THE IMAGE PROCESSING FOR THE TARGET CENTRE DETECTION IN DIGITAL IMAGE

MSc Thesis, 1992

R.G. Xue

Supervisor: Professor H. Ruther

Supervisor: Professor G. De Jager

Department of Surveying Engineering

Department of Electronics Engineering

University of Cape Town

THE IMAGE PROCESSING FOR THE TARGET CENTRE DETECTION IN DIGITAL IMAGE

R.G.XUE

UNIVERSITY OF CAPE TOWN

ACKNOWLEDGEMENTS

I would like to thank the following people and bodies, without whom this study would not have been possible.

Professor **H. RÜTHER**, Head of department of Surveying Engineering, University of Cape Town, supervisor, for his support and guidance. He has spend a lot of time for me to indicate the right direction of the study.

Professor **G. De Jager**, Department of Electronics Engineering, University of Cape Town, supervisor, for his support and guidance.

Mr A. Ganger, Mr G.VAN Der Vlugt, Mrs val Atkinson, for their a lot help and advice.
I also would like to thanks my whole family for their help.

Abstract

Accurate estimation of the parameters of geometric shapes present in a grey-level image is required in various machine vision and computer vision problems. Depending on the required level of accuracy and the condition under which the estimation is to be carried out, various combinations of external factors must be taken into account. In this thesis a generally applicable practical method for accurate estimation of the parameters of circular and elliptical objects (targets) on digital images is suggested. The accuracy of the estimated parameter depends not only on the global interpolation technique used, but, as well, on the compensation of errors caused by environment, instrumentation and object characteristics. In this thesis, first, as a preliminary step in accurate parameter estimation of quadratic curves, a sequential distortion-compensation procedure is formulated. The major contribution to the distortion arises from the transformation of a shape from the object space to the image space and this problem must be overcome.

For photogrammetric or metric machine vision applications target centres must be located on the image. Features such as photogrammetric targets can be represented by border pixels, that is discrete edge points. For the accurate estimation of the coordinates of such edge points a subpixel edge detector based on the principle of the sample moment preserving transform is employed. The target centre can then derived from the image coordinates of these edge points.

As an alternative means of target centre detection a novel accurate method for circular and elliptical objects is suggested. The method is named RGM (Radius of Gyration Method). It is

derived from the Moment Method. Application of this technique provided satisfactory results. Another edge detection idea also suggested in this thesis. The proposed methods were applied to segmented and enhanced Images in order to explore the possibility to improve target centring accuracy by preprocessing of the image.

This thesis comprises of five chapters. Chapter one describes basic principles of the digital image, digital image construction and the present status of the digital photogrammetry system, named PHOENICS (PHOtogrammetric ENgineering and Industrial digital Camera System), as developed by H. Rüther (1989).

The target's shape analysis in the digital image are presented in chapter two.

Chapter three presents the algorithms to detect and locate target on the digital image. These are the least squares adjustment technique, moment method, moment-preserving for edge detection as well as test methods for the evaluation of the various alglorithms. The novel RG method is presented in chapter four. Chapter five introduces the theory of some image processing methods.

CONSTENT

Chapter One: The Image Constructure 1

1.1 Intruduction 1

1.2.1 Image Source Analysis 2

1.2.2 Image Format 3

1.2.3 Digital Image 6

1.3 Sampling 6

1.4 Gray Value of the Pixel 9

1.5 The PHOENICS System 9

1.5.1 The PIP Video Digitizer Board 10

1.5.2 The CCD Video Camera 13

1.6 The PIP Frame Grabbing and PHOENICS Software .. 13

1.7 The Technique of Image Creation 14

Chapter Two: Target Analysis and Measurement 16

2.1 Introduction 16

2.2 Ideal Target Design 17

2.3 Representation of Real Target
on Digital Images 19

2.4 Target Board Design and Measurement 21

2.5 Camera/Target Array Configuration 25

Chapter Three: The Center of the Target Detection 29

3.1 Introduction 29

3.2 Least-Square Method 29

3.3.1 Using Bestfitting Method to Detect
the Centre of the Elliptical Targets 32

3.3.2 The Bestfitting Method to Detect
the Circular Target 36

3.4 The Moment Proservation Method

for the Edge Detection 36
3.5 Program Design 39
3.6 Discussion of Some Problems and
Improving the Program 40
3.7 The Program Test 41
3.8 Suggest a Edge Detection Method 43
Chapter Four: A New Centre of Target Detection Method .. 45
4.1 Intorduction 45
4.2 The Moment Centre Gravity Method 45
4.3 The Radrus of Gyration Method 48
Chapter Five: Image Processing of Targets Array 54
5.1 Introduction 54
5.2 Spatial Domain 54
5.3 Averaging Method 56
5.4 Median Filtering 58
5.5 Image Shapping 58
5.6 Prewitt and Sobel Operators
for Edge Detection 61
5.7 Hough Transform to Detect
Centre of the Elliptical Target................ 63
5.8 Results 64
Six Conclusion 69
Reference 71
Appendix A 78
Appendix B 93

CHAPTER ONE: THE IMAGE CONSTRUCTION

1.1 INTRODUCTION

Recently there has been a growing trend towards collecting and processing images in digital form for photogrammetric applications. While there has been considerable work done on general digital image processing in such areas as image coding, image restoration, and feature extraction, there has been little or no attention paid to the effects of such processing on the geometric fidelity of the image. The problem is motivated by the need for accurate measurements from remotely sensed imagery, which is of prime importance to the photogrammetric communities. Many of these images are in digital from. Thus, photogrammetric analysis which deals mostly with metric aspects of images must be combined with digital image processing and feature extraction procedures, such as edge detection and location techniques. Photogrammetrists worldwide have responded to the challenge presented by the developments in computer hardware, solid state cameras and image processing technology. A number of socalled real time photogrammetry or near real time photogrammetry camera systems have emerged in recent years and successful applications to close range photogrammetry problems been reported. In this chapter, we will present the basic digital image principles such as digitizing of an image, digital image structure and image sampling. Finally, the photogrammetric hardware of PHOENICS a digital camera system developed at UCT is explained.

1.2.1 IMAGE SOURCE ANALYSIS

Light is radiant energy that produces a visual sensation by stimulation of the retain of the human eye. A real light source always has finite nonvanishing dimensions, with the "point source" being a particularly important case in optics. When a photographic image of an object is taken, the object surface may be seen as an array of point sources in the 2-dimensional object space in a (α, β) coordinate system. Each of the array points on the object surface has it's own light reflecting characteristics. Each point of the object surface can be seen as a secondary point light source, reflecting light energy from a primary source such as the sun or an electrical light. When using an image formation system, the light reflected from the point source is focused on a 2-D image plane with coordinates (x, y) as shown in Fig. 1.2.2-1. The physical image structure can thus be interpreted as an array of pixels. Each pixel of the image corresponds to a point from object space. The light energy intensity received by each pixel can be converted into an image signal. This can be a conversion to an analog signal by an analog system such as a TV or video camera, or to a digital signal derived by A/D conversion through a frame grabber from an analog image or directly from a digital video camera. When used in a photogrammetric model, these signals can reproduce the object's shape, size, colour and gray scale distribution states.The quality of the image is related to many factors such as the environment, lighting and equipment. A good quality image will have hight spatial and gray level resolution, correct gray level contrast, and no noise. The quality of these parameters is obviously correlated directly to the correctness and completeness of the information captured on the image. The digital image consists of an array of picture elements or pixels each providing a gray value representing the amount of light reflected from the object onto the imaging chip in the pixel's location. The density and size of the pixels together with their gray level resolutions are crucial for the information content of the final image. The image is represented numerically by an array of discrete points, each located in the pixel centre and associated with a gray value.

1.2.2 IMAGE FORMAT

Consider a point M, in the object space (α, β), and the image of that point in the image plane (x, y) which is created by an image formation system. The image formation system creates the image point (x, y) by acting upon the radiant energy propagating from the corresponding object point. However, the image formation system receives radiant energy components not only from one object point (α, β) but from all other points in the object. The energy radiated or reflected from the surface points will vary as a function of each surface point's characteristics, such as orientation of the surface to the imaging system and colour and texture of the surface. The final gray value recorded for each pixel will further depend on the nature and sensitivity of the imaging system. Each point of the object has as its image in the plane (x, y) a diffraction spot whose shape is determined by the shape of the aperture which delimits the bundle. It is necessary to combine the intensities of all the diffraction spots in order to form the image of the entire object. an image content is dependent on object shape, colour, brightness, and contrast. Spatial resolution is obviously a primary parameter in the quality of the digital image and in turn the representation of the object. If the distance between two neighbouring pixels could be equal to zero, then the resolution would be infinite and thus ideal. The distance between two neighbouring pixels on an imaging surface is restricted by technical limitations. At present the separation between two pixel on an imaging chip in a CCD camera is typically of the order of 10 to 15 micrometers. Generally, the image is defined as a rectangular matrix of the form

$$0 \leq x \leq L_x, \; 0 \leq y \leq L_y \qquad (1.2.2\text{-}1)$$

where L_x is length of the image and L_y is width of the image. x and y are coordinate system in image plane.

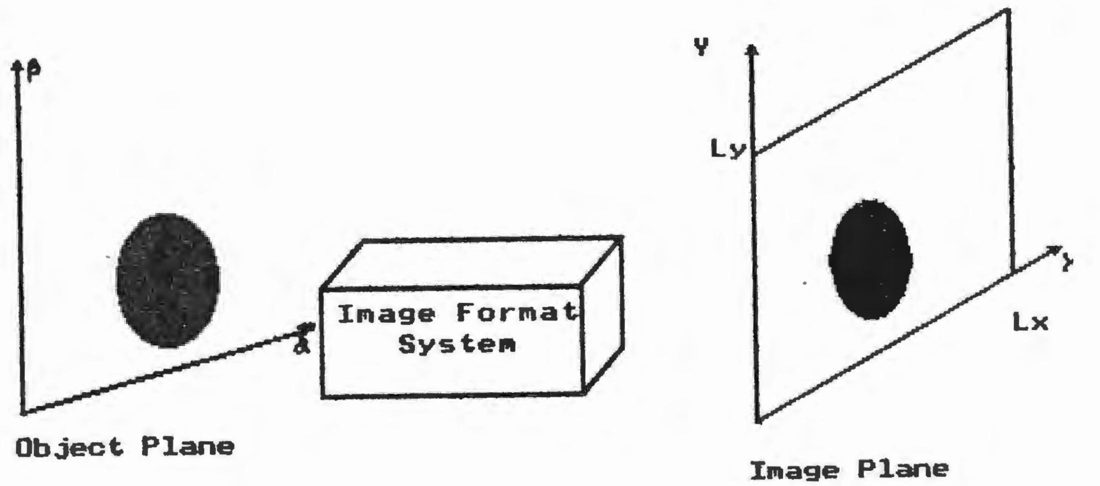


Figure 1.2.2-1 Image Format System

As discussed above, the point value of the image function is called the intensity or gray level. Any point level is a finite value, that can be expressed by a positive real number.

The image function f(x, y) is a duality, finite, positive function, which can be either continuous or discrete. If the function f(x, y) is a continuous analytic function, the image is called an analogue image [35], if it is discrete then the image is digital.

The information of each point on the analogue image can be described by the voltage signal that corresponds to the emissive light energy of the point on the object. Fig.1.2.2-2 shows the TV image signal that is typical of the analogue image signal. The digital image is central to this thesis and requires further discussion.

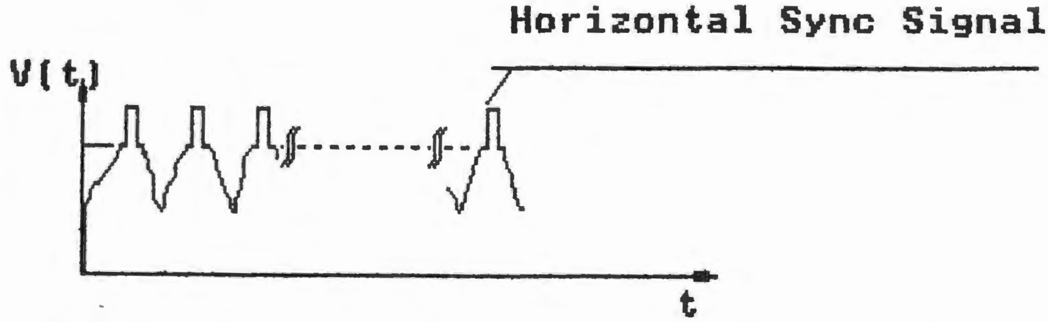


Fig. 1.2.2-2 TV signal in coordination system of time and voltage. The voltage is change is change follow continuously with time

1.2.3 DIGITAL IMAGE

In order to be able to process an image in a computer it must be in digital form, that is in form of a discrete function. The analogue image received from the video or CCD camera can be converted to a digital image by means an A/D converter. This can for example, be achieved through the interfacing of an image processing card between camera and computer. An image processing card typically has an A/D converter as one of it's components. The digital signal is a discrete function in image apace represented in a data array in which for each pixel a numerical value in binary form is contained.

1.3 SAMPLING

Sampling, i.e. the previously mentioned conversion from analog to digital format, is the first step in the digital image formation. In the analog format the image is given as a 2-D continuous and band limited function $f(x, y)$. x, y $(-\infty,\infty)$.

