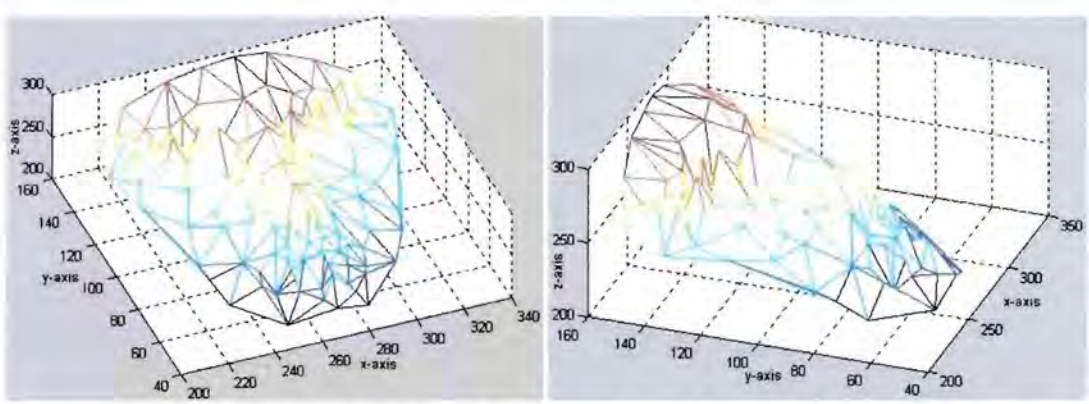


Figure 7.8: Constructing a 3D mesh to cover facial features through Delaunay triangulation

Bilinear interpolation was applied to the above 3D mesh and this produced a dense three-dimensional surface as shown in figure 7.9, giving a better three-dimensional representation of the facial features covered by the marked nodes.

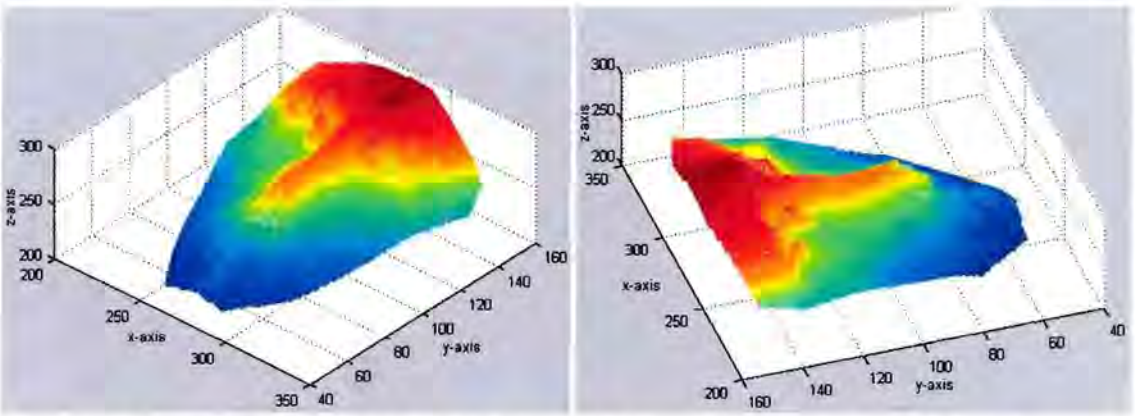


Figure 7.9: Obtaining a 3D surface from 155 marked nodes: The 3D facial surface seen from different angles

Although the facial features can be seen clearly in figure 7.9, some of the detail might be slightly inaccurate. This is due to manual errors that occurred when the nodes were marked on the image pair, i.e. inaccuracies that occurred when marking the corresponding nodes on the image pair.

7.5) Runtime to Obtain 3D Results

The time it takes to obtain a dense three-dimensional surface depends on the number of nodes used, and is also dependent on the interpolation intervals between 3D points and the size of the area to be reconstructed. If the interpolation is performed over more points to obtain a finer surface, the runtime will increase. Also, the higher the image resolution, the longer 3D reconstruction will take. Obtaining a three-dimensional surface from 512-by-492 resolution images using 14 matched nodes (e.g. figure 7.5) will take 15-25 seconds while it will take 30-35 seconds to obtain a 3D surface from 155 matched nodes (e.g. figure 7.9).

7.6) Summary and Discussion

The corresponding 2D coordinates obtained during the matching process were applied to obtain a set of 3D coordinates through camera calibration. These 3D coordinates were then applied to create a 3D mesh through Delaunay triangulation. The mesh was further interpolated, resulting in a three-dimensional surface of the matched features in an image pair.

The Direct Linear Transform was applied to obtain the three-dimensional coordinates. The accuracy of the DLT was first determined with a calibration frame, by comparing 3D coordinates obtained with the DLT with known 3D coordinates of marker centres of the frame. A good accuracy was obtained with the DLT, with an average error of less than 1 mm.

Delaunay triangulation was used to reconstruct a three-dimensional mesh from the 3D coordinates. This mesh was further interpolated using single pixel interpolation intervals, giving sufficient surface detail for successful three-dimensional surface reconstruction. From this kind of surface reconstruction it will be possible to obtain facial measurements for use in diagnosing FAS.

The three-dimensional results are highly dependent on the matching accuracy. Inaccurate matching will give incorrect 2D coordinates, and this will in turn lead to incorrect 3D coordinates and surface reconstruction. The number of nodes matched also affects the surface reconstruction – the more nodes matched, the more detailed the reconstruction will be.

Chapter 8

Discussion

Image processing software was developed to perform matching and three-dimensional reconstruction of facial features from a stereo image pair. It enables a person to identify certain points (ideally lying on distinct features) on one of the images from a selected stereo image pair (consisting of a left and a right facial image). The corresponding points in the second image are obtained through matching, where image enhancement techniques are applied to improve the matching. Information from the matched corresponding points is used to obtain 3D coordinates through camera calibration using the Direct Linear Transform (DLT). These coordinates are then applied for three-dimensional reconstruction of the features enclosed by the matched points.

The algorithm for matching features on an image pair and creating a 3D surface of these features is summarized in the following steps:

1) The user selects the number of nodes for matching and marks the node positions on a left facial image from a stereo image pair.
2) The coordinates of the node positions are recorded and the image pair is modified through image enhancement in preparation for the matching process.
3) A single small square node is created at the recorded coordinates on both images in the image pair.
4) A search window is created around the node in the right image, and the right node is shifted one pixel at a time inside the search window.
5) Comparisons between the left and right node is made for each shift, based on the pixel intensities of the pixels covered by the nodes.
6) The position of the right node with the best correspondence to the left node indicates the best match, and this position data is stored.
7) Steps (4)-(7) are repeated for the number of nodes selected until all the nodes are matched.
8) The corresponding points in the image pair are used with the DLT to obtain 3D coordinates.

9) Delaunay triangulation and bilinear interpolation are applied to these coordinates to create a 3D surface of the facial area represented by the matched nodes.

The aim of the thesis project was to develop stereo matching software for identifying and matching areas containing the same features in an image pair, without a person needing to mark a point on both images. From the matched areas, three-dimensional coordinates could be obtained and the accuracy of facial measurements from the 3D coordinates determined. The feasibility of constructing three-dimensional images of features and faces from which facial measurements can be made was also investigated.

8.1) Stereo Matching

Stereo matching tests were performed on different image pairs and different image enhancement techniques were applied to these images. A variety of factors were looked at during the development of the matching algorithm to see how they affected the matching process. These factors include image resolution, application of different enhancement techniques and the application of an infrared flash and texture projection during image acquisition.

The use of Canny edge detection and feature enhancement through histogram equalization and contrast stretching was found to improve matching accuracy in general. Its effectiveness is even further enhanced when used in combination with texture projection. The use of texture projection on to the face gave the best matching accuracy results (83.69% mean accuracy). It especially improved the matching accuracy on featureless areas such as the cheeks and smooth skin around the eyes. The texture projection was applied to 512-by-492 resolution images together with edge detection and feature enhancement techniques. The resolution of an image also plays an important role in the matching accuracy, since higher resolution will result in more image detail. Better image detail will increase matching accuracy. A mean accuracy of just over 81% was achieved on the 1024-by-1344 resolution images by applying only edge detection and feature enhancement. Applying texture projection (especially texture projection that doesn't have a repetitive pattern) to these high-resolution images will likely lead to a higher matching accuracy. However, the image resolution affects the runtime.

As with texture projection, the use of infrared light also aids the matching process. It has advantages such as enhancing visibility of features in the images and obtaining similar brightness in image pairs, therefore reducing the need to apply certain image enhancement techniques. A mean matching accuracy of 78.3% was obtained with infrared light. In this case no edge detection or histogram equalization was used – only contrast stretching was applied. Although the matching accuracy is lower with the application of an infrared flash on the 512-by-492 resolution images in comparison with not using an infrared flash on the 1024-by-1344 resolution images, it is likely because of the lower image resolution, and not because of the infrared flash. Using an infrared flash together with texture projection led to a higher matching accuracy (average of 82.90%) than with an infrared flash alone, but this is still not as good as the results obtained with the application of only texture projection. From this it is clear that infrared should not be used together with texture projection during image acquisition, since the infrared reduces the intensity of the texture projection and therefore its effectiveness.

The use of a search window also plays a big role in the matching process – the smaller the size of the search window, the smaller the runtime and also the chance of obtaining a false match. One must however keep in mind that it should not be too small, since the correct match might then fall outside its boundaries and not be detected.

The matching accuracy discussed above was assessed by visual inspection. An improvement might be to match nodes from the left image to the right image, and then afterwards using the matched nodes on the right image as a reference and matching the nodes on the left image. This way accurate matches can be confirmed, while error matches are detected and either corrected or discarded.

A statistical comparison was also performed between manually marked nodes and matched nodes around the eyes and mouth. Maximum differences of approximately 3 mm between these coordinates were obtained. However, the x-coordinates around the eyes and mouth differ on average by less than 1 mm, while the y-coordinates differ on average by approximately 0.5 mm. Furthermore, the standard deviation for all the coordinates indicates a good average matching accuracy. The biggest coordinate difference occurred in the x-direction (horizontally), compared to the y-direction, and this can be related to the use of a rectangular search window with a

greater length than height (increasing the possibility of obtaining a false match horizontally).

Most of the coordinate differences between the marked and matched nodes are not statistically significant at the 5% level (table 6.5 and 6.6). Only 3 out of 9 coordinate differences around the eyes and 2 out of 8 coordinate differences around the mouth are statistically significant. However, of these coordinate differences that are statistically significant, all except one differ on average by less than 1 mm. The statistically significant differences of coordinates around the mouth might be due to the fact that the lips and skin of some individuals have a similar complexion, leading to false matches. This problem might be overcome with the use of texture projection. For the eyes, the statistically significant differences are related to points affected by reflectance from the eyes, which changed position in the image pair, affecting the matching. This reflection can be reduced with the use of an infrared flash. Furthermore, matching accuracy can also be affected by manual errors, i.e. inaccuracies that occurred when marking the corresponding nodes manually on the right image.

The circularity difference of the upper lip obtained between the manually marked nodes and the matched nodes is not statistically significant at the 5% level. A mean absolute difference of 2.46 and a standard deviation of 3.51 were obtained. This is a good indication that the matching results obtained with the stereo matching algorithm are satisfactory.

The upper lip circularity was measured with the use of a semi-ellipse, since it saves time (compared to outlining the upper lip manually) and since an ellipse has a shape that can easily be used to approximate the vermilion border of the upper lip. However, fitting the semi-ellipse to the upper lip gives only an approximation of the upper lip circularity, due to its shape limitations.

8.2) Three-Dimensional Reconstruction

Three-dimensional coordinates were successfully obtained with the use of the Direct Linear Transform. Delaunay triangulation and bilinear interpolation were also successfully applied to create a three-dimensional surface of these 3D coordinates.

A good accuracy was obtained with the DLT. Results obtained from the DLT were compared to known values, and an average difference of 0.51 mm in the x-direction, 0.63 mm in the y-direction and a 0.79 mm difference in the z-direction showed that the DLT has a mean accuracy within 1 mm. The sets of 3D equations developed for 3D reconstruction as described in the literature review (paragraph 3.2) would not be appropriate for the images used in this project for the following reasons: It is clear that image alignment is necessary for the equations presented by Cheng *et al.* (2000), and Cumani & Guiducci (1997), since disparity only occurs in the x-direction. Furthermore, Kearfott *et al.* (1993) noted that depth accuracy decreases with increasing object-camera distance (refer back to paragraph 3.2.1). Cumani & Guiducci (1997) noted that equations *(27)-(30)* hold for the ideal case and that unit focal length is assumed. Each of the previous methods developed to obtain 3D coordinates was developed for a specified study and tested on specific images with specified resolutions. For example, Cheng *et al.* (2000), used electron microscope images, and Cumani & Guiducci (1997) focused on almost vertical tubular objects (AVTO's) and their properties.

For the 3D surface reconstruction, Delaunay triangulation was decided upon instead of Voronoi diagrams or surface skinning with splines. This decision was made because Delaunay triangulation gives a clearer visual appearance than Voronoi diagrams of the 3D surface for this application, and because it does not require a great deal of computer memory and a large number of control points like skinning.

8.3) Processing Time

The runtime was mainly affected by the image resolution of the image pair. When 1024-by-1344 resolution images were used, it took 25-35 seconds to load the image pair, and another 30-45 seconds to obtain the matching results from 19 nodes after those nodes were marked on the left image by the user. When 512-by-492

resolution images were used, loading an image pair took 5-10 seconds, while it took 10-25 seconds for matching 19 nodes. The number of nodes didn't affect the runtime significantly during matching, but did play a bigger role during three-dimensional reconstruction of the surface. For example, when 512-by-492 resolution images were used, a 3D surface from 14 matched nodes took 15-25 seconds and from 155 matched nodes it took 30-35 seconds to obtain a 3D surface. During 3D reconstruction, the interpolation intervals between 3D points also affected the runtime. If more points were interpolated, the runtime increased. The number of points to be interpolated was also dependent on both the size of the surface area and the specified interpolation intervals.