In the fourier transform terminology this function has the form equation

$$F(u, v) = \mathcal{F}(f(x, y)) \qquad (1.3\text{-}1)$$

For $U \in \{-W_u, W_u\}$ and $V \in \{W_v, W_v\}$ function (1.3-1) becomes

$$F(U, V) = 0 \qquad (1.3\text{-}2)$$

where U and V are expressed as u and v direction bands in the Fourier domain . All information in the original image will be contained in a small area that is a W_u x W_v rectangle, as shown in Fig. 1.3(d). Consider the sampling function

$$s(x\, y) = \sum_{m=-\infty}^{\infty} \sum_{n=-\infty}^{\infty} \delta(x-m\Delta x,\ y-n\Delta y) \qquad (1.3\text{-}3)$$

f(x, y) is analog image in time domain, s(x, y) is sampling function in time domain, Function(1.3-3) is a 2-D impulse function matrix as shown in Fig. 1.3(c). The function $F_s(x, y)$ can be obtained by the product of sampling function s(x, y) and the analog image function f(x, y) expressed as

$$f_s = s(x,y)\, f(x,y) = \sum_{m=-\infty}^{\infty} \sum_{n=-\infty}^{\infty} f(x,y)\, \delta(x-m\Delta x, y-n\Delta y) \qquad (1.3\text{-}4)$$

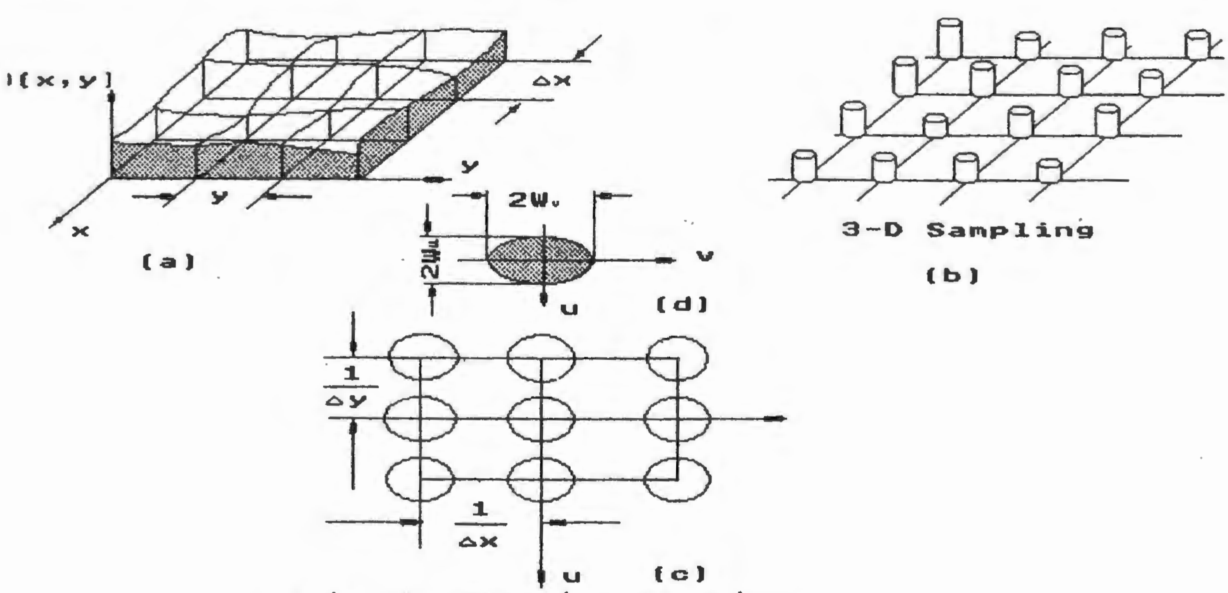
$$= \sum_{m=-\infty}^{\infty} \sum_{n=-\infty}^{\infty} f(m*\Delta x, n*\Delta y)\, \delta(x-m*\Delta x, y-n*\Delta y)$$

Using function (1.3-4) as a sampling method as well as using a small rectangular point matrix for uniform sampling, the original image is defined in digital form. A sampling position is described by

$$x = m * \Delta x, \quad y = n * \Delta y$$

with m, n = counter (0, 1, 2,....) in x and y direction, Δx Δy = sampling interval in x and y direction. The Fourier transform of a sample function is

$$S(u, v) = \mathcal{F}(s(x, y)) = \frac{1}{\Delta x}\frac{1}{\Delta y}\sum_{m=-\infty}^{\infty}\sum_{n=-\infty}^{\infty}\delta\left(u-m\frac{1}{\Delta x}, v-n\frac{1}{\Delta y}\right) \qquad (1.3\text{-}5)$$



(a) is the Sampling Function
(b) is 3-D Sampling
(c) is 2-D Sampling
(d) is R Area on u-v Plane
Fig. 1.3-1. The Image Sampling

Because of $f_s = s(x, y)\, f(x, y)$, the Fourier transform function of the sampling function and the analog function is given by a convolution of S(u, v) with F(u, v).

$$F_s(u, v) = \mathscr{F}(f_s(x, y)) = \mathscr{F}(s(x, y)\, f(x, y))$$
$$= S(u, v) * F(u, v) \qquad (1.3\text{-}6)$$

1.4 GRAY VALUE OF THE PIXEL

The information contained in each pixel of the digital image is the pixel gray value . When the image has been completely sampled, W have a sampling function, f_s (m,n), that is a rectangular array. Before f_s(m,n) can be utilised for further processing it must undergo quantization, that is converting sampling voltage to logical values with $k = 2^m$ levels, where m = 0,...,7. In practice, pixel gray values are resolved into 4 to 8 bits, so that each pixel can have 16, 32, 64, 128 or 256 levels of gray. Similar to an increase in spatial resolution, an increase in gray level resolution results in improved image quality. Usually, one allocates low numerical values to dark and high values to bright.

1.5 THE PHOENICS FRAME GRABBING SYSTEM

The **PHOENICS**(**PHO**togrammetric **EN**gineering and **I**ndustrial Digital **C**amera **S**ystem) system has been designed by **H. Rüther** and **Parkyn** [17] in 1990. The hardware configuration of PHOENICS (as shown in Fig. 1.5-1) is composed of a PC, a video frame grabber board, one or more CCD cameras, one or more external monitors, and a Parallon 8088 compatible parallel processor. It is currently being applied to surface measurements of a variety of objects [36] and in a modified to the position of patients for proton beam treatment of brain tumours. The following algorithms and image processing techniques are employed:

- Target detection,
- Image quality analysis,To generate an optimum image of a number of targets a target board was designe
- Object edge detection,
- Target centre location.

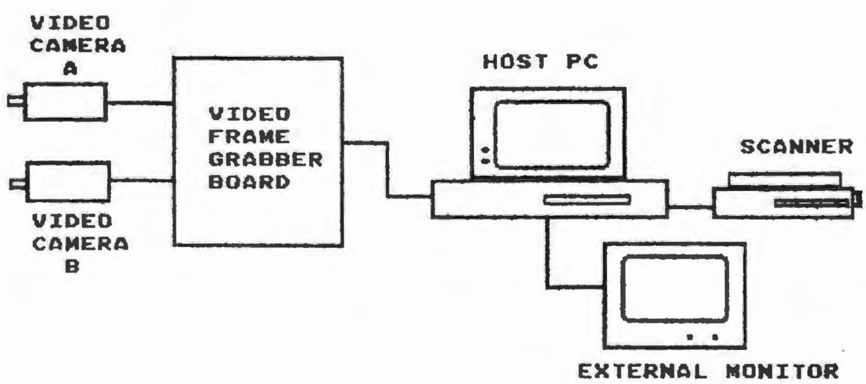


Fig. 1.5-1 **PHOENICS** Photographic System

1.5.1 THE PIP VIDEO DIGITIZER BOARD

PIP is a video frame grabber-digitizer board with IBM PC competiblitity (Fig. 1.5-1). It has resolution of 512 x 512 pixels mode with eight bits per pixel. PIP has two sync, the one being an external sync mode and the other being an internal sync mode. PIP is capable of operating in continuous or single frame grab mode and it also has builtin video keying capabilities. Frames which have been stored in the onboard frame buffer can be loaded into the PC's memory or onto disk. Fig. 1.5.1-1 shows a block diagram of PIP.

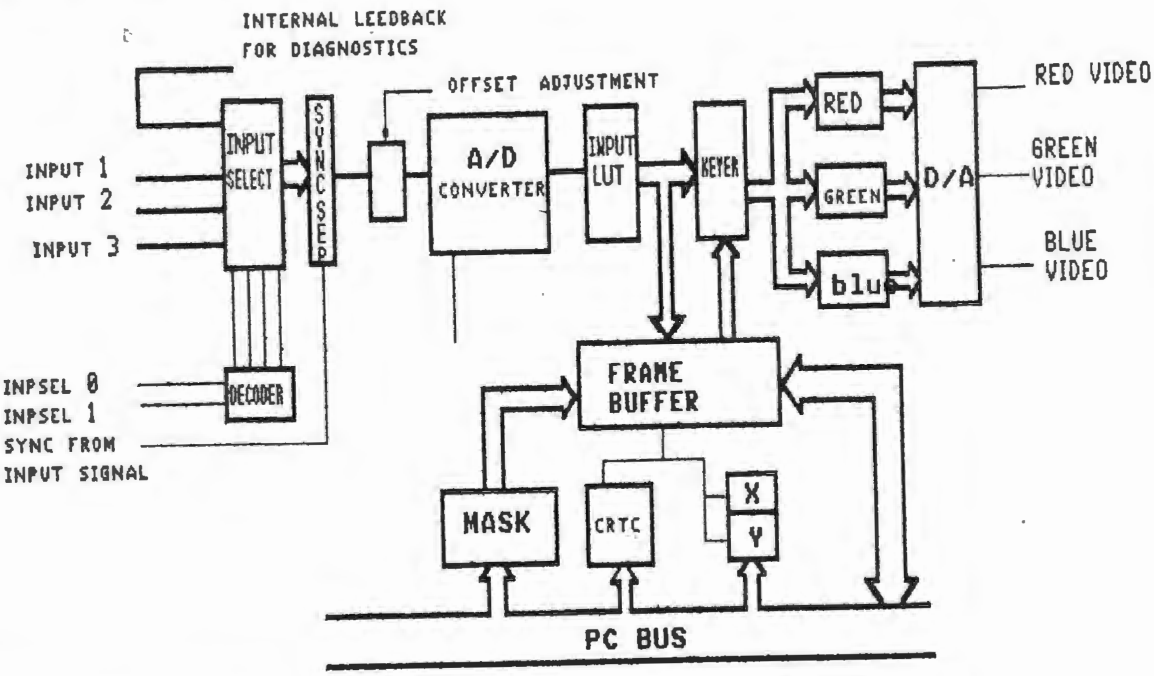


Fig. 1.5.1-1 PIP Image Processing Interface Diagram

Input signals are selected, in software, from one of three input ports on PIP. PIP will lock onto the input signal's sync and use the sync signal to drive the video function of the board. PIP is provided with 256k bytes of frame buffer RAM, which is used to store frame grab data. PIP's processor is a Synertek SY6845ECRT controller. The frame buffer is read and write accessible from the system unit, x,y coordinates are used as reference for the position location within the buffer. The frame buffer address registers can be set to automatically increment whenever a pixel is read or written to by the user. When a frame is grabber, It is taken from the selected input port and digitized. It then passes through the input LUT and is stored in the frame buffer. There are two frame grabbing models: continuous, and snapshot. The operations typically executed on the camera/frame grabber combination for each image are, in simplified form [17]:

- Image integration and transfer on the CCD sensor;
- Output of the image in analogue form;
- Digitising of the video image, that is converting the analogue signal into 8 bit bytes by means of an A/D converter;
- Passing the data through a LUT in which the gray values coming from the camera are transformed for output in a user selected form;
- Storing of the image in the frame buffer;
- Converting the image with a D/A converter for display on a monitor.

The first and second steps are camera functions. An external processor's control can be used to snap images as required. Other steps are image functions that can be accomplished on board the Matrox card using a combination of inbuilt hardwired operations and a macro command language based in a software package. The Matrox PIP board allows for transfer of the image array, memory resident on the board to the systems memory via direct memory access (DMA) and to the hard or floppy disk. An image size is 512 x 512 pixels with 256 gray levels, requiring a 262,144 bytes (2,097,152 bits) buffer. The 256 gray levels can be described by 8 bits:

$$Buffer\ size = 512x512x8 \qquad (1.5.1\text{-}1)$$

The gray scales of each pixel can be:

$$G(i, j) = \sum_{n=0}^{7} 2^n + 1 \qquad (1.5.1\text{-}2)$$

where G(i, j) is the gray value of each pixel in the digital image, i is the number of row in the digital image window, and j is the number of the colum ofthe digital image.

1.5.2 THE CCD VIDEO CAMREA

The CCD (Charge Coupled Device) camera will produce fully interlaced black and white television pictures with 625 lines at a frame frequency of 50 Hz. Its head is fitted with a solid-state CCD chip. The resolution of the CCD sensor used for this project is 500 x 582 pixels with a physical pixel size of 17 x 11 um. However, only 468 x 568 pixels on the sensor array are active. When combined with a 512 x 512 frame grabber format the 468 horizontal pixels are resampled to convert to 512 frame grabber pixels, While 56 vertical pixel rows are lost in the transfer.

1.6 THE PIP AND PHOENICS SOFTWARE

The PIP software is a library package that enables the control of the PIP board within the computer system. This software consists of routines that provide PIP hardware control of functions such as frame buffer input/output and synchronisation. LUT manipulation, graphics, image display and image processing functions are included as library function into the PIP software. The PHOENICS system software has two major components:

- image processing software
- photogrammetric software

both components are under continuous revision with the intention to produce a flexible master system from which specific application oriented turnkey versions can be developed as required. At the present stage of the PHOENICS development, images can be captured, stored in various forms, enhanced and threshoulded. Noise can be reduced, targets can be detected and target enters can be found without operator assistance. Grids projected on an object surface can be followed on the image and grid intersection coordinates are provided. The detected targets and intersections are indicated by circles displayed on the RGB monitors. Independent image correlation does not yet form part of PHOENICS. Instead, the operator is prompted by the software to identify points indicated on RGB monitors. In the final phase(the photogrammetric treatment of the image), direct linear transformation (DLT) or bundle adjustment algorithms are employed for the determination of 3-dimensional object point coordinates. Development of automatic image correlation algorithms is in progress.

1.7 THE TECHNIQUE OF IMAGE CREATION

As discussed earlier the quality of the image is dependent on a variety of factors. The main factors are the intensity and the nature of the light radiating from the image formation system. The light of the image source is very important for selecting an appropriate threshold value in order to detect the edge of the object or curve and to do the image analysis. If high intensity light are used during image grabbing, the image is often distorted by noise from reflecting lights off the object surface. If week intensity light are used. Gray scale resolution of the image is reduced. Another problem is the effect of perspective view on an object shape. For example, a circle will appear as an ellipse on the image when viewed from an angle.

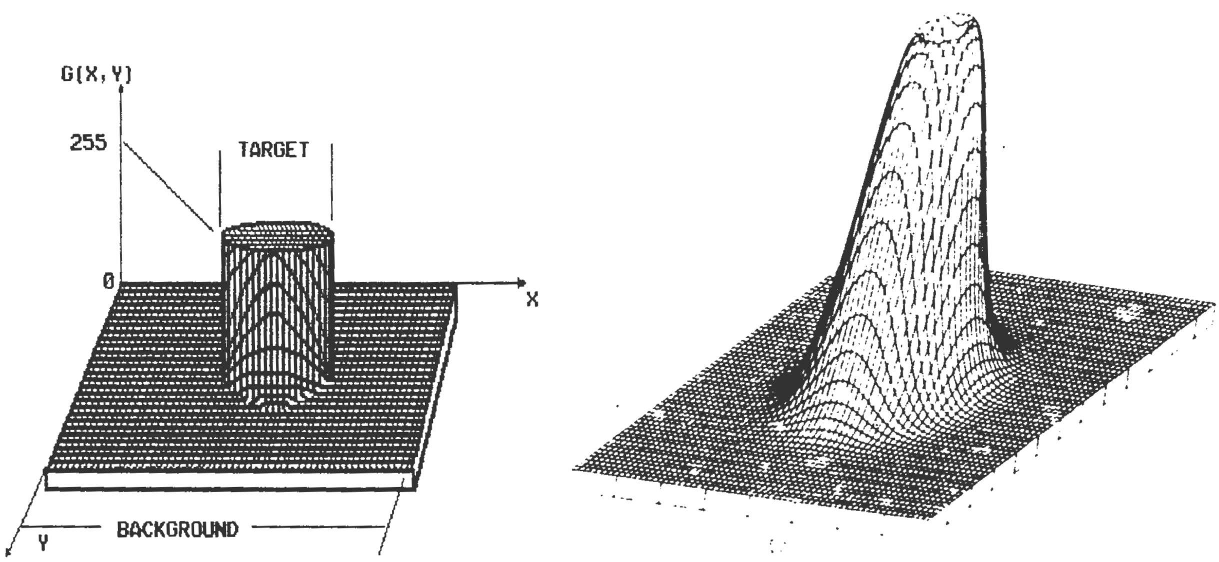
CHAPTER TWO: TARGET ANALYSIS AND MEASUREMENT

2.1 INTRODUCTION

Photogrammetric targets are marks placed in selected positions on the object in order to make it possible to determine the position of these object points in space.Target detection is the process in which such targets are automatically detected and their position on the image is determined. Targets are typically either concentric circles or intersections of gridlines. The target detection process is based on the identification of distinguishing properties of the target images such as gray level differences between target and background, target size or target shape. Target detection falls into the domain of machine or computer vision and image processing. Research on automated target detection started as early as 1960 in parallel with the development of statistical classification methods [35][8][20]. Digital image processing techniques such as edge and line detection and curve tracking made it possible to extract linear and circular features and to thus identify images of targets and determine their position on the image. On the digital image the target is represented by a small array of pixels in with gray values difference to those of the background. To obtain the accuracies typically required for photogrammetric processing it is generally not sufficient to determine the target centre within this array with pixel accuracy, instead subpixel location of the target centre must be aimed at. Under good environmental conditions, i.e. if targets can be designed to differ significantly from the background, targets can be easily detected. The determination of the target centre with subpixel accuracy is more complex and numerous methods for this process have been developed and investigated [4][5][6][9][12][15][18].