8.4) Final Comments

An effective stereo matching algorithm has been developed based on matching nodes in an image pair and obtaining 3D data from these matched nodes for successful 3D reconstruction. By matching only the nodes, it is possible to create a mesh covering the desired facial features without the position and structure of the mesh being influenced by the error of parallax. A problem might however occur when connecting some of the matched nodes with lines to create a mesh. This problem will only occur when nodes that are far from one another and have irregular features between them (e.g. the nose) are directly connected. The line connecting these distant nodes will be straight and the change in depth due to these facial features won't be recognized. This possible problem can only be prevented by marking nodes close to one another where changes in depth occur.

The three-dimensional reconstruction is directly dependent on the matching, and if the matching accuracy is not high, flaws will occur in the reconstructed 3D surface. Facial measurements from such a reconstructed surface for FAS screening will not result in a trustworthy diagnosis. A few recommendations for improving the matching accuracy are given in the next chapter.

Chapter 9

Conclusions and Future Work

9.1) Conclusions

A characteristic facial phenotype associated with FAS is used in its diagnosis, and measurements of certain facial features are compared to population norms in order to identify subjects with FAS. Facial measurements can be obtained with the use of stereophotogrammetry, making use of two pictures of a child's face taken from different angles with digital cameras. With these two pictures and the application of stereophotogrammetric algorithms, three-dimensional information of the face can be obtained, from which reliable measurements can be made.

Although various three-dimensional or stereophotogrammetric systems and techniques have been developed in the past, they're mostly expensive and complicated, either using exclusive instrumentation or needing specialist participation. An easily operated and cost-effective method to screen for FAS on a large scale is required in South Africa, and a stereophotogrammetric tool requiring minimum equipment and specialist participation has been developed by the MRC/UCT Medical Imaging Research Unit (Meintjies *et al.*, 2002). However, measurement using this tool currently relies on manual selection of points on a computer monitor.

A matching algorithm has successfully been developed to match corresponding points in an image pair, which also reduces the time taken for manual point selection on a pair of stereo images. The matching technique has good accuracy (± 83%). Using high-resolution images with infrared light or texture projection might improve results. It must still be determined how matching accuracy influences feature measurement and therefore FAS diagnosis.

Accurate three-dimensional coordinates can be obtained through calibration techniques. However, the instrumentation for calibration is not always available and an accurate method to obtain 3D coordinates without the need for additional

equipment must be developed. Three-dimensional reconstruction is successfully achieved as long as accurate 3D coordinates are available. Matching more nodes and also denser placement of nodes will lead to 3D images with increased resolution. The runtime for the automatic matching of the nodes and to ultimately obtain a 3D reconstructed surface is satisfactory.

This thesis presents a step towards improving the efficiency of large scale FAS screening using the instrument developed by Meintjies *et al.* (2002), and also towards the construction of 3D facial images that can be used in FAS screening and other anthropometric applications.

9.2) Recommendations for Future Development

Obtaining accurate 3D coordinates and 3D reconstruction is dependent on the accuracy of the matching and will only be effective if efficient matching results are obtained. The matching procedure itself can be improved by incorporating one or more of the following steps into the algorithm:

- The implementation of certain constraints in the algorithm, e.g. a smoothness constraint, which indicates that three-dimensional surface depth changes smoothly (is continuous everywhere) and abrupt changes only occur at boundaries.
- After nodes are marked on the left images and matched on the right image, copy the matched nodes back to the left image and repeat the matching procedure to see if the same matching results occur. This way matching is performed on both images to confirm correct matches and identify false matches. The false matches can either be rematched or eliminated.
- After node matching, comparing the distances between the marked nodes in the left image to the distances between the matched nodes in the right image, and in that way determine and fix error matches.
- Another idea to identify possible false matches is to use a type of correspondence threshold value. Currently a match is identified by shifting the right node and comparing it with the left node after each shift. The best match is obtained at the position where the smallest difference between the

two nodes occurs. If this "smallest difference" is bigger than a certain value (i.e. a correspondence threshold value), it might indicate a false match.

- Currently the size of the search window is predetermined according to the collection of image pairs used, and the same size search window is applied for the whole image collection. If image pairs can be aligned as much as possible, it will lead to the use of a smaller search window size and possibly a different search window size for each image pair.
- Using colour images instead of grayscale images to see whether it might improve matching.
- Iterative optimization methods such as genetic algorithms would be more efficient in matching large numbers of nodes for 3D reconstruction.

An automatic selection method that selects relevant points automatically in an image is also being developed (Douglas *et al.*, 2003). This may be combined with the results from my project for further improvement, since the need for the user to select the necessary points on one image manually would be removed.

A limitation for this project is that stereo matching alone might not be effective for thorough three-dimensional reconstruction of facial features or the whole face, since some points may not be visible on the left and right images (due to occlusion). Therefore more than two cameras might be needed to ensure that all the points of the face occur more than once in the images.

References

Abdel-Aziz Y. I., Karara H. M., 1971, "Direct linear transformation from comparator coordinates into object space coordinates in close-range photogrammetry", *Proceedings of the Symposium on Close-Range Photogrammetry*, pp. 1-18, Falls Church, VA: American Society of Photogrammetry.

Astley S. J., Clarren S. K., 1995, "A fetal alcohol screening tool", *Alcoholism: Clinical and experimental research*, Vol. 19 (6), pp. 1565-1571

Astley S. J., Clarren S. K., 2001, "Measuring the facial phenotype of individuals with prenatal alcohol exposure: correlations with brain dysfunction", *Alcohol and Alcoholism*, Vol. 36(2), pp. 147-159

Astley S. J., Stachowiak J., Clarren S. K., Clausen C., 2002, "Application of the fetal alcohol syndrome facial photographic screening tool in a foster care population", *The Journal of Pediatrics*, Vol. 141(5), pp. 712-717

Ayache N., 1991, "Artificial Vision for Mobile Robots: Stereo vision and Multisensory perception", *The MIT Press, Cambridge Massachusetts, London England*, Chap. 3

Ayoub A., Garrahy A., Hood C., White J., Bock M., Siebert J. P., Spencer R., Ray A., 2003, "Validation of a vision-based, three-dimensional facial imaging system", *Cleft Palate Craniofacial Journal*, Vol. 40(5), pp. 523-529

Ayoub A. F., Siebert P., Moos K. F., Wray D., Urquhart C., Niblett T. B., 1998, "A vision-based three-dimensional capture system for maxillofacial assessment and surgical planning", *British Journal of Oral & Maxillofacial Surgery*, Vol. 36, pp. 353-357

Bartoli A., 2003, "Towards Gauge Invariant Bundle Adjustment: A Solution Based on Gauge Dependent Damping", *Proceedings of the ninth IEEE International Conference on Computer Vision (ICCV 2003)*, 2-Volume set

Bokil A., Khotanzad A., 1995, "A constraint learning feedback dynamic model for stereopsis", *IEEE transactions on pattern analysis and machine intelligence*, Vol. 17(9-12), pp. 1095-1100