2.2 IDEAL TARGET

The shape of target of digital image in 3-D coordinate are displayed in Fig. 2.2-1.



(a) Ideal target gray scale distribution in 3-Dimension
(b) Real target gray scale distribution in 3-Dimension

Fig. 2.2-1 3-D target gray value diagram

Figure 2.2-1 (a) shows an ideal target in form of an elliptical cylinder with a vertical gradient in gray value at the target edge. The flat plane represents the background and the top of the elliptic cylinder the uniform target surface. In the digital image such an ideal case would be represented by uniform high gray levels (255) on the target and equally uniform low background gray level (0). The Mathematically ideal target equation is thus:

$$G(X, Y) = \begin{cases} 255, & X,Y \text{ inside the target function} \\ 0, & X,Y \text{ outside the target function} \end{cases} \qquad (2.2\text{-}1a)$$

The target function is decided by a target shape. For example, for the square target, above function will take the form:

$$G(X,Y) = \begin{cases} 255, & X_o \le X \le X_1, \quad Y_o \le Y \le Y_1 \\ 0, & X < X_o,\ X > X_1,\ Y < Y_o,\ Y > Y_1 \end{cases} \qquad (2.2\text{-}1b)$$

where X_o and Y_o are left and top of a target edge position, X_1 and Y_1 are right and bottom of a target edge position. One of the objectives of this thesis is the detection of the centre of elliptical and circular targets. The circle is a special case of the ellipse. The general ellipse is defined by the function 2.2-2 as follows:

$$\frac{[(X-X_0)COS\theta+(Y-Y_0)SIN\theta]^2}{A_o^2}+\frac{[(Y-Y_0)COS\theta-(X-X_0)SIN\theta]^2}{B_o^2}=1 \qquad (2.2\text{-}2)$$

Where : X_o = X coordinate of ellipse centre,

Y_o = Y coordinate of ellipse centre

θ = angle of rotation of the ellipse

A_o = semi-major axis of the ellipse

B_o = semi-minor axis of the ellipse

Because of the pixel nature of the digital image ,it is only possible to generate an image of an ideal ellipse from equations (2.2-1) and (2.2-2) if the ellipse axes are oriented parallel to the image axes. In all other cases asymmetric pixel configurations will arise. However, Rubinstein and Rüther have successfully produced synthetic target images by convolution of a point spread function with the ellipse equation. The convolution results in a target image with a continuous fall-off from the target edge to the background. For testing purposes ideal circular and elliptical targets were generated using PIP graphic commands. The ideal targets were used to test software (R.G. XUE and H.RÜTHER - appendices A) for the automated detection of the target centres by a variety of algorithms described later.

2.3 REPRESENTATION OF REAL TARGETS ON DIGITAL IMAGES

Target location is widely used in disciplines such as military science, industry, medical science,photogrammetry and surveying engineering. In practical applications the target's quality is a very important factor for the accuracy of the centre location. The quality of the hardware may impair image of the target's quality which include target's shape, distortion, resolution and generate noise etc in the digital image. The difference between a typical real target and the ideal target equivalent is shown in Figures 2.2-1.a and 2.2-1.b. The real target edge typically shows a gradual fall-off or shoulder with gray values changing from high on the target plateau to low values on the background. Section through the target result in a gray value curve from the background gray level to the target level and back to the background in each column or row of the pixel array as shown in Fig. 2.3-1.

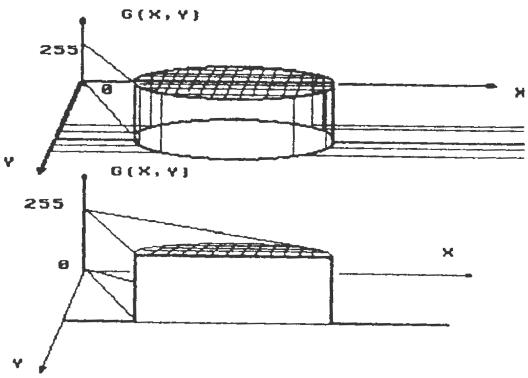


Fig. 2.3-1(a) Ideal target and target section

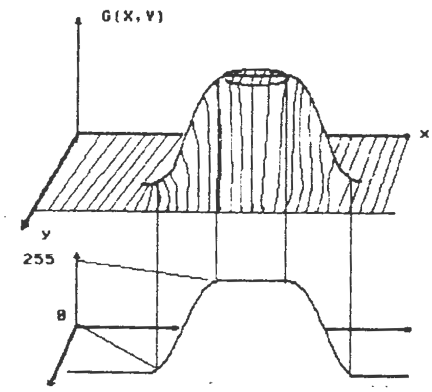


Fig. 2.3-1 (b) real target and target section

Consider a good quality image without noise and a narrow shoulder. This means the edge of the target is clearly defined and can be easily detected, as most edge detection algorithms relay on changes of gray levels as indicators of edges. It is obvious that increasing shoulder width results in decreasing definition of the edge and subsequently in a less reliable information from the edge detection algorithm. Another very important aspect is the shape of the object. Because an image is a perspective view, a circular target will appear elliptical in shape on the image plane. The orientation of the ellipse axis depends on the relative position of the target image formation system and image plane and the object. To test the software for the quality of the detection of target centres, a high quality target board was constructed. This will be described later and test results will be reported in the following section.

2.4 TARGET BOARD DESIGN, MEASUREMENT AND TEST

To generate an optimum image of a number of targets a target board was designed. The board design had to satisfy two criteria

- high precision of target position
- optimal conditions for image processing.

In practice this meant that

- targets had to be suitable for precise measurements, both, by mechanical means for the board calibration and by image processing techniques for the image processing stage.
- targets had to be of distinctly higher reflectance than the background of low reflectance to facilitate automatic detection and clear edge definition.

A total of 20 circular targets was attached to the board in a 4x5 array. The background was covered with a black paint of low reflectance to minimise noise and enhance target/background contrast. The 8 mm diameter targets were painted white using Tipex. Originally white paint was used on the target but the painted surface proved to be of too high a reflectance resulting in reflectance peaks in parts of the target surface or in overflow of the target onto neighbouring pixels. The application of Tipex proved superior to the white paint as it provided a rougher surface cover with more homogeneous reflectance properties over the entire target surface. Once the target were mounted on the target board the board was calibrated by measuring each target centre position with a precession of ± 0.1 mm. For this calibration the plotting table of the Aus-Jena Topocard Analog Plotter in the Department of Surveying and Geodetic Engineering was employed. The table has a horizontal and vertical micrometer scale as shown in Fig. 2.4-1.

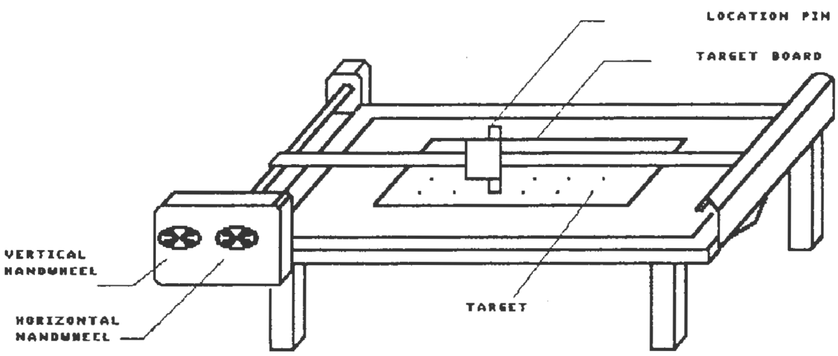
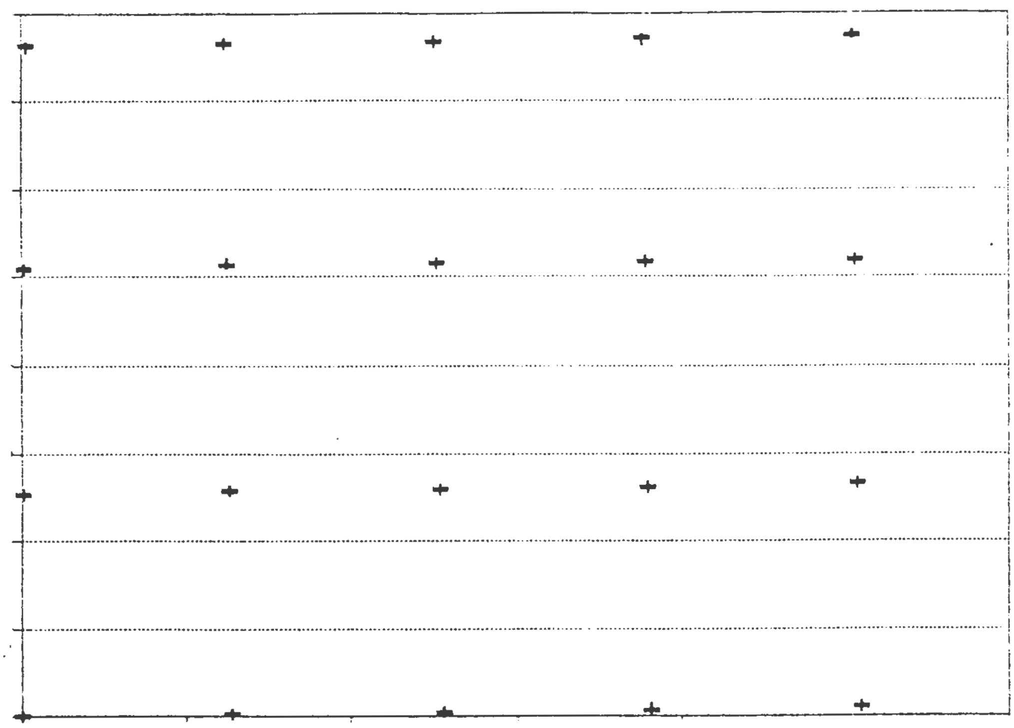


Fig. 2.4-1 The plotting table for the measurement of the target position

Two hand wheels control the movement of a location pin, which is placed over the target centre using a microscope. The target centre position can then be read from the graduated cylinder, which has a smallest subdivision of 10 micrometers. To compensate for instrumental errors in the plotting table each target was measured three times with a 90 degrees rotation of the board after completion of each full set of target measurements. Using a Similarity Transformation the three observation series were reduced to the same datum. Finally all point centres were determined by forming the arithmetic mean of the three x, y coordinate values available for each target. The coordinates of the board centre were determined as part of the calibration process. The calibration results served as a basis for one of the two methods employed for the evaluation of target centering algorithms. The observations of the targets with the board in its second position, i.e.after rotation through 90 degree, was transformed into the first observation series. This procedure served to increase both, precision and reliability of the point position coordinates. For the final calibration result the mean of the original and the transformed coordinates was evaluated.



(a) The horization lines are original target position
(b) The vertical lines are transform position after rotation 90 degree with original target position

Fig.2.4-2 Using mechanic method measurement the target board.

The transformation used was a similarity transformation based on the well known function :

$$X = b\,x - a\,y + \Delta x$$
$$Y = a\,x + b\,y + \Delta y \quad (2.4\text{-}1)$$

where

$$a = s \sin\theta$$
$$b = s \cos\theta$$

Two points are required for this transformation, while a total of 20 points was observed. These redundancies were taken into account in a least squares model of the transformation, resulting in a revised form.

$$X + v_x = b\,x - a\,y + \Delta x$$
$$Y + v_y = a\,x + b\,y + \Delta y \quad (2.4\text{-}2)$$

Where v_y and v_x are residuals. The result from the mechanical measurements were then used to define the centre of the board. For each row or column, the centre can be obtained by

$$X_{nc} = \frac{X_{ni} - X_{no}}{2} \qquad (2.4\text{-}3a)$$

$$Y_{mc} = \frac{Y_{mj} - Y_{mo}}{2} \qquad (2.4\text{-}3b)$$

Where X_{ni} is the last centre of the target in a row, X_{no} is the first centre of the target in a row. The number of the row is n. Y_{mj} is the last target in a column, Y_{mo} is the first target in a column, m is the number of the column. Therefore we can obtain n times X_c and m times Y_c. Linking up the n X_c- positions as a line and m Y_c-positions respectively as another line, they will intersect at a point which is the centre of the board. The targets are symmetrical about the centre point of the target board.

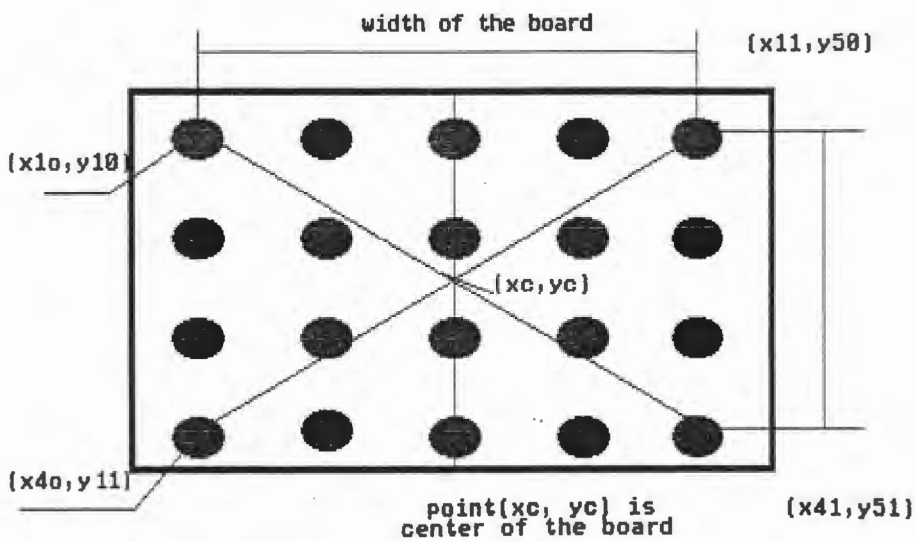


Fig. 2.5-1 The target board model

2.5 CAMERA/TARGETS ARRAY CONFIGURATION

The targets are symmetrical about the centre of the target board as in Fig. 2.5-1.

When using the camera to take an image, the target board is located so that the optical axis at the camera intersects will the plane at 90 degree. That is, all points on the target board will be symmetric about the camera access. Assume the camera is a light source outside the target board and the target board is a reflecting surface as in Fig. 2.5-2(b). A simple physical principle in geometrical optics is that for a reflected ray the

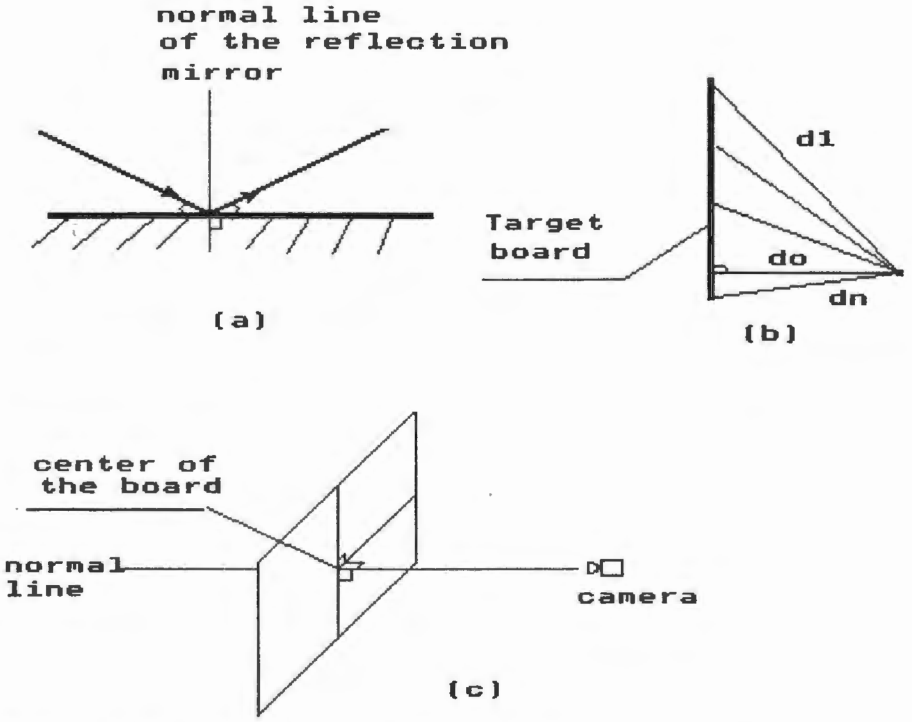


Fig. 2.5-2 Physical principle of light reflecting

incident angle is equal to reflection angle as in Fig. 2.5-2(a). This principal known as the autocollimation provides a method to fix the target board perpendicular to the optical axis of the camera. The equipment includes two video cameras, an image grabber, two small circular mirrors, a TTL digital RGB monitor and a host computer. The first step, using the Topocard drawing plotting, is to measure the centre of the board. Then place the small circular mirror in the back of the target board centre and position another small mirror in the front of the lens of camera A as shown in Fig.2.5-3. Camera A is connected to the PIP image grabber in channel 0 and camera B connected to channel 1. Place a small circular target on the centres of the mirror and the monitor. The second step is to use camera B to view camera A, which has a small circular mirror in the front of the lens. Using this mirror, camera B can find and display itself on the monitor. Then, using the PIP software package slowly adjust both of the camera positions until the centre of the mirror is displayed over the marked centre of the monitor. Then, exchange the mirror from camera A to camera B and select channel 0 using the PIP card. The second step is repeated again. After a few time adjustments, both cameras can view each other's centre, This means that both camera axis are collinear. The third step is to place target board between the two cameras. The back of the board faces camera B and the front of the board faces to the camera A. A mark is made on the centre of the mirror which is on the back side of the board. Put another mark on the front side of the centre of the board. Using camera B, view the mirror on the back of the target board. Slowly adjust the target board until the mark on the centre of the mirror is displayed on the centre of the screen. Now the target board and cameras are configured as in Fig 2.5-3 the target array is symmetrical about both cameras. Change the PIP channel to 0. Using camera A, grab an image of the target array. In this thesis, the distance between the two cameras is 5m and the target board is midway between the two cameras. The focal length of the camera is adjusted to provide target sizes greater than 3x3 pixels on the image.

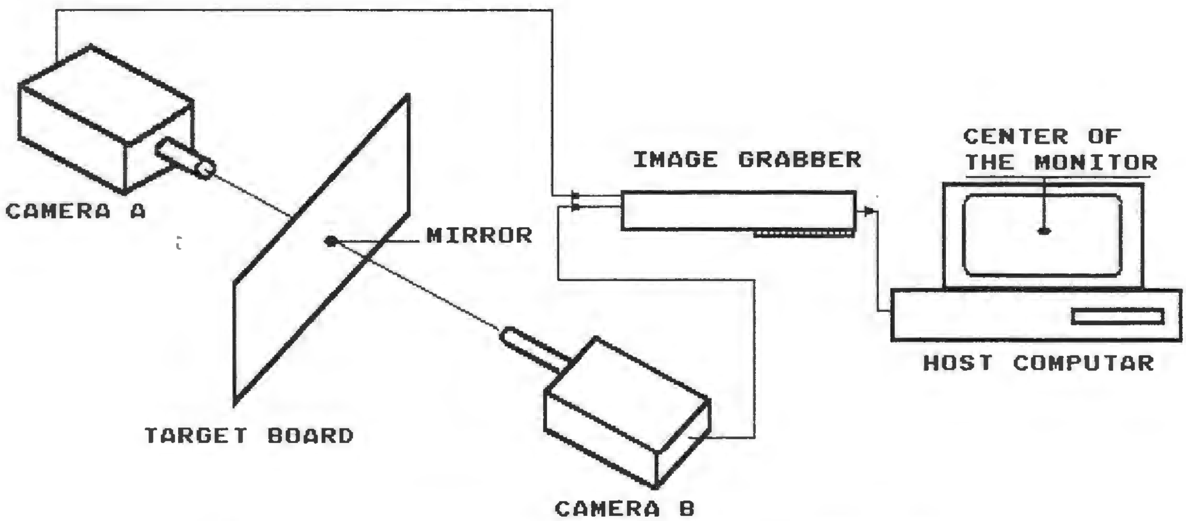


Fig. 2.5-3 Camera/Target configuration

CHAPTER THREE THE CENTRE OF THE TARGET DETECTION

3.1 INTRODUCTION

As stated in chapter two, target shape distortion will occur as a result of perspective projection and possible noise inside target and on the target edge in digital image [11] [27] [29]. This is because of the perspective projection. It is possible to determine the centre of an elliptical target as well as for a circular target by means of filling the letter into target sources. Many engineers use circular targets and very accurate results are needed to locate the target position. The technique of target location has developed very rapidly in recent years, using image processing methods to detect the target and determine it's shape and centre for target location. In this thesis, a highly accurate algorithm is used to detect the centre of the target. Through computation, we can locate the centre of the target to subpixels accuracy which can give accuracies to between than 1 part in a hundred of a pixel. The other very important for the accuracy of the centre detection is the quality of the edge detection method, which directly affects the calculated result. This chapter will discuss these factors in detail.

3.2 LEAST-SQUARE METHOD

The Least-Square Method [31][8][15] is used to determine parameters in the function Y=Y(X) if more than the necessary number of observation for determination at the unknown parameters is given. Let the sum of the squares S of the deviations of the observation data y_i, from the corresponding calculation value of the function $Y_i(x_i)$, be a minimum, where (x_i, y_i) are observation data pairs and the function is determined as

$$f(X,Y)=AX^2+BXY+CY^2+DX+EY+G=0 \qquad (3.2\text{-}1)$$

From Eq. 3.2-1, the Y can be presented as

$$Y(X)=\frac{-(BX+E)\pm\sqrt{(B^2-4AC)X^2+2(BE-2DC)X+E^2-4CG}}{2C} \qquad (3.2\text{-}2)$$

If there are N observation data pairs, through defining A, B, C,D,E and G to make the sum of the square of the deviations S a minimum.

$$S=\sum_{i=1}^{n}[y_i-Y_i(x_i)]^2 \qquad (3.2\text{-}3)$$

$$S=\sum_{i=1}^{n}[y_i+\frac{Bx_i+E\mp\sqrt{(B^2-4AC)x_i^2+2(BE-2DC)x_i+E^2-4CG}}{2C}]^2 \qquad (3.2\text{-}4)$$

Substitute Equation.3.2-2 into Eq.3.2-3, we can obtain

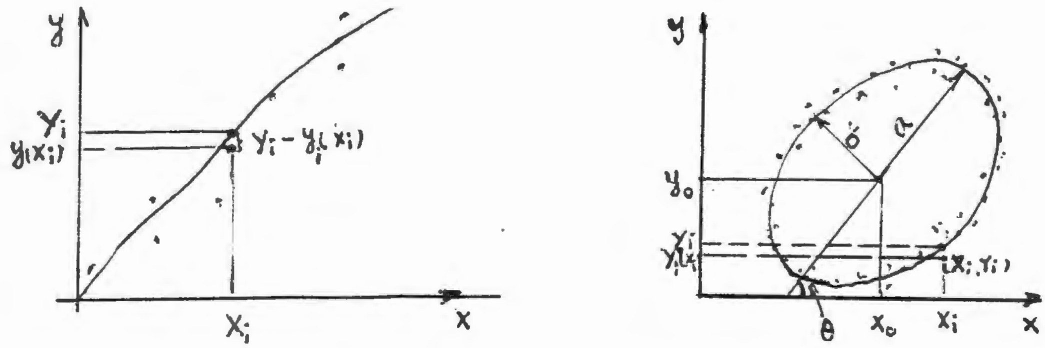


Fig.3.2-1 The Least Square Method use for detection an approximate function

When the partial derivative of S with respect to A, B, C, D, E, and G are equal to zero as

$$\frac{\partial S}{\partial A}=2\sum_{i=1}^{n}\left[y_i-Y_i(x_i)\right]\frac{\partial Y_i(x_i)}{\partial A}=0 \qquad (3.2\text{-}5a)$$

$$\frac{\partial S}{\partial B}=2\sum_{i=1}^{n}\left[y_i-Y_i(x_i)\right]\frac{\partial Y_i(x_i)}{\partial B}=0 \qquad (3.2\text{-}5b)$$

$$\frac{\partial S}{\partial C}=2\sum_{i=1}^{n}\left[y_i-Y_i(x_i)\right]\frac{\partial Y_i(x_i)}{\partial C}=0 \qquad (3.2\text{-}5c)$$

$$\frac{\partial S}{\partial D}=2\sum_{i=1}^{n}\left[y_i-Y_i(x_i)\right]\frac{\partial Y_i(x_i)}{\partial D}=0 \qquad (3.2\text{-}5d)$$

$$\frac{\partial S}{\partial E}=2\sum_{i=1}^{n}[y_i-Y_i(x_i)]\frac{\partial Y_i(x_i)}{\partial E}=0 \qquad (3.2\text{-}5e)$$

$$\frac{\partial S}{\partial G}=2\sum_{i=1}^{n}[y_i-Y_i(x_i)]\frac{\partial Y_i(x_i)}{\partial G}=0 \qquad (3.2\text{-}5f)$$

Then the quantity S will take its minimum value. From the above functions, we can obtain the parameters A, B, C, D, E and G. An application of this least-square method is given in Network Adjustment [H RÜTHER -Lecture notes] which is used for fitting curve for approximate calculation to detect the curve's shape and location. It will be discussed later.

3.3.1 USING BESTFITTING TO DETECT THE CENTRE OF THE ELLIPTICAL TARGETS

The general ellipse function is expressed in Eq.2.2-2. The equation in 2.2-2 can be converted to the general quadratic equation in Eq.3.2-1. There are five parameters A_o, B_o, θ, X_o and Y_o in the general ellipse function and there are six coefficients A, B, C, D, E and G in the quadratic equation. Through the derived function, the corresponding relations between the Eq. (3.2-1) and Eq.(2.2-2) are expressed as

$$A = B_o^2 COS^2\theta + A_o^2 SIN^2\theta \qquad (3.3.1\text{-}1a)$$

$$B = (B_o^2 - A_o^2) SIN2\theta \qquad (3.3.1\text{-}1b)$$

$$C = B_o^2 SIN^2\theta + A_o^2 COS^2\theta \qquad (3.3.1\text{-}1c)$$

$$D = 2AX_o - BY_o \qquad (3.3.1\text{-}1d)$$

$E = 2CY_o - BX_o$ (3.3.1-1e)

$G = AX_o^2 + BX_oY_o + CY_o^2 - A_o^2B_o^2$ (3.3.1-1f)

An ellipse function can be determined by using equations (3.3.1-1). The partial derivative of f(x, y) with respect to A_o, B_o, θ X_o and Y_o respectively, are as follows

$$\frac{\partial f(X,Y)}{\partial A_o} = 2A_o(N^2 - B_o^2) = R' \quad (3.3.1\text{-}2a)$$

$$\frac{\partial f(X,Y)}{\partial B_o} = 2B_o(M^2 - A_o^2) = S' \quad (3.3.1\text{-}2b)$$

$$\frac{\partial f(X,Y)}{\partial \theta} = 2NM(B_o^2 - A_o^2) = T' \quad (3.3.1\text{-}2c)$$

$$\frac{\partial f(X,Y)}{\partial X_o} = 2(A_o^2 N\sin\theta - B_o^2 M\cos\theta) = U' \quad (3.3.1\text{-}2d)$$

$$\frac{\partial f(X,Y)}{\partial Y_0} = -2(B_0^2 M\sin\theta + A_0^2 N\cos\theta) = V' \quad (3.3.1\text{-}2e)$$

$$\frac{\partial f(X,Y)}{\partial V_x} = -\frac{\partial f(X,Y)}{\partial X_o} = -U' \quad (3.3.1\text{-}2f)$$

$$\frac{\partial f(X,Y)}{\partial V_y} = -\frac{\partial f(X,Y)}{\partial Y_o} = -V' \quad (3.3.1\text{-}2g)$$

where

$M = (X-X_o)COS\theta + (Y-Y_o)SIN\theta$

$N = (Y-Y_o)COS\theta - (X-X_o)SIN\theta$

with the above functions, we can compute the matrix of the coefficients J and the matrix K as follows

$$J = \begin{bmatrix} U'_1 & V'_1 & R'_1 & S'_1 & T'_1 \\ U'_2 & V'_2 & R'_2 & S'_2 & T'_2 \\ . & . & . & . & . \\ . & . & . & . & . \\ U'_n & V'_n & R'_n & S'_n & T'_n \end{bmatrix} \quad (3.3.1\text{-}3a)$$

$$\begin{matrix} & dX_1 & dY_1 & dX_2 & dY_2 & . & . & . & .dX_n & dY_n \end{matrix}$$

$$K = \begin{bmatrix} -U'_1 & -V'_1 & 0 & 0 & . & . & . & . & 0 & 0 \\ 0 & 0 & -U'_2 & -V'_2 & . & . & . & . & 0 & 0 \\ 0 & 0 & 0 & 0 & -U'_3 & -V'_3 & . & & 0 & 0 \\ . & . & . & . & . & . & . & . & . & . \\ 0 & 0 & 0 & 0 & 0 & 0 & . & . & -U'_n & -V'_n \end{bmatrix} \quad (3.3.1\text{-}3b)$$

Where (X_1, Y_1) (X_2, Y_2)....(X_n, Y_n) are observation data.