Boyle R. D., Thomas R. C., 1988, "Computer Vision: A First Course", *Blackwell Scientific Publications*, pp. 35 - 41

Burke P. H., 1971, "Stereophotogrammetric measurement of normal facial asymmetry in children", *Human Biology*, Vol. 43, pp. 536-548

Byun J., Nagata T., 1996, "Determining the 3-D pose of a flexible object by stereo matching of curvature representations", *Pattern recognition,* Vol. 29 (8), pp. 1297-1307

Canny J., 1986, "A Computational Approach to Edge Detection", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 8(6), pp. 679-698

Cheng Y., Hartemink C. A., Hartwig J. H., Dewey Jr. C. F., 2000, "Three-dimensional reconstruction of the actin cytoskeleton from stereo images", *Journal of Biomechanics,* Vol. 33, pp. 105-113

Cumani A., Guiducci A., 1997, "Recovering the 3D structure of tubular objects from stereo silhouettes", *Pattern Recognition,* Vol. 30 (7), pp. 1051-1059

Dipanda A., Woo S., Marzani F., Bilbault J. M., 2003, "3-D shape reconstruction in an active stereo vision system using genetic algorithms", *Pattern Recognition,* Vol. 36, pp. 2143-2159

Douglas T. S., Martinez F., Meintjies E. M., Vaughan C. L., Viljoen D. L., 2003, "Eye feature extraction for diagnosing the facial phenotype associated with fetal alcohol syndrome", *Medical and Biological Engineering and Computing 2003,* Vol. 41, pp. 10-106

ESAT, 2004. Catholic University of Leuven, Department of Electrical Engineering. A tutorial. Last accessed 2004/01/05

URL http://www.esat.kuleuven.ac.be/~pollefey/tutorial/node53.html

Farkas L. G., Bryson W., Klotz J., 1980, "Is Photogrammetry of the Face Reliable?", *Photogrammetry of the face,* Vol.66 (3), pp. 346-355

Ferrario V. F., Sforza C., Poggio C. E., Schmitz J. H., 1998, "Facial volume changes during normal human growth and development", *The anatomical record 250,* pp. 480-487

Ferrario V. F., Sforza C., Puleo A., Poggio C. E., Schmitz J. H., 1996, "Three-dimensional facial morphometry and conventional cephalometrics: A correlation study", *International Journal of Adult Orthodontics and Orthognathic Surgery,* Vol. 11 (4), pp. 329-338

Ferrario V. F., Sforza C., Poggio C. E., Serrao G., 1996, "Facial three-dimensional morphometry", *American Journal of Orthodontics and Dentofacial Orthopedics,* Vol. 109 (1), pp. 86-93

Garcia D., Orteu J. J., Penazzi L., 2002, "A combined temporal tracking and stereo-correlation technique for accurate measurement of 3D displacements: application to sheet metal forming", *Journal of Materials Processing Technology,* Vol. 125-126, pp. 736-742

Gonzalez R. C., Woods R. E., 1992, "Digital image processing", *Addison-Wesley Publishing Company,* Chap. 4

Grattoni P., Cumani A., Guiducci A., Pettiti G., 1993, "Automatic harvesting of asparagus: An application of robot vision to agriculture", *Proceedings of SPIE on Mobile Robots*, Vol. 8, pp. 200-210

Hajeer M. Y., Ayoub A. F., Millett D. T., Bock M., Siebert J. P., 2002, "Three-dimensional imaging in orthognathic surgery: The clinical application of a new method", *International Journal of Adult Orthodontics and Orthognathic Surgery,* Vol. 17 (4), pp. 1-13

Huang D., Yan H., 2003, "Modeling of deformation using NURBS curves as controller", *Signal Processing: Image communications,* Vol. 18, pp.419-425

ICMIT, 2004. International consortium for medical imaging technology. 3D Reconstruction demonstrations. Last accessed 2004/03/31
URL http://icmit.mit.edu/projects/pia/image3d/images.html

Izquierdo E., Ohm J., 2000, "Image-based rendering and 3D modeling: A complete framework", *Signal Processing: Image Communication,* Vol. 15, pp. 817-858

Jähne B., 1993, "Digital Image Processing: Concepts, Algorithms, and Scientific Applications", *Springer-Verslag* , Second edition, Chapters 2&4

James G., Burley D., Clemens D., Dyke P., Searl J., Wright J., 2000, "Modern Engineering Mathematics", *Addison Wesley,* Second edition, pp. 30-32

Jin L., Fernández Pierna J. A., Xu Q., Wahl F., De Noord O. E., Saby C. A., Massart D. L., 2003, "Delaunay triangulation method for multivariate calibration", *Analytica Chimica Acta,* Vol. 488, pp. 1-14

Karara H. M., Abdel-Aziz Y. I., 1974, "Accuracy aspects of non-metric imageries", *Photogrammetric Engineering,* Vol. 40 (9), pp. 1107-1117

Kearfott K. J., Juang R. J., Marzke M. W., 1993, "Implementation of digital stereo imaging for analysis of metaphyses and joints in skeletal collections", *Medical & Biological Engineering & Computing,* Vol. 31, pp. 149-156

Kostousov V. B., Molochnikov I. L., 2002, "Flexible net approach for stereo matching", *Photogrammetric Computer Vision,* Vol. 34 (3B), pp. 126-128

Lee H., Kim T., Park W., Lee H. K., 2003, "Extraction of digital elevation models from satellite stereo images through stereo matching based on epipolarity and scene geometry", *Image and Vision Computing,* Vol. 21, pp. 789-796

Mathworks, 2004. The Mathworks. Last accessed 2004/01/05
URL http://www.mathworks.com/

May P. A., Brooke L., Gossage J. P., Croxford J., Adnams C., Jones K. L., Robinson L., Viljoen D., 2000, "Epidemiology of fetal alcohol syndrome in a South African community in the Western Cape Province", *American Journal of Public Health*, Vol. 90 (12), pp. 1905-1912

Meintjies E. M., Douglas T. S., Martinez F., Vaughan C. L., Adams L. P., Stekhoven A., Viljoen D., 2002, "A stereo-photogrammetric method to measure the facial dysmorphology of children in the diagnosis of Fetal Alcohol Syndrome", *Medical Engineering and Physics*, Vol. 24, pp. 683-689

Mikhail E. M., Bethel J. S., McGlone J. C., 2001, "Introduction to modern photogrammetry", *John Wiley & Sons, Inc.*

Mostafavi M. A., Gold C., Dakowicz M., 2003, "Delete and insert operations in Voronoi/Delaunay methods and applications", *Computers & Geosciences*, Vol. 29, pp. 523-530

Mulchrone K. F., 2002, "Application of Delaunay triangulation to the nearest neighbour method of strain analysis", *Journal of Structural Geology*, Vol. 25, pp. 689-702

PCI Geomatics, 2004. PCI Geomatics geographic software company. Online help gateway. Last accessed 2004/03/31
URL http://www.pcigeomatics.com/cgi-bin/pcihlp/FPRE

Piegl L., Tiller W., 1996, "Algorithm for approximate NURBS skinning", *Computer-Aided Design*, Vol. 26 (9), pp. 699-706