Set

$$(-U'_1)dX_1+(-V'_1)dY_1+R'_1dA_o+S'_1dB_o+T'_1d\theta+U'_1dX_o$$
$$+V'_1dY_o+ F(X_1,Y_1)=0 \quad (3.3.1\text{-}4)$$

where i (i = 1...n) is the number of the observation points. It can be expressed by the matrix function as:

$$\boxed{\mathbf{KV} + \mathbf{JX} + \mathbf{W} = 0} \quad (3.3.1\text{-}5)$$

where the matrix **V**, **X** and **W** are respectively expressed as:

$$\mathbf{V} = \overset{dV}{\begin{bmatrix} dX \\ dY \end{bmatrix}} \quad \mathbf{X} = \overset{dX}{\begin{bmatrix} dA_o \\ dB_o \\ d\theta \\ dX_o \\ dY_o \end{bmatrix}}$$

$$\mathbf{W} = \overset{dW}{\begin{bmatrix} F(X_1, Y_1) \end{bmatrix}} \quad (3.3.1\text{-}6)$$

Hence the quasiweights are

The weight matrix **P** is expressed as the following matrix

$$\boldsymbol{P}=\boldsymbol{K}\times\boldsymbol{K}^{\boldsymbol{T}}=U'^2+V'^2$$

$$=4[(B_O^2MCOS\theta-A_O^2NSIN\theta)^2+(B_O^2MSIN\theta+A_O^2NCOS\theta)^2 \quad (3.3.1\text{-}6)$$

$$\mathbf{P}=\begin{bmatrix} P_1 & 0 & 0 & 0 & \dots & \dots & \dots \\ 0 & P_2 & 0 & 0 & \dots & \dots & \dots \\ 0 & 0 & P_3 & 0 & \dots & \dots & \dots \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ 0 & 0 & 0 & 0 & \dots & \dots & P_n \end{bmatrix} \quad (3.3.1\text{-}7)$$

P is a diagonal matrix containing the weights of the observations. The matrix **X** is the correction to the parameters which can be obtained from **V** = $\mathbf{P}^{-1}\mathbf{K}^T\mathbf{Z}$. The matrix **Z** are correlates given by **Z** = $(\mathbf{KK}^T)^{-1}(\mathbf{JX}+\mathbf{W})$

$$\begin{aligned} \mathbf{X} &= -(\mathbf{J}^T(\mathbf{KP}^{-1}\mathbf{K}^T)\mathbf{J})^{-1}\mathbf{J}^T(\mathbf{KP}^{-1}\mathbf{K}^T)^{-1}\mathbf{W} \\ &= (\mathbf{J}^T\mathbf{PJ})^{-1}\mathbf{J}^T\mathbf{PW} \end{aligned} \quad (3.3.1\text{-}8)$$

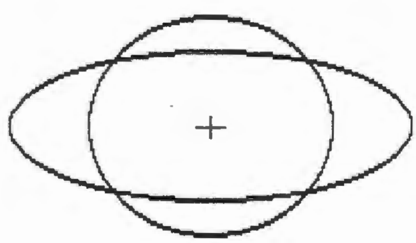
From the matrix **X**, we can determine all parameters. The accuracy of the final results of the parameters is dependent on observation data of the edge of the target. There are many methods to do edge detection. Most of these algorithms use the changing, gray values of pixels which lie on edges, to locate the edge of the object in image processing method. Although these algorithms can easily compute the edge of the object, but these algorithms will occur error in some case, spacial, when original edge information of the object that is expressed by gray value is change in some image processing. Mikhail [11] used the Moment Preserving Method to indicate the edge on the target. It solves the problem and nicely keeps the original edge informations.

3.3.2 THE BESTFITTING CIRCLE

The function for a circle has three coefficients: the centre position (X_o, Y_o) in the coordinate system X, Y and the radius R. The function is

$$(X-X_o)^2+(Y-Y_o)^2=R^2 \quad (3.3.2\text{-}1)$$

Circles have a symmetrical function. It is a specific case of the ellipse with the semi-major axis equal to the semi-minor axis. Thus, the bestfitting centre of the circle on an elliptical shape should have an approximate result as shown below:



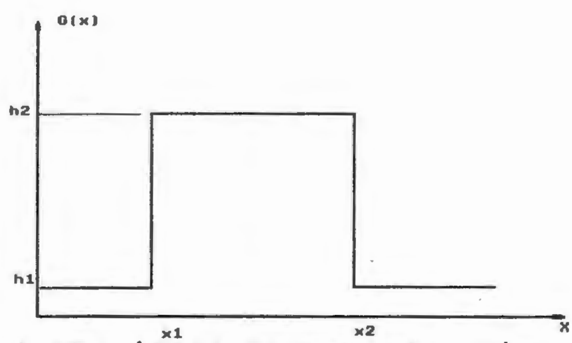
3.4 THE MOMENT PRESERVATION METHOD FOR EDGE DETECTION

Many edge detection techniques have been described in literature on image processing and pattern recognition;[10] [13] [21] [27]. Some of these papers will be discussed later. In this thesis is the Moment Preserving Method (MPM) is employed [32] [34]. The MPM is an effective method for edge detection [7] [10]. Tabatabai and Mitchell [5], Mikail, Akey and Mitchell [12], RÜTHER [16],etc have all discussed the MPM in detail. The MPM can define edge points in subpixels. Consider an ideal image of a target such that the gray level of the pixels in the background of the image is equal to zero and the gray levels of the target pixels are equal to 255.

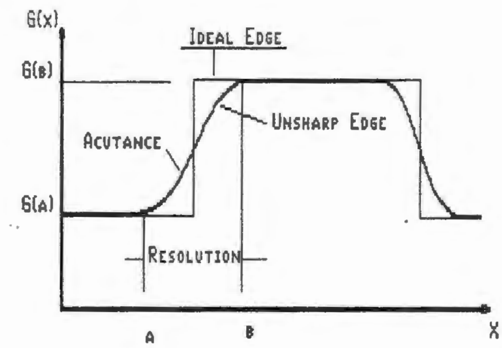
$$G(X_b\ Y_b)=0$$

$$G(X_t,Y_t)=255$$

For each row or column the form of the gray value distribution function is displayed in one dimension in Fig. 3.4-1.



(a) The ideal target edge distribution as a step



(b) Real target edge distribution

Fig.3.4-1 Target edge form in digital image

The four parameters defining the edge are: X_1, X_2, h_1 and h_2 as shown in the Fig.3.4-1. The parameters k_1 and k_2 are the numbers of the samples. The Moment Preserving Method is used as the criterion of best fit of a set <u>l</u> of n data point to the ideal edge f(s).

$$k_1 = X_1 - \frac{1}{2}$$

$$k_2 = X_2 - \frac{1}{2} \qquad (3.4\text{-}1)$$

According to Mikhail's work, was extended by RÜTHER to incorporate a second edge on the same row, the follows equation

$$m_{12} = \frac{1}{2n}[(k_2^2 - k_1^2)h_2 - (k_2^2 + k_1^2)h_1] + \frac{nh_1}{2} = \frac{1}{n}\sum_{j=1}^{n}(i - \frac{1}{2}) * I(i) \qquad (3.4\text{-}1$$

will be added into the original function as follow

$$m_j = \frac{1}{n}(k_1 - k_2)(h_1^j - h_2^j) + h_1^j = \frac{1}{n}\sum_{i=1}^{n} I^j(i) \qquad for\ \ j=1,2,3$$

be using for location edge position. Both functions will determine mentioned above.

where

$$k_1 = X_1 - \frac{1}{2}$$

$$k_2 = X_2 - \frac{1}{2} \qquad (3.4\text{-}2)$$

To calculate the variable

$$\sigma = \sqrt{m_2 - m_1^2}$$

$$C = \frac{3m_1 m_2 - 2m_1^{\ 3} - m_3}{\sigma^3}$$

$$\alpha = \frac{C}{\sqrt{4 + C^2}}$$

The parameters are equal to:

$$h_1=m_1-\frac{1+\alpha}{\sqrt{1-\alpha}}\times\sigma$$

$$h_2=m_1+\frac{1-\alpha}{\sqrt{1+\alpha}}\times\sigma$$

$$X_2=\frac{1}{2}[\frac{n(1+\alpha)}{2}+\frac{2(2m_{12}-h_1n)}{(1+\alpha)(h_2-h_1)}+1]$$

$$X_1=X_2-(1+\alpha)\frac{n}{2}$$

The MPM is used with least squares to determine the edge location. In Fig. 3.4-1 examples of an ideal and real targets are given. The ideal target edge represents steps in that the gray value suddenly changes from h_1 to h_2 and return back to h_1 at X_1 and X_2, respectively. X_1 and X_2 are the edge of the target in any one of the rows or columns in target window. The real target's edge can be described by a smooth slop gray value distribution in a few pixels. But only one subpixel can be obtained by Moment Preservation Method, the subpixel is the edge position.

3.5 PROGRAM DESIGN

We have designed and developed some software to detect the centre of targets by means of the Ellipse-Bestfitting Method and Moment Centre Gravity Method as described above. The main program contains a few subroutines: to read the gray value of the pixels from the image; the edge detection and centre detection. Another routine used, written by H. Rüther in True Basic, searches the target area and defines the threshold value. Using the above components, an algorithm can be constructed to locate edge points and the centre of the targets. The code is show in appendix A.

3.6 DISCUSSION OF THE PROBLEM AND IMPROVING THE PROGRAM

The Moment Preservation Method has some problems which need more discussion.

(a) The error points round the ellipse when using MPM to detect edge of the ellipse target

(b) The MPM detect the edge of the target in rows direction

Fig.3.6-1 Using MPM detect the edge of target

In Fig. 3.6-1, the left graphic shows the incorrectly detected points in the ellipse image. These error points are caused by the noise in the image.

The MPM is affected by the gray value of each pixel. The gray value of each pixel is not the same within any row or column in order to detect two edge. In fact the noise always stays in the original image. The error is unavoidable. Another cause for error shown in Fig. 3.6-2 is when the target's window is a little bigger than the target boundary, with one or two surplus rows and columns on each side. When using Moment Preservation Method detects these rows and columns, some incorrectly points will be found. The program have to be some changes in order to solve these problem. The edge detection in the program is separated

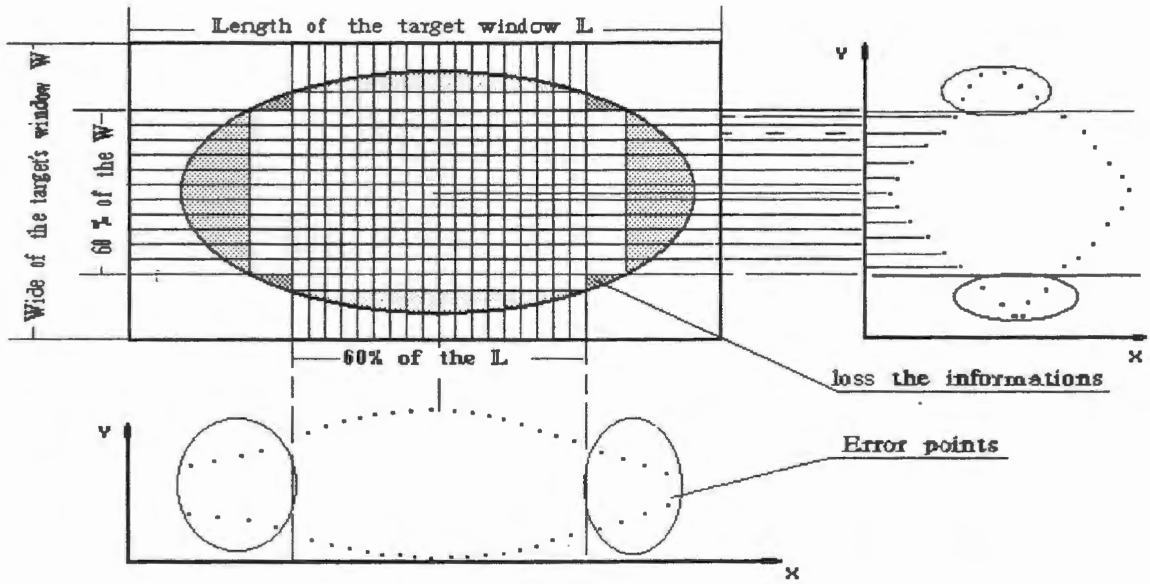


Fig.3.6-2 Use 60% information from each side of the target window to detect the edge of the target in MPM. Incorrect points are shown in the small circle

into two sections. One routine detects row by row and another one detects column by column. They find the edge points as shown in Fig 3.6-2. From Fig 3.6-2 we can find some points which, in 60% of the L and 60% of the W, are approximate to the real edge, and L is length (column) of a target window and W is width (row) of a target window. These points all concentrate in the middle area as shown Fig 3.6-2. Thus, using the Moment Preservation Method detect the edge of the object, we will lose some edge information to exchange the more accurate edge data. The particular detection rows W_e and columns L_e of the targets are equal to L_e=60%*L and W_e=60%*W. The first row and column will start from i_1 j_1, which are the first row or colum of the target window

3.7 THE PROGRAM TEST

$$i_1=\frac{W}{2}(1-60\%) \quad (integer), \quad i_n=\frac{W}{2}(1+60\%) \quad (integer)$$

$$j_1=\frac{L}{2}(1-60\%) \quad (integer), \quad j_n=\frac{L}{2}(1+60\%) \quad (integer)$$

The program is tested by proving the accuracy of the final target centres using the MPM combined with the Bestfitting of Ellipse. The result of the centre detection by above method will be compared with the centre result which is obtained by Moment Centre Gravity method. The equipment as shown in Fig. 6.7-1, designed for to test the programs which are for to detect the centre of the target. When the micrometer is turned, the target will get a known displacement.

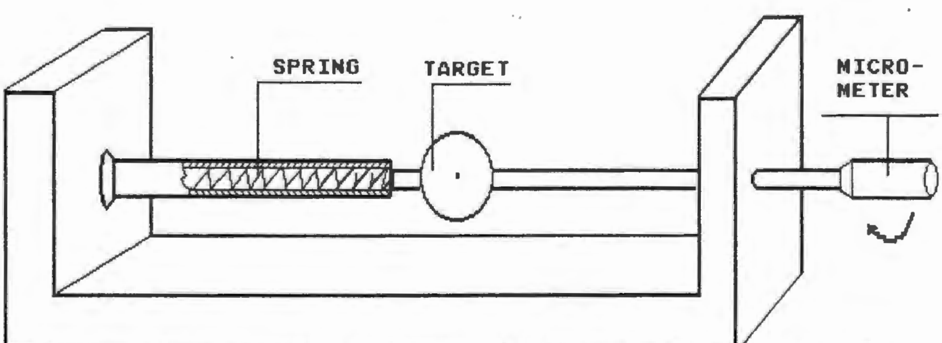
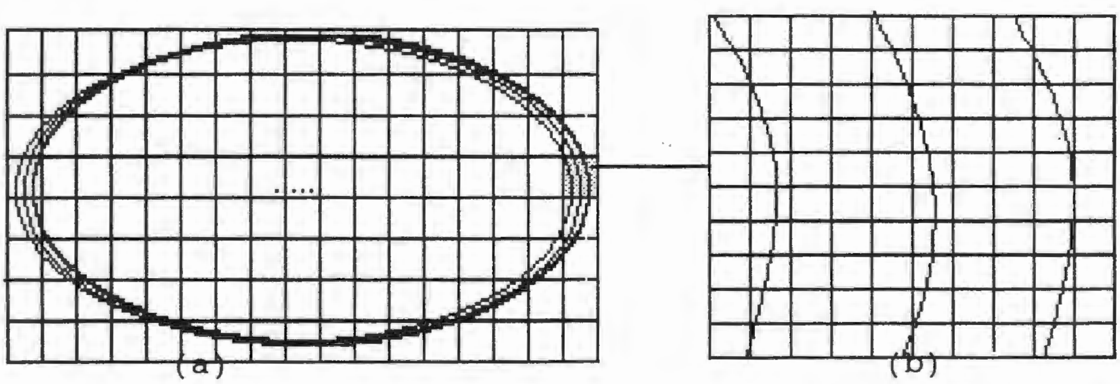


Fig. 3.7-1 the equipment is for to test the accuracy of algorithms which are for to detect centre of target

Considering that both methods for computing accuracy are dependent on the image quality, the test was carried out at two levels of displacements. One displacement is 0.1mm which is a very small and the other is 4mm which is a relatively big displacement. The small displacement will result in similar image quality. When the target has a very small displacement, the target image will only have a few subpixels displacement as shown in Fig. 3.7-2. The test uses three sets of five measurements for each of the two displacements. Fifteen target images for detecting the centre of the target were obtain at same time, which could be used for comparison with the direct measurements. The big displacement of the target is used for observation the movement of the of the targets. Due to the target having a big displacement, the target image will have a few movement of a pixels. If the results obtain a high similarly with each other, then the algorithm can be seen as accurate.



(a) The target window in digital image (b) One pixel has separate as 10 x10 subpixels

Fig.3.7-2 The target position is changed in small displacement

3.8 PROPOSAL OF A EDGE DETECTION METHOD

An edge detection method is now suggested. Assuming the gray level distribution function is a smooth, continuous, symmetrical and bounded function such as in Fig. 3.8-1.

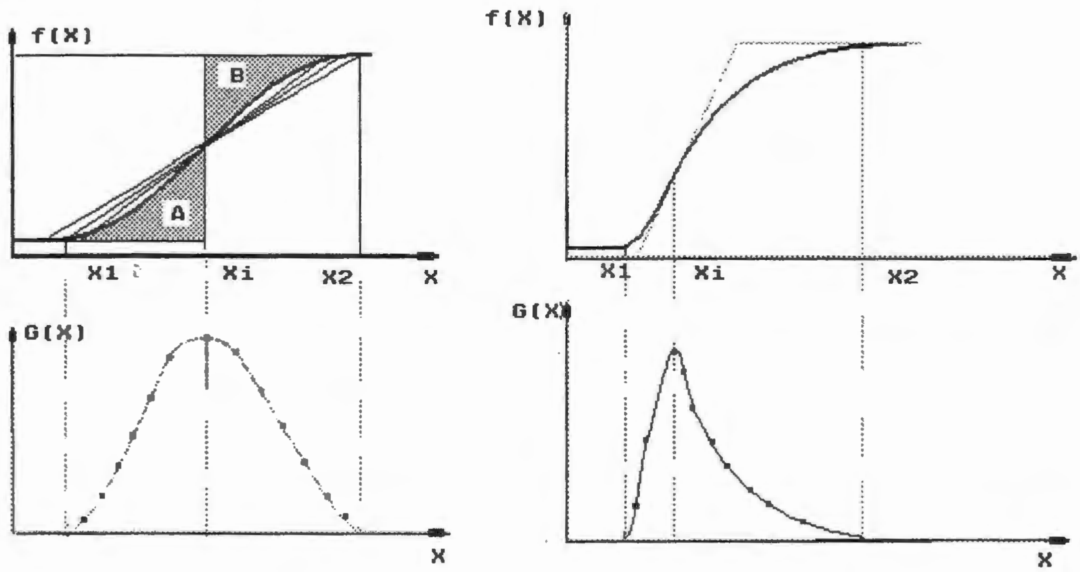


Fig.3.8-1 A edge shape and its differential function curves

This function can be defined by the Bestfitting method. If the function is as follows.

$$F(X) = AX^3 + BX^2 + C \qquad (3.8\text{-}1)$$

The parameters A, B and C can be defined by the Bestfitting method. Thus the function is an approximation function. The edge point must lie on the function's curve.

$$AREA\ A = f(x_2)(X_2 - X_i) - \int_{X_i}^{X_2} f(X)\,dX$$

$$AREA\ B = \int_{X_1}^{X_i} f(X)\,dX - f(X_1)(X_i - X_1) \qquad (3.8\text{-}2)$$

The function 3.8-2 is given a point X_i which links up the two areas A and B and lets area A equal area B. The X_i is the edge point. It has characteristics as followings:

- The point is a fulcrum so that any straight line which passes through the point, cuts the edge curve into two equal areas. Of course, such a point exists for the function
- The slope of the tangent line which passes through the point should be a maximum.
- The point should be a point of inflection so that it can be defined by the following function

$$\frac{\partial^2 f(X)}{\partial X^2} = 0 \qquad (3.8\text{-}3)$$

Due to the above statement, the function can be obtained as :

$$\int_{X_1}^{X_2} f(X)\,dx + X_1 f(X_1) - X_2 f(X_2) + [f(X_2) - f(X_1)]\,X_i = 0$$

$$X_i = \frac{\int_{X_1}^{X_2} f(X)\,dx + X_1 f(X_1) - X_2 f(X_2)}{f(X_1) - f(X_2)} \qquad (3.8\text{-}4)$$

Where the X_1 is the first pixel of the edge, the X_2 is the last pixel of the edge, $f(X_1)$ is the gray value at X_1 , $f(X_2)$ is the gray value at X_2 and X_i is the approximate edge point in subpixels. Fig. 3.8-2 shows a function which is defined as the curve of the edge of the object function 3.8-1. The another set of curves shown in fig . 3.8-1 can be discussed as the same way. We will discuss the method in another paper later.

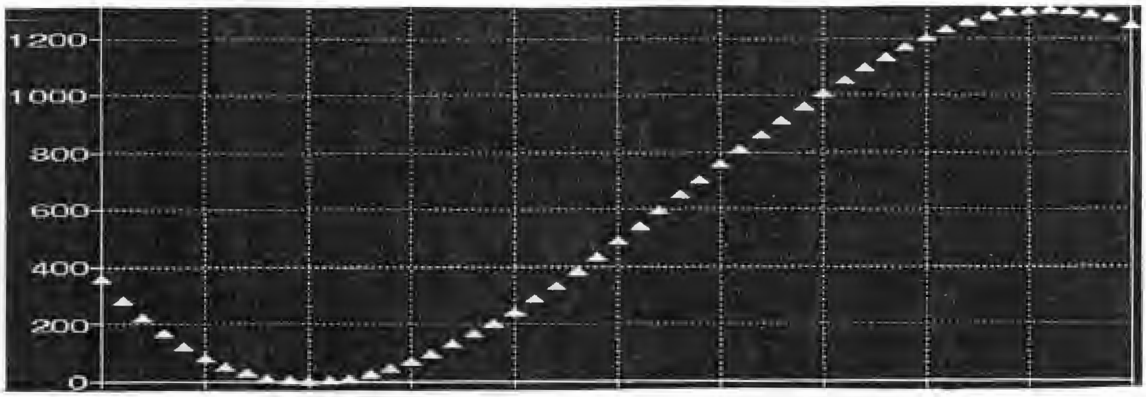


Fig.3.8-2 The curve of the edge is determined by Eq. 3.8-1

CHAPTER FOUR A NEW METHOD FOR THE CENTRE OF TARGET DETECTION

4.1 INTRODUCTION

Estimation of the centre of the target to subpixel accuracy in a gray level image is the present task. The Moment Centre Gravity Method (MCGM) assumes that the geometric centre of the object is equal to the weighted centre of gravity of the object. Davis and Rüther have discussed the moment method in detail. Mikhail and Akey [12] used moment preservation method to determine the edge position to subpixel accuracy. Nagata, Tamura and Ishihashi [31] used the least-square estimator to detect an ellipse. Safaee-Rad, Tchoukanov, Benhabib and Smith [27] have discussed the question concerning the Quadratic Curve parameter estimation. The basic idea is to keep the original image information for to locate the centre of the target. Because the image gray scale distribution is not regular, we can not easily and accurately define the edge of the target. Therefore the original information is a very important factor in target centre location. The gray scale distribution has its influence on accuracy when the gray scale distribution is not uniform. These method result in the accuracy being dependent on the image quality.

4.2 THE MOMENT Centre GRAVITY METHOD

The Moment Centre of Gravity algorithm is an area method which is widely used for calculating the centre of the target. In this thesis, the moment centre gravity algorithm is the abbreviated as moment method. The traditional equation is:

$$X_c=\frac{\sum_{i=1}^{n}\sum_{j=1}^{m}G(X_i,Y_j)X_i}{\sum_{i=1}^{n}\sum_{j=1}^{m}G(X_i,Y_j)} \qquad (4.2\text{-}1a)$$

$$Y_c=\frac{\sum_{i=1}^{n}\sum_{j=1}^{m}G(X_i,Y_j)Y_j}{\sum_{i=1}^{n}\sum_{j=1}^{m}G(X_i,Y_j)} \qquad (4.2\text{-}1b)$$

Where $G(X_i,Y_j)$ is the gray scale intensity of a pixel located in row i, column j of the digital image. X_c is the centre of the target in horizontal direction and Y_c is the centre of the target in vertical direction. The moment method uses the information from every pixel in the target window. Thus, the centre of the target is an approximate value. The gray scale intensity of a pixel in the target window represents the target's shape and size. When a noise disturbs the image, some of the pixels in the target's window will produce a noise gray value. The moment method is one of the most satisfactory methods for solving noise jamming in the digital image because this method is an average and it is assumed that the geometric centre of the target is equal to the weighted centre of the gray scale in 3-D target area of the image. So, the image distortion will impair accurate estimation of the centre of the target. Especially, if the gray scale intensity of the target is not symmetrical, then the centre of the target will be shifted. Two target's gray scale distribution are shown below:

000	000	000	010	008	000	000	000	000	000	000	000
000	000	219	255	254	214	128	000	000	000	000	000
000	255	255	255	255	255	255	233	067	000	000	000
212	255	255	255	255	255	255	255	251	075	000	000
249	255	255	255	255	25	55	255	255	242	031	000
214	255	255	255	255	25 C	55	255	255	255	158	010
130	252	255	255	255	25	55	255	255	255	242	002
017	230	255	255	255	255	255	255	255	255	255	018
000	059	245	255	255	255	255	255	255	255	223	002
000	000	053	223	255	255	255	255	255	255	130	008
000	000	000	011	110	208	254	255	228	140	001	000
000	000	000	000	000	000	008	015	010	002	000	000

(a)

000	000	000	010	008	000	000	000	000	000	000	000
000	000	219	255	254	214	128	000	000	000	000	000
000	255	255	255	255	255	226	110	067	000	000	000
212	255	255	255	255	255	233	170	130	075	000	000
249	255	255	255	255	255	245	225	155	124	031	000
214	255	255	255	255		255	235	200	155	058	000
130	252	255	255	255	255	255	230	175	106	078	000
017	230	255	255	255		250	220	152	101	055	000
000	059	245	255	255	255	245	204	132	081	030	000
000	000	053	223	255	255	235	200	121	063	000	000
000	000	000	011	110	208	224	187	112	054	000	000
000	000	000	000	000	000	008	015	010	000	000	000

(b)

Both targets' right side gray intensity are different in the target window. The two targets' geometric centres respectivly estimate (6.5, 6.5) and (6, 6.5). Using the moment method the obtained result are equal to (6.102094, 6.550835) and (4.64685, 6.338049). Both of the targets centres are shifted. It is clear that the size of the centre error is dependent on the image quality. The program was written by R.G. XUE and H. Rüther to find the target centre and the result was compared with the result of the ellipse fitting method.

4.3 A RADIUS OF GYRATION METHOD

In this thesis, we introduce a new method to detect the centre of the targets. The method can be named as **Radius of Gyration Method (RGM)** which can be used mainly in cases where the gray value is a distribution function $G(X_i, Y_j)$. Consider a rectangular object which has an even density. Set the width W and high H both equal to 1 and length L is arbitrary, then the rotational radius R of the object for the Y axis in the coordinate system (X, Y), as show in the Fig. 4.3-1, can be expressed as follows

$$R = \left(\frac{\int_0^V v(r)^2 \, dv}{V}\right)^{\frac{1}{2}} \qquad (4.3\text{-}1)$$

where $dv = 1 \times 1 \times dr$, $V = 1 \times 1 \times L$. *Function* 4.3-1 *can be obtain*

$$R = \sqrt{\frac{L^2}{3}} \qquad (4.3\text{-}2)$$

Where V is the volume of the object, r is the distance of an infinitesimal volume dv to the Y axis. It is clear that the position defined by R is not at the centre of the object.

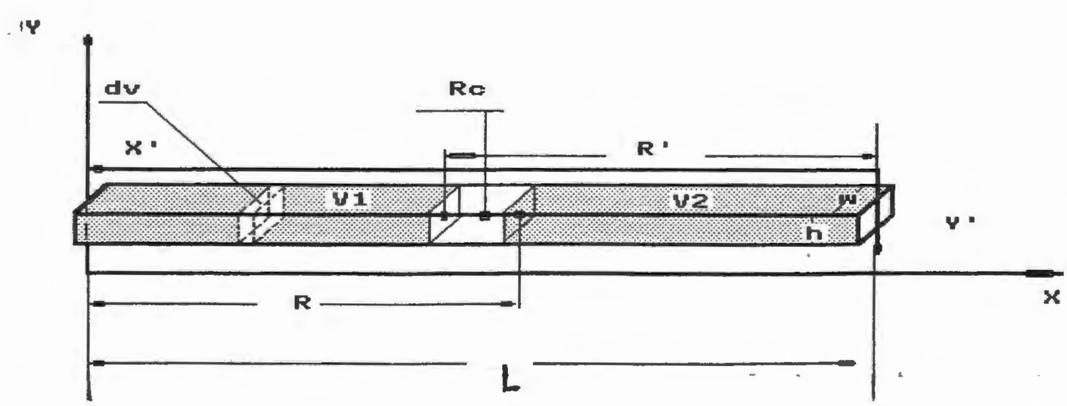


Fig. 4.3-1 The radius of gyration Method use in the even density object

(X′,Y′) is second coordinate system of the object, it will be defined an other rotational radius R′ which can be obtain by transform the coordinate system (X′, Y′) to coordinate system (X, Y). The position defined by R will be symmetrical to the position defined by R′. The centre of the object can be defined as the midpoint between the two position which are defined by R and R′ or the position can be defined by the following functions

$$\Delta X = X_R - X_{R'} \qquad (4.3\text{-}3)$$

$$X_C = X_{R'} + X_1 = \frac{\Delta X \times V_2}{V_1 + V_2} + X_{R'}$$

where V_1 is the volume of the object from the one side to X_R and V_2 is the volume of the target from the other side to X_R as shown in Fig. 4.3-1