Pratt W. K., 1991, "Digital Image Processing", *John Wiley & Sons, Inc.*, Second edition, Chapters 10&16

Ras F., Habets L. L. M. H., Van Ginkel F. C., Prahl-Andersen B., 1996, "Quantification of facial morphology using stereophotogrammetry – demonstration of a new concept", *Journal of Dentistry*, Vol. 24 (5), pp. 369-374

Salvi J., Armangué X., Batlle J., 2002, "A comparative review of camera calibrating methods with accuracy evaluation", *Pattern Recognition,* Vol. 35, pp. 1617-1635

Schalkoff R. J., 1989, "Digital Image Processing and Computer Vision", *John Wiley & Sons, Inc.*, Chapters 2, 4&6

Siebert J. P., Marshall S. J., 2000, "Human body 3D imaging by speckle texture projection photogrammetry", *Sensor Review,* Vol. 20 (3), pp. 218-226

Spiegel M. R., 1968, "Mathematical Handbook of Formulas and Tables", *Mcgraw-Hill Book Company,* Part 1(4)

Stevens W. P., 1997, "Reconstruction of three-dimensional anatomical landmark coordinates using video-based stereophotogrammetry", *Journal of Anatomy,* Vol. 191, pp. 277-284

Sun C., 1997, "A Fast Stereo Matching Method", *Digital Image Computing: Technique and Applications,* pp. 95-100

Sun C., 2002, " Fast Stereo Matching Using Rectangular Subregioning and 3D Maximum-Surface Techniques", *International Journal of Computer Vision,* Vol. 47 (1-3), pp. 99-117

Unser M., 1999, "Splines: A Perfect Fit for Signal and Image Processing", *IEEE Signal Processing Magazine,* November 1999, pp. 22-38

Wang Y., 1998, "Principles and applications of structural image matching", *ISPRS Journal of Photogrammetry & Remote Sensing,* Vol. 53, pp. 154-165

Wong K. W., 1975, "Mathematical formulation and digital analysis in close-range photogrammetry", *Photogrammetric Engineering and Remote Sensing,* Vol. 41 (11), pp. 1355-1373

Appendix A: Diagram of the Matching Process

A diagram of the matching algorithm as described in paragraph 6.1 and particularly paragraph 6.1.3 is given in figure A1 for clarification purposes.

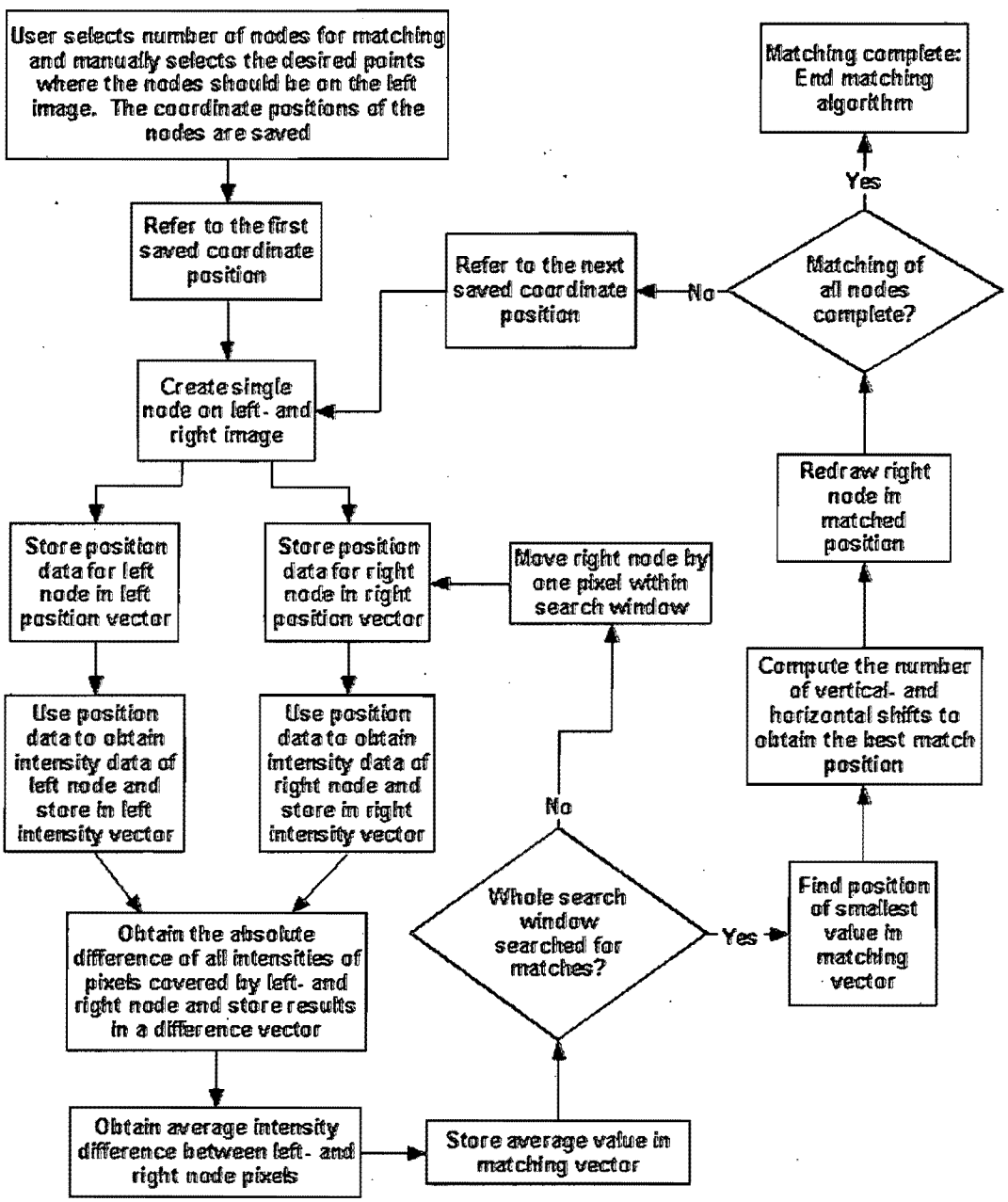


Figure A1: Diagrammatic illustration of the developed matching algorithm

Appendix B: The Direct Linear Transform

Abdel-Aziz and Karara (1971) proposed the Direct Linear Transform (DLT) to model the transformation between image space and the object space. As the name implies, the DLT estimates the three-dimensional coordinates of landmarks by a linear transformation of their two-dimensional coordinates in multiple images. Because the camera location, orientation, and focal length need not be measured, these may be adjusted freely to adapt to differences in the sizes of objects being photographed and to constraints of the workspace.

The DLT requires a minimum of six control points - which must be well distributed in 3D space - for the transformation from 2D image space into 3D object space. A calibration frame with 12 reflective markers as control points (refer back to figure 3.2 and figure 7.1) is used for this purpose. These control points are needed to solve the transformation parameters required for the transformation, and the parameters are solved using a least squares adjustment.

If the intersection of the principal axis of the lens with the image plane (i.e. the optical center of the image) is defined to have coordinates (0, 0), then the DLT is defined by the equations (Stevens, 1997):

$$x + xr^2K_1 + 2xyK_2 + (r^2 + 2x^2)K_3 = \frac{L_1X + L_2Y + L_3Z + L_4}{L_9X + L_{10}Y + L_{11}Z + 1} \qquad (B1)$$

$$y + yr^2K_1 + 2xyK_2 + (r^2 + 2y^2)K_3 = \frac{L_5X + L_6Y + L_7Z + L_8}{L_9X + L_{10}Y + L_{11}Z + 1} \qquad (B2)$$

In equations *(B1)* and *(B2)* the two-dimensional coordinates of a landmark in a single image are denoted by (x,y), while (X,Y,Z) are the three-dimensional coordinates of the same landmark in space.

The squared distance from (x,y) to the center of the image is r^2, where $r^2 = x^2 + y^2$. $L_{1\text{-}11}$ are transformation coefficients relating (x,y) to (X,Y,Z).

Each control point on the frame provides two equations for L_{1-11} and therefore it's clear that a minimum of six control points are required to solve L_{1-11}, as mentioned earlier. K_1, K_2 and K_3 are lens distortion parameters.

Equations *(B1)* and *(B2)* can be simplified and rewritten as:

$$x + Kr^2 x = \frac{L_1 X + L_2 Y + L_3 Z + L_4}{L_9 X + L_{10} Y + L_{11} Z + 1} \qquad (B3)$$

$$y + Kr^2 y = \frac{L_5 X + L_6 Y + L_7 Z + L_8}{L_9 X + L_{10} Y + L_{11} Z + 1} \qquad (B4)$$

Here K is a term that was introduced by Karara and Abdel-Aziz (1974), to correct for lens distortion. The parameters are further computed using a linear least squares adjustment:

$$L = \left(P^T P\right)^{-1}\left(P^T A\right) \qquad (B5)$$

Where:

$$L = \left[L_1, L_2, L_3, \mathrm{K}, L_{11}, K\right]^T$$

$$A = \left[x_1, y_1, x_2, y_2, \mathrm{K}, x_n, y_n\right]^T$$

$$P = \begin{bmatrix} X_1 & Y_1 & Z_1 & 1 & 0 & 0 & 0 & 0 & -x_1X_1 & -x_1Y_1 & -x_1Z_1 & -x_1r_1^2 \\ 0 & 0 & 0 & 0 & X_1 & Y_1 & Z_1 & 1 & -y_1X_1 & -y_1Y_1 & -y_1Z_1 & -y_1r_1^2 \\ \mathbf{M} & \mathbf{M} & & & & & & & & & & \\ X_n & Y_n & Z_n & 1 & 0 & 0 & 0 & 0 & -x_nX_n & -x_nY_n & -x_nZ_n & -x_nr_n^2 \\ 0 & 0 & 0 & 0 & X_n & Y_n & Z_n & 1 & -y_nX_n & -y_nY_n & -y_nZ_n & -y_nr_n^2 \end{bmatrix}$$

The subscript n refers to the number of the control points. The transformation coefficients and other relevant parameters must be solved for the left and right images respectively. The DLT is thus solved for each camera to give two sets of transformation coefficients mapping image space to object space. The 3D coordinates of any point visible in both the images of a stereo image pair with image coordinates (x_l, y_l) and (x_r, y_r), respectively, are then computed:

$$M = \left(R^T R\right)^{-1}\left(R^T N\right) \qquad (B6)$$

Where:

$$M = [X, Y, Z]^T$$

$$N = \begin{bmatrix} x_l - L_{l4} + x_l r_l^2 K_l \\ y_l - L_{l8} + y_l r_l^2 K_l \\ x_r - L_{r4} + x_r r_r^2 K_r \\ y_r - L_{r8} + y_r r_r^2 K_r \end{bmatrix}$$

$$R = \begin{bmatrix} L_{l1} - L_{l9}x_l & L_{l2} - L_{l10}x_l & L_{l3} - L_{l11}x_l \\ L_{l5} - L_{l9}y_l & L_{l6} - L_{l10}y_l & L_{l7} - L_{l11}y_l \\ L_{r1} - L_{r9}x_r & L_{r2} - L_{r10}x_r & L_{r3} - L_{r11}x_r \\ L_{r5} - L_{r9}y_r & L_{r6} - L_{r10}y_l & L_{r7} - L_{r11}y_r \end{bmatrix}$$

Here l and r refer to the left and right images respectively.

Appendix C: Frame Coordinates for DLT Accuracy Testing

The accuracy of the DLT was determined by comparison of 3D coordinates of markers on a calibration frame obtained from the DLT (using two sets of 2D coordinates from an image pair of the frame) with the known 3D coordinates of the markers. The indicated coordinates are those of the 12 markers on the calibration frame.

2D coordinates (mm)				3D coordinates (mm)					
Left image		Right image		Obtained			Accurate known		
x	y	x	y	X	Y	Z	X	Y	Z
484	77	418	101	127.34	-357.16	363.57	126.68	-357.64	363.16
476	117	451	140	130.26	-358.83	264.60	130.62	-359.10	264.64
467	166	489	187	132.97	-358.83	168.19	133.58	-358.20	167.72
396	268	316	284	191.13	-210.49	358.72	192.57	-208.37	358.91
423	326	380	339	165.54	-211.24	257.52	165.22	-212.28	258.22
466	387	460	395	133.71	-212.70	166.97	132.70	-213.10	167.24
104	78	40	91	403.66	-352.57	370.09	403.43	-353.13	370.79
65	116	42	129	406.62	-354.67	271.95	406.39	-354.50	272.78
22	162	46	177	408.43	-354.79	175.92	408.81	-354.23	175.31
198	263	118	281	340.90	-208.01	365.72	340.73	-208.66	363.44
129	315	88	334	370.45	-210.98	263.19	369.98	-210.68	265.22
42	372	37	395	408.46	-209.07	175.23	408.76	-209.46	174.25

Table C1: The 2D frame coordinates obtained from the image pair, and the 3D coordinates obtained with the DLT compared to known 3D frame coordinates

Appendix D: List of Matlab Functions

A description of all the relevant Matlab m-files written is listed in this appendix. Table D1 lists all the files developed for the stereo matching and 3D-reconstruction algorithm, while table D2 lists the files developed for the statistical comparison study. The real source files can be found on the attached CD.