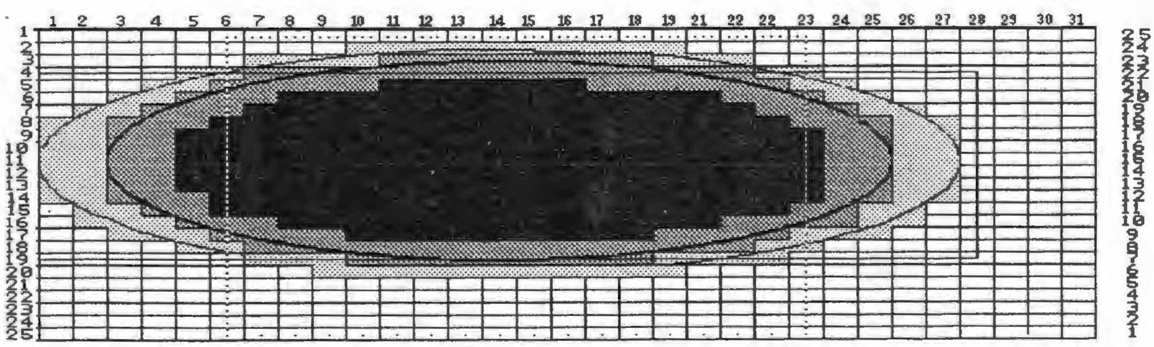


Fig 4.3-2 The Target is in Two Coorginate System

In the image processing, the volume can be expressed by the sum of the gray value of all pixel in the target window as shown in Fig. 4.3-2 and Fig. 4.3-3. The function 4.3-1 and 4.3-3 can be expressed as follows

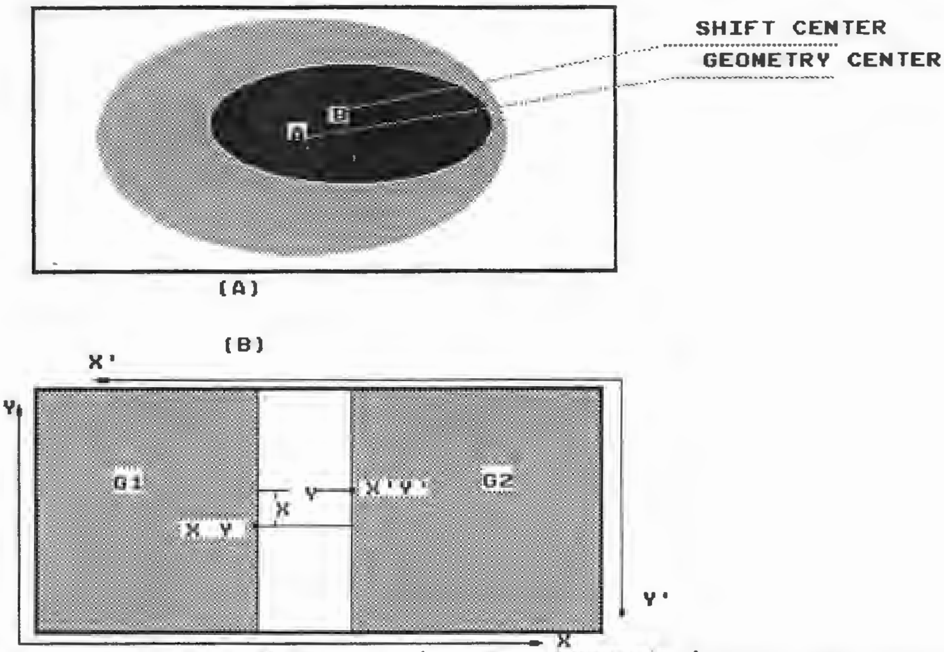


Fig. 4.3-3 The RGM use in the target image, ∆x and ∆y are difference of two radiuses position

$$X_{r1} = \sqrt{\frac{\sum_{i=1}^{n}\sum_{j=1}^{m} G(X_i, Y_j)\, X_i^2}{\sum_{i=1}^{n}\sum_{j=1}^{m} G(X_i, Y_J)}} \qquad Y_{r1} = \sqrt{\frac{\sum_{i=1}^{n}\sum_{j=1}^{m} G(X_i, Y_j)\, Y_j^2}{\sum_{i=1}^{n}\sum_{j=1}^{m} G(X_i, Y_j)}}$$

$$X_{R2} = \sqrt{\frac{\sum_{i=1}^{n}\sum_{j=1}^{m} G(X_i, Y_j)\, X_i^2}{\sum_{i=1}^{n}\sum_{j=1}^{m} G(X_i, Y_j)}} \qquad Y_{r2} = \sqrt{\frac{\sum_{i=1}^{n}\sum_{j=1}^{m} G(X_i, Y_j)\, Y_j^2}{\sum_{i=1}^{n}\sum_{j=1}^{m} G(X_i, Y_j)}}$$

(4.3-4)

The centre of the target can be obtained from the following function

$$X_c = \frac{X_{r1} + X_{r2}}{2}$$

$$Y_c = \frac{Y_{r1} + Y_{r2}}{2} \qquad (4.3\text{-}5)$$

or also can expressed as

$$X_C = X_{r2} + X_1 = \frac{\Delta X \sum_{i=x_{r1}}^{n} \sum_{j=1}^{m} G_2(X_i, Y_j)}{\sum_{i=1}^{x_{r2}} \sum_{j=1}^{m} G_1(X_i, Y_j) + \sum_{i=x_{r1}}^{n} \sum_{j=1}^{m} G_2(X_i, Y_j)} + X_{r2}$$

$$Y_C = Y_{r2} + Y_1 = \frac{\Delta Y \sum_{i=1}^{n} \sum_{j=y_{r1}}^{m} G_4(X_i, Y_j)}{\sum_{i=1}^{n} \sum_{j=1}^{Y_{R2}} G_3(X_i, Y_j) + \sum_{i=1}^{n} \sum_{j=y_{r1}}^{m} G_4(X_i, Y_j)} + Y_{r2}$$

(4.3-6)

function 4.3-6 can be get a result which will be equivalent with MCGM result. Where G_1, G_2, G_3 and G_4 are sums of the pixels of the target respectively in the image as shown Fig. 4.3-3. If the gray level is even and symmetrically distributed in the window of the target, the perfect result as shown in the following examples would be obtained.

```
0 0 0 0 0 0 0 0 0
0 5 5 5 5 5 5 5 0
0 5 5 5 5 5 5 5 0
0 5 5 5 5 5 5 5 0
0 5 5 5 5 5 5 5 0
0 5 5 5 5 5 5 5 0
0 5 5 5 5 5 5 5 0
0 5 5 5 5 5 5 5 0
0 0 0 0 0 0 0 0 0

THE MOMENT METHOD
Xc = 5
Yc = 5

THEE CRR METHOD
Xo = 5
Yo = 5
```

(a)

```
0 0 0 0 0 0 0 0 0
0 5 3 4 9 2 5 1 0
0 5 3 4 9 2 5 1 0
0 5 3 4 9 2 5 1 0
0 5 3 4 9 2 5 1 0
0 5 3 4 9 2 5 1 0
0 5 3 4 9 2 5 1 0
0 5 3 4 9 2 5 1 0
0 0 0 0 0 0 0 0 0

THE MOMENT METHOD
Xc = 4.655172
Yc = 5

THE CRR METHOD
Xo = 4.674555
Yo = 5
```

(B)

```
0 0 0 0 0 0 0 0 0
0 1 3 4 9 2 5 1 0
0 1 3 4 9 2 5 1 0
0 1 3 4 9 2 5 1 0
0 1 3 4 9 2 5 1 0
0 1 3 4 9 2 5 1 0
0 1 3 4 9 2 5 1 0
0 1 3 4 9 2 5 1 0
0 0 0 0 0 0 0 0 0

THE MOMENT METHOD
Xc = 5.08
Yc = 5

THE CRR METHOD
Xc = 5.076648
Yc = 5
```

(C)

from these examples, it can be shown that the correct centre position can be obtained by both of the methods in example (a), while examples (b) and (c), the RGM method can point out slightly better result. The same size window is used in these three examples. If we can define perfect edge positions of the symmetrical object, then we can set gray level of the pixels in the same accuracy, regardless of the targets gray scale distribution. That means the target is set with an even gray scale density. But in the real case, we can not clearly know which subpixel is the edge of the object. The basic knowledge is that the moment method result accuracy for the centre of target in digital image depending on the gray scale distribution state. The gray scale distribution in the target window is existed more even, the result of the centre of the target will be more accuracy. However, the RG method is more independent on the gray scale distribution state although it is still relating with the gray scale distribution state. Using example in section 4.2, the moment method's example (a) can be used to RGM method to get the result X_c = 6.133707, Y_c = 6.547373. It is clear that the RGM's method is better than the Moment Method in the subpixel. The RG method has a problem, ie, it is necessary to discuss how to choose the target window size and the corresponding position which will influence the computation accuracy. If the window is chosen as below:

0	0	0	0	0	0	0	0	0
5	5	5	5	5	5	5	0	0
5	5	5	5	5	5	5	0	0
5	5	5	5	5	5	5	0	0
5	5	5	5	5	5	5	0	0
5	5	5	5	5	5	5	0	0
5	5	5	5	5	5	5	0	0
5	5	5	5	5	5	5	0	0
0	0	0	0	0	0	0	0	0

The RGM can be obtained the centre position as

X_c = 4.07379,
Y_c = 5

The centre of the target is shifted a little. If the window can be chosen such that the bordering frame around the target is symmetrical around the target, then, this RG method is better than the MOMENT method, regardless of the targets gray scale distribution. If the gray scale distribution of the target in the image has seriously deviated from the one side of the target then the RG method can give a quite satisfactory result which is better than the moment method.

0 0 0 0 0 0 0	0 0 0 0 0 0 0	0 0 0 0 0 0 0
0 1 4 3 6 0 0	0 1 4 3 6 7 0	0 1 4 3 6 7 8
0 1 4 3 6 0 0	0 1 4 3 6 7 0	0 1 4 3 6 7 8
0 1 4 3 6 0 0	0 1 4 3 6 7 0	0 1 4 3 6 7 8
0 1 4 3 6 0 0	0 1 4 3 6 7 0	0 1 4 3 6 7 8
0 1 4 3 6 0 0	0 1 4 3 6 7 0	0 1 4 3 6 7 8
0 1 4 3 6 0 0	0 1 4 3 6 7 0	0 1 4 3 6 7 8
0 0 0 0 0 0 0	0 0 0 0 0 0 0	0 0 0 0 0 0 0
THE MOMENT METHOD	THE MOMENT METHOD	THE MOMENT METHOD
Xc = 4	Xc = 4.66667	Xc = 5.310345
Yc = 4.5	Yc = 4.5	Yc = 4.5
THE CRR METHOD	THE CRR METHOD	THE CRR METHOD
Xc = 4	Xc = 4.635716	Xc = 5.220533
Yc = 4.5	Yc = 4.5	Yc = 4.5

CHAPTER FIVE IMAGE PROCESSING OF TARGET WINDOW

5.1 INTRODUCTION

The **BESTFITTING METHOD**, the **MOMENT METHOD** and the **RGM** method are accurate methods to detect the centre of targets. The results of these methods are dependent on the image gray scale distribution. Considering the target gray level shift and shape and size distortion on the original image, we try to use some image processing functions to process the image before detecting the centre of the target. These image functions include Averaging method, Sobel method, Sharpening and Median method. Some of the image processing functions, such as the Median method and Averaging method can smooth the image which may solve the noise problem. Some of the image processing functions are useful primarily as enhancement techniques for highlighting edges in the image. A clear object edge will be easily detected. Some of the image processing functions use image segmentation for edge detection.

5.2 SPATIAL-DOMAIN

The term spatial domain refers to the aggregate of pixels composing an image, and spatial-domain procedures are procedures that operate directly on these pixels. A typical image processing functions in the spatial domain may be expressed as

$$\begin{aligned} g(x,y)=T[f(x,y)]=&w_1 f(x-1,y-1)+w_2 f(x-1,y)+w_3 f(x-1,y+1)\\ &+w_4 f(x,y-1)+w_5 f(x,y)+w_6 f(x,y+1)+w_7 f(x+1,y-1)\\ &+w_8 f(x+1,y)+w_9 f(x+1,y+1) \end{aligned} \qquad (5.2\text{-}1)$$

where f(x,y) is the input image, g(x,y) is the processed image, w_i is a weight and T is an operator on f, defined over some neighbourhood of (x,y). A square or rectangular subimage area centred at (x, y), as shown in Fig. 5.1, can be used for such a neighbourhood.

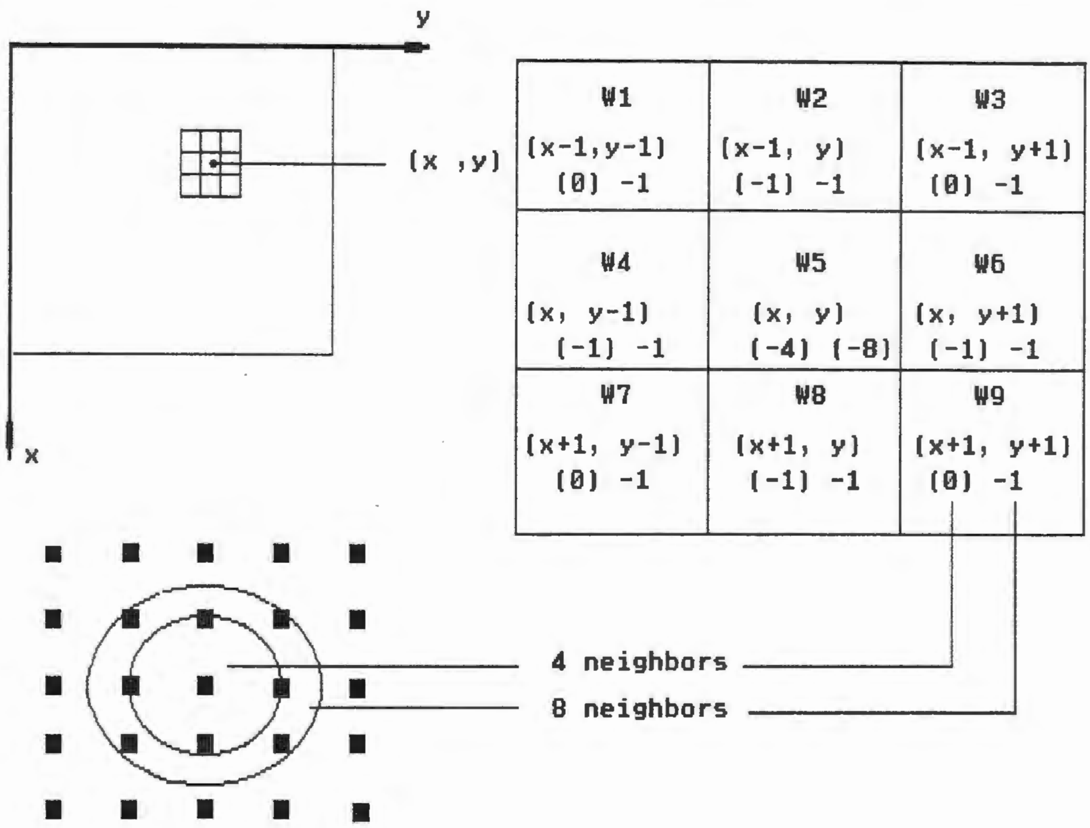


Fig. 5.2-1 3x3 pixels 4-neighbours mask and 8-neighbours mask

The centre of the subimage is moved from pixel to pixel starting, say, at the top left corner, and applying the operator at each location (x, y) to yield the value of g at that location. The formulation is based on the use of so-called masks. Basically, a mask is a small (3 x 3) 2-D array, the centre of the mask is moved around the image, such as shown in Fig.5.2-1. At each pixel position in the image, we multiply every pixel that is contained within the mask area by the corresponding mask coefficient; that is, the pixel in the centre of the mask is multiplied by 8, while its 8-neighbours are multiplied by -1. The results of these nine multiplications are then summed. If all the pixels within the mask area have the same value (constant background), the sum will be zero. In this thesis, we have used masks of include 4 neighbours and 8-neighbours.