Filename (Alphabetical)	Description
Calibrate.m	This program is used to calibrate the object space using the control frame.
Compute3D.m	Function to find 3D coordinates using measured 2D coordinates from 2 cameras and parameters obtained from the program DLT.m.
ComputeXYZ.m	This function finds the (X,Y,Z) coordinates of the calibration frame.
Coord_adjust.m	This program takes the 3D coordinates obtained from the DLT and checks for negative coordinates. If negative coordinates are found, all coordinates are adjusted so that all coordinates are positive for 3D reconstruction.
Delaunay_3D.m	Delaunay triangulation is applied to create a 3D mesh. Bilinear interpolation is applied to create a denser mesh, representing a 3D surface.
DLT.m	This function takes the matrices u and v, representing the (x,y) coordinates respectively of the images, and finds the Direct Linear Transformation parameters and the variance.
Edge_apply_black1.m Edge_apply_black2.m	These programs apply contrast stretching as indicated, after which edges are detected from either the imadjust (contrast-stretched) image or the histeq (histogram equalization) image respectively, and copied on to the image with intensity of 0 (black). Image modification is also

	performed by changing pixels with intensity of 255 to an intensity of 254. This is done to avoid errors when identifying nodes with intensity of 255 during matching.
Edge_apply_white1.m Edge_apply_white2.m	These edge-application programs apply contrast stretching as indicated, after which edges are detected from the imadjust (contrast-stretched) image or the histeq (histogram equalization) image respectively, and copied on to the image with intensity of 254 (white). Image modification is also performed by changing pixels with intensity of 255 to an intensity of 254. This is done to avoid errors when identifying nodes with intensity of 255 during matching.
Image_enhance_hi.m	This program applies only contrast stretching onto an image. Contrast stretching values of (0.3 0) and (1 1) are used. Image modification is also performed by changing pixels with intensity of 255 to an intensity of 254. This is done to avoid errors when identifying nodes with intensity of 255 during matching.
Image_enhance_lo.m	This program applies only contrast stretching onto an image. Contrast stretching values of (0 0.2) and (0 0.8) are used. Image modification is also performed by changing pixels with intensity of 255 to an intensity of 254. This is done to avoid errors when identifying nodes with intensity of 255 during matching.
Line_creation_delaunay_l.m Line_creation_delaunay_l2.m	Either white lines or black lines are created on left image with the aid of Delaunay triangulation. This creates a 2D mesh on the image.
Line_creation_delaunay_r.m Line_creation_delaunay_r2.m	Either white lines or black lines are created on right image with the aid of Delaunay triangulation, so that a 2D mesh is created on the image.
Node_amount.m	The user is asked to select the number of nodes for matching, and the number is recorded.

Node_creation1.m Node_creation2.m Node_creation3.m	Nodes are created on both left and right images for displaying the initial node positions on both images. Nodes of size 9-by-9 are created with intensity of 255 (white). The size of the search window is also specified as well as the position where the search for the corresponding point must start (if necessary). The size of the search window is respectively specified as 40-by-15 for the images with texture projection; 40-by-25 for the new infrared images; 115-by-15 for the initial infrared images.
Node_creation4.m	Nodes are created on both left and right images for displaying the initial node positions on both images. Nodes of size 11-by-11 are created with intensity of 255 (white). The size of the search window is also specified as 40-by-60, since the high-resolution images obtained with the Sony cameras are used.
Node_input.m	The left image is displayed and node positions are marked on the image by the user using the crossbars on the image. The coordinates of these nodes are saved.
Node_input_r.m	The left image is displayed with the marked nodes drawn. The right image is also displayed so that the same node positions can be marked on the right image by the user using the crossbars on the image. The coordinates of these nodes are saved. The nodes must be marked in exactly the same order as the nodes on the left image were marked.
Node_matching1.m Node_matching2.m Node_matching3.m	The matching of the nodes is performed on the enhanced images. The node is created on the left and right enhanced image and compared to the node on the right image after each shift. Position data is stored and used to obtain intensity data of the covered nodes. A matrix with the same

	dimensions as the search window is created to store the data of the node on the right image after each shift of the right node. The new coordinates of the matched nodes are determined and recorded and the matched nodes are redrawn on the right facial image. Node size of 9-by-9 or 11-by-11 is matched (depending on the applied program).
Nodedata_storage1.m	For both images in the image pair a vector is created to store the coordinates of the initial node pixels as well as the number of pixels.
Pic_display1.m Pic_display3.m	All the results are shown. These include images with marked and matched nodes, with resulting 2D mesh, as well as 3D surface reconstruction. The 2D coordinates and obtained 3D coordinates are also displayed. (Because of different image pairs used, different variables were used and thus different programs were needed.)
Pic_display2.m Pic_display4.m	All the results are shown. These include images with marked and matched nodes as well as images with the resulting 2D mesh. The 2D coordinates of the marked and matched nodes are also displayed. (Because of different image pairs used, different variables were used and thus different programs were needed.)
Pixelmodify1.m Pixelmodify2.m	Image pairs are slightly modified by changing pixels with intensity of 255 to intensity of 254 (or pixels with intensity of 0 to intensity of 1) before nodes are created on the images. This is necessary since the nodes are identified by their intensity of 255 (or 0, depending on the applied program). Thus errors might occur if other nodes with intensity of 255 or 0 are mistaken as node pixels.
Read_irpics.m Read_irpics2.m	Grayscale images (512-by-492 resolution) are read and resized. The new set of images with

	infrared flash and texture projection is used. (In one program the first set of infrared pictures are used and in the other program the new set is used.)
Read_sonypics.m	Colour images (1024-by-1344 resolution) are read, converted to grayscale and resized. Histogram equalization is also applied to improve image clarity.
Read_texturepics.m	Grayscale images (512-by-492 resolution) are read into Matlab. The new set of images with applied texture projection is used.
Run_ir_match.m	This program calls all the other relevant programs so that the matching algorithm can be performed successfully on the images obtained with the Digital Smart Cameras (with the infrared flash only).
Run_irtext_match.m	This program calls all the other relevant programs so that the matching algorithm can be performed successfully on the images obtained with the Digital Smart Cameras (with the applied texture projection and infrared flash).
Run_irtext_reconstruct.m	This program calls all the other relevant programs so that matching and 3D-reconstruction can be performed successfully on the images obtained with the Digital Smart Cameras (with the applied texture projection and infrared flash).
Run_selfmark.m	This program calls all the other relevant programs so that nodes can be marked manually on the left and right facial image. This is done on the images obtained with the Digital Smart Cameras, with texture projection applied. The corresponding marked nodes are then used for 3D-reconstruction.
Run_sony_match.m	This program calls all the other relevant programs so that the matching algorithm can be performed successfully on the high-resolution images

	obtained with the Sony Cameras.
Run_texture_match.m	This program calls all the other relevant programs so that the matching algorithm can be performed successfully on the images obtained with the Digital Smart Cameras (with the texture projection only).
Run_texture_reconstruct.m	This program calls all the other relevant programs so that matching and 3D-reconstruction can be performed successfully on the images obtained with the Digital Smart Cameras (with the texture projection only).
SmallnodeImage_creation.m	Copies of images are made and the matched nodes are drawn as smaller (5-by-5) nodes with intensity of 255 on these images. This is done for displaying purposes to get a better idea of the node position of the corresponding nodes in the image pair.
Smallnode_image_creation_l.m Smallnode_image_creation_r.m	The marked nodes are drawn as smaller (5-by-5) nodes with intensity of 0 (black) on the left image and right image respectively. This is done for displaying purposes to get a better idea of the node position of the corresponding nodes in the image pair.

Table D1: List of Matlab m-files used in the developed stereo matching and three-dimensional reconstruction algorithm

Filename (Alphabetical)	Description
Blackmatch.m	The matching of the nodes is performed on the enhanced images of the eyes. The nodes are created on the left and right enhanced image and compared to the node on the right image after each shift. Position data is stored and used to obtain intensity data of the pixels covered by nodes. A matrix with the same dimensions as the search window is created to store the data of the node on the right image after each shift of the right node. The new coordinates of the matched nodes are determined and recorded and the matched nodes are redrawn on the right facial image. Nodes are matched on eye images with edges redrawn with intensity of 0 (black). 11-by-11 Size nodes are matched.
Edge_apply.m	This program applies contrast stretching, after which edges are detected from the histeq (histogram equalization) image, and copied on to the image with intensity of 0 (black) and of 254 (white) respectively. Image modification is also performed by changing pixels with intensity of 255 to an intensity of 254. This is done to avoid errors when identifying nodes with intensity of 255 during matching.
Ellipsenode_creation_b.m	Nodes are created on both left and right mouth images for displaying the initial node positions on both images. Nodes of size 11-by-11 are created with intensity of 255 (white). These images with the nodes displayed on them can be compared to the images with the ellipse fitted around the upper lip.
Ellipsepic_display_b.m	The results of matching 4 nodes around the upper lip and fitting a semi-ellipse to the coordinates of these nodes are shown. These include images with a semi-ellipse fitted to the marked left and right image nodes, and images with a semi-ellipse fitted to the matched nodes of size 11-by-11. The picture ID and the 2D coordinates of the marked and matched nodes are also

	displayed.
Eyenode_amount.m	Six node points are specified and the user is asked to mark them on the eyes. The coordinates of the marked nodes are saved.
Eyenode_creation.m	Nodes are created on both left and right eye-pair images for displaying the initial node positions on both images. Nodes of size 11-by-11 are created with intensity of 255 (white). The position is specified where the search for the corresponding point must start.
Eyepic_display.m	The results of matching 6 nodes on the eyes are shown. These include images with marked and matched nodes of sizes 11-by-11. The picture ID and the 2D coordinates of the marked and matched nodes are also displayed.
Fitellipse_b.m Fitellipse_c.m	A semi-ellipse (whole ellipse in the 1st program) is fitted around the upper lip. The ellipse parameters are computed for fitting an ellipse to the manually marked left and right nodes, as well as for fitting an ellipse to the matched nodes of node size 11-by-11. This way the resulting ellipses can be compared. In the 2nd program, line parameters for a line closing the bottom of the semi-ellipse is also computed and the line drawn.
Fitellipse_3D_b	A semi-ellipse is fitted around the upper lip. The semi-ellipse parameters are computed for fitting a semi-ellipse to 3D coordinates obtained from the manually marked left and right nodes, as well as for fitting a semi-ellipse to the matched nodes of node size 11-by-11. This way the resulting semi-ellipses can be compared.
Get_eye_image.m	Matrices (in .mat format) of children's eyes are read into Matlab and used as images.
Get_mouth_image.m	Grayscale images of children's mouths are read into Matlab.
Load_ellipsedata2.m	Data read from the relevant Excel file is saved and applied. The x- and y-coordinates of the 4 nodes around the upper lip are saved in vectors for further use in creating a semi-ellipse around the upper lip.

Load_eyedata.m Load_mouthdata.m	Data read from the relevant Excel file is saved and applied. The x- and y-coordinates of the 6 nodes around the eyes (or 4 nodes around the mouth) are saved in vectors for further use in demonstrating the matched vs. marked nodes.
Mouthmatch11.m Mouthmatch9.m	The matching of the nodes is performed on the enhanced mouth images. The node is created on the left and right enhanced image and compared to the node on the right image after each shift. Position data is stored and used to obtain intensity data of the covered nodes. A matrix with the same dimensions as the search window is created to store the data of the node on the right image after each shift of the right node. The new coordinates of the matched nodes are determined and recorded and the matched nodes are redrawn on the right facial image. Node sizes of 11-by-11 and of 9-by-9 are matched respectively.
Mouthnode_amount.m	Six node points are specified and the user is asked to mark them on the mouth. The coordinates of the marked nodes are saved.
Mouthnode_creation.m	Nodes are created on both left and right mouth images for displaying the initial node positions on both images. Nodes of size 9-by-9 and of size 11-by-11 are created with intensity of 255 (white). The position is specified where the search for the corresponding point must start.
Mouthpic_display.m	The results of matching 4 nodes around the mouth are shown. These include images with marked and matched nodes of sizes 9-by-9 and 11-by-11 respectively. The picture ID and the 2D coordinates of the marked and matched nodes are also displayed.
Nodedata_storage2_b.m Nodedata_storage3.m	For both images in the image pair a vector is created to store the coordinates of the initial node pixels as well as the number of pixels. This is done for the 11-by-11 node images.
Pixelmodify3_b.m	Image pairs are slightly modified by changing pixels with intensity of 255 to intensity of 254 before nodes are

	created on the images. This is necessary since the nodes are identified by their intensity of 255. Thus errors might occur if other nodes with intensity of 255 are mistaken as node pixels. This is performed on images for creation of 11-by-11 nodes.
Run_3Dellipsefit	This program calls all the other relevant programs so that 3D coordinates of the marked and matched nodes can be obtained. A semi-ellipse is then fitted around the upper lip 3D coordinates for the determination of upper lip circularity. This is performed on the mouth images that were cropped from the high-resolution images obtained with the Sony Cameras. This program was used to perform a statistical comparison study to determine the matching efficiency of the developed matching algorithm.
Run_ellipsefit_uplip.m	This program calls all the other relevant programs so that an ellipse can be fitted around the upper lip for the determination of the upper lip circularity. This is performed on the mouth images that were cropped from the high-resolution images obtained with the Sony Cameras. This program was used to perform a statistical comparison study to determine the matching efficiency of the developed matching algorithm.
Run_eyematch.m	This program calls all the other relevant programs so that the matching algorithm can be performed successfully on the eye images. These images were obtained from the high-resolution images obtained with the Sony Cameras. This program was used to perform a statistical comparison study to determine the matching efficiency of the developed matching algorithm.
Run_mouthmatch.m	This program calls all the other relevant programs so that the matching algorithm can be performed successfully on the mouth images. These images were cropped from the high-resolution images obtained with the Sony Cameras. This program was used to perform a statistical comparison study to determine the matching

	efficiency of the developed matching algorithm.
Whitematch.m	The matching of the nodes is performed on the enhanced images of the eyes. The node is created on the left and right enhanced image and compared to the node on the right image after each shift. Position data is stored and used to obtain intensity data of the covered nodes. A matrix with the same dimensions as the search window is created to store the data of the node on the right image after each shift of the right node. The new coordinates of the matched nodes are determined and recorded and the matched nodes are redrawn on the right facial image. Nodes are matched on eye images with edges redrawn with intensity of 255 (white). 11-by-11 Size nodes are matched.
Writepics.m	Images with ellipse drawn are saved/ written to a file for further use. (This program is currently not in use since all the images are already saved.)

Table D2: List of Matlab m-files used in the statistical comparison study