5.3 AVERAGING METHOD

Averaging methods include Neighbourhood Averaging and Averaging of Multiple Image [14] [25] [30]. The Neighbourhood Averaging Technique is used in this thesis. Neighbourhood Averaging is a straight forward spatial-domain technique for image smoothing. Given an NxN image f(x, y), the procedure is to generate a smoothed image g(x,y) whose gray level at every point (x, y) is obtained by averaging the gray-level values of the pixels of f contained in a predefined neighbourhood of (x, y). In other words, the smoothed image is obtained by using the relation

$$g(x,y)=\frac{1}{M}\sum\sum_{(n,m)\in S} f(x,y) \qquad (5.3\text{-}1)$$

for x, y = 0,1,...., N-1. S is the set of coordinates of point in the neighbourhood of the point (x, y), including (x, y) itself, and M is the total number of points in the neighbourhood. If a 3x3 neighbourhood is used, we note by comparing Eq.5.1-1 and Eq.5.2-1 that the former equation is a special case of the latter with $W_i=1/9$. For a given neighbourhood, the blurring effect produced by neighbourhood averaging can be reduced by using a thresholding procedure that is, instead of using Eq. 5.2, we form g(x,y) according to the following criterion:

$$g(x,y)=\begin{cases}\frac{1}{M}\sum\sum_{(M,N)\in S} f(m,n) & \text{if } \left| f(x,y)-\frac{1}{M}\sum\sum_{(M,N)\in S} f(m,n)\right| < T \\ f(x,y) & \text{otherwise,}\end{cases} \quad (5.3)$$

where T is a specified non-negative threshold.The motivation behind the approach is to reduce blurring by leaving unchanged regions of an image with large (compared to T) variations in gray level. Using Eq.5.3-2 can reduce the amount of edge blurring.

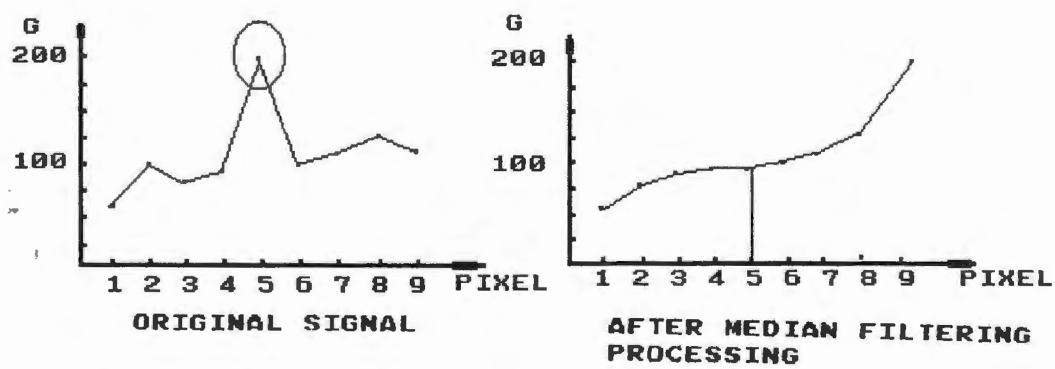


Fig.5.4-1 Median Method

5.4 MEDIAN FILTERING

Median filtering is particularly effective when the noise pattern consists of strong, spike like components and where the characteristic to be preserved is edge sharpness. Median filtering in which we replace the gray level of each pixel by the median of the gray level in a neighbourhood of that pixel, instead of by the average is used. Recall that the median m of a set of values is such that half of the values in the set are less than m and half are greater than m. In order to perform median filtering in a neighbourhood of a pixel we must first sort the values of the pixel and its neighbours, determine the median, and assign this value to the pixel. For example, if a 3x3 neighbourhood has values (65 100 85 95 200 110 100130 105), the values are sorted as (65 85 95 100 100 105110 130 200), which has a median of 100. After sorting the values the median value replaces the original value of 200. If 200 is a noise signal, then using the median filtering can cut out the noise. But if 200 is not noise, then the median filtering will reduce the image resolution.

5.5 IMAGE SHARPENING

Fig. 5.5-1 shows a waveform sharpening function in one dimension.

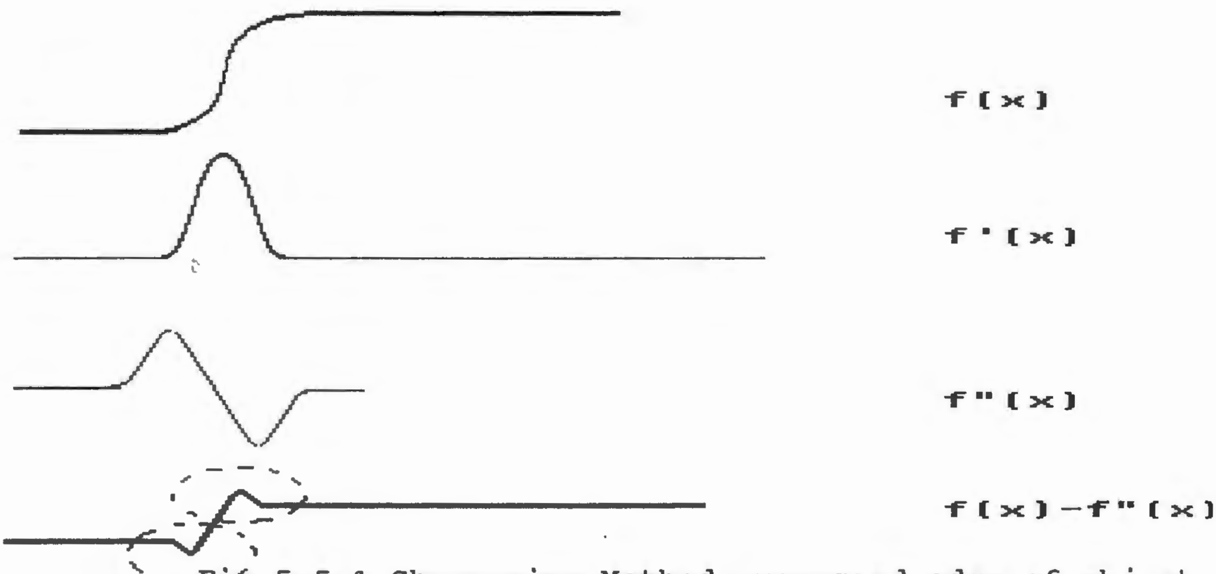


Fig.5.5-1 Sharpening Method processed edge of object

According to theory of Vector analysis, the gradient G[f(x,y)] of a differentiable function f(x,y) is defined as follow

$$\boldsymbol{G}[f(x,y)]=[\frac{\partial f}{\partial x},\ \frac{\partial f}{\partial y}] \qquad (5.5\text{-}1)$$

G[f(x,y)] is a vector and has two important characteristics, namely:

(1) the direction of maximum increasing rate of a function f(x,y) is pointed by G

(2) the quantity of gradient G, which can be written as mag[G] for simplicity, and given by

$$G[f(x,y)]=mag[\boldsymbol{G}]=[(\frac{\partial f}{\partial x})^2+(\frac{\partial f}{\partial y})^2]^{\frac{1}{2}} \qquad (5.5\text{-}2)$$

equals the maximum rate of increase of f(x, y) per unit distance in the direction of G. Equation (5.5-2) is the basis for a number of approaches to image differentiation. It is noted that this expression is in the form of a two-dimensional derivative function and that it is always positive. In practice, the scalar function G[f(x, y)] is commonly referred as the gradient of f. For a digital image, the derivatives in Eq.(5.5-2) are approximated by differences. One typical approximation is given by the relation

$$G[f(m,n)] \cong \{[f(m,n)-f(m+1,n)]^2+[f(m,n)-f(m,n+1)]^2\}^{\frac{1}{2}} \qquad (5.5\text{-}3)$$

Similar results are obtained by using absolute values, as follows:

$$G[f(m,n)] \cong |f(m,n)-f(m+1,n)|+|f(m,n)-f(m,n+1)| \qquad (5.5\text{-}4)$$

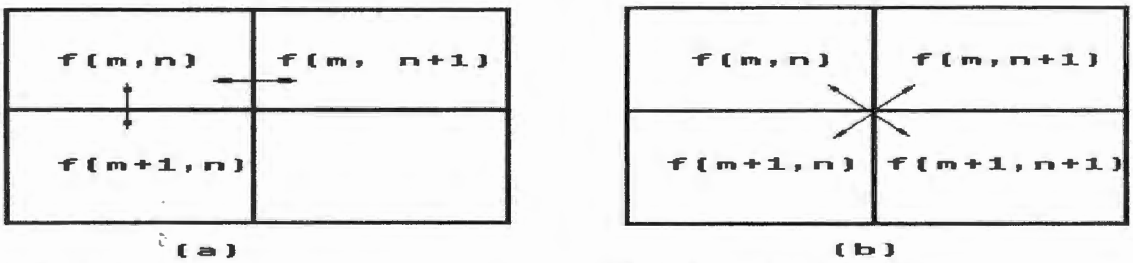
Table 5.5-1

Analog Image f(x, y)		Digital Image f(m, n)	
Differentiation		Differences	
		front	back
first	$f'_x(x,y)$ $f'_y(x,y)$	f(m+1,n)-f(m,n) f(m,n+1)-f(m,n)	f(m,n)-f(m-1,n) f(m,n)-f(m,n)
second	$f'_x{}'_x(x,y)$ $f'_y{}'_y(x,y)$	f(m+1, n) + f(m-1, n) - 2f(m, n) f(m, n+1) + f(m, n-1) - 2f(m, n)	

The relationship between pixels in Eq.(5.5-3) and Eq.(5.5-4) is shown in Fig. 5.3.

Digital image sharpening can be expressed as follows

where α is a parameter whose values can be selected values for



(a) for 2-neighbours (b) for 4-neighbours

Fig.5.5-2 A processing pixel-the neighbour pixels relation

$$G(m,n) = [1+N\alpha]\, f(m,n) - \alpha \sum_{m,n \in N}\sum f(i,j) \qquad (5.5\text{-}5)$$

sharpening the image. If α is made bigger, the image sharpening will be more powerful. N is a parameter setting the neighbourhood area. If N equal 4, the image is sharpened by a 4-neighbours mask, and if N equal 8, the mask is a 8-neighbourhood one such as

$$\mathbf{M}_1 = \begin{bmatrix} 0 & -\alpha & 0 \\ -\alpha & 1+4\alpha & -\alpha \\ 0 & -\alpha & 0 \end{bmatrix} \qquad \mathbf{M}_2 = \begin{bmatrix} -\alpha & -\alpha & -\alpha \\ -\alpha & 1+8\alpha & -\alpha \\ -\alpha & -\alpha & -\alpha \end{bmatrix}$$

where M_1 is 4-neighbours mask matrix and M_2 is 8-neighbours mask matrix

5.6 PREWITT AND SOBEL OPERATORS FOR EDGE DETECTION

The gradient of the function f at coordinates (x, y) is expressed by Eq.(5.5-2) and Eq.(5.5-4). The Prewitt operator is used to obtain a 2-D quadratic curved surface Eq (5.6-1) to fit the image function f(x,y).

$$Z(x,y)=ax^2+bxy+cy^2+dx+ey+g \quad (5.6\text{-}1)$$

The square of the difference between both functions is given by Eq.(5.6-2)

$$\varepsilon=\sum_{x,y\in a}\sum[Z(x,y)-f(x,y)]^2 \quad (5.6\text{-}2)$$

where a is a fitting area and ϵ is the square of the difference value. Set ϵ to a minimum, the differentiations with respect all the coefficients of the function Z(x,y)

$\frac{\partial\varepsilon}{\partial a},\frac{\partial\varepsilon}{\partial b},\cdots,\frac{\partial\varepsilon}{\partial g}$ are equal to zero. Through these differentiation functions we can obtain all the coefficients values of Z(x,y). The function must then be differentiated with respect to x and y. The magnitude of the gradient is expressed as Eq.(5.6-3)

$$Mag[G]=[(\frac{\partial Z}{\partial x})^2+(\frac{\partial Z}{\partial y})^2]^{\frac{1}{2}} \quad (5.6\text{-}3)$$

The Prewitt operator can check for noise and smooth the noise. If the fitting area is expanded (through enlarging the mask size), a better result will be obtained. The Sobel operator is a nonlinear computation of the edge magnitude at (m,n) defined by

where $$S(m,n)=(d_x^2+d_y^2)^{\frac{1}{2}} \quad (5.6\text{-}4)$$

$$d_x=[f(m-1,n-1)+2f(m,n-1)+f(m+1,n-1)]$$
$$-[f(m-1,n+1)+2f(m,n+1)+f(m+1,n+1)]$$

$$d_y=[f(m+1,n-1)+2f(m+1,n)+f(m+1,n+1)]$$
$$-[f(m-1,n-1)+2f(m-1,n)-f(m-1,n+1)]$$

The mask of the Sobel in vertical and horizontal directions is shown as the following

$$\begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix} \quad \begin{bmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix}$$

Using the Sobel operator to detect step edge an edge width of at least two pixels can be obtained.

5.7 HOUGH TRANSFORM

Hough [37] first introduced the Hough Transform, HT, as a method of detecting complex patterns of points in binary image data. It is a method for the fast detection of straight lines and curves in a digital image. The original HT algorithm only recognizes straight lines in the image. Duda and Hart[38] have developed the HT to detect curves in images. Ballard [9] has used the Generalised Hough Transform, GHT, to detect arbitrary shapes in images. Cowart [43] has used HT to detect unresolved target, Davies [44] has studied the Generalised Hough Transform to find ellipses. Dachs and De Jager [2] have discussed the HT filtering in Hough space to enhance or detect linear features. Olivier and De Jager[13] have used HT to detect the ellipse. Illingworth and Kittler have discussed the HT and its applications in detail. From the above studies it is clear that the HT has become increasingly popular in the past few years as a tool for image processing, computer vision, and scene understanding. The recognition of an ellipse in an image is based on the fact that all points on an ellipse should satisfy an equation,

$$f(x,y,\theta,a,b)-1=0 \qquad (5.8\text{-}1)$$

Thus two criteria need to be satisfied in order to classify an ellipse. Firstly the points must lie on a threshold, close to zero, to say it lies on an ellipse. Secondly sufficiently many points need to be found that lie on the ellipse in order to call it an ellipse. De Jager and Olivier have used HT to detect the ideal target. The algorithm was coded in a PASCAL programme, The results clearly presented that (x,y, ,a,b,) of ellipse was correctly found. The algorithm work well for single well defined ellipse. The two thresholds are very critical in the design of the algorithm and they must be made a function of the circumference of the ellipse. When more than one object is present in an image, the time taken to scan through all the five degrees of freedom becomes extremely large. The algorithm becomes too sensitive to the thresholds when more than one object is present in an image.

5.8 Results

Using Helmerts Transformation to analysis the target positions and rotation angle accuracy. The transformation used was a similarity transformation based on the well known function :
where

$$X = b\,x - a\,y + \Delta x \qquad (5.8\text{-}1)$$
$$Y = a\,x + b\,y + \Delta y$$

$$a = s \sin \theta$$
$$b = s \cos \theta$$

where the s is the scale factor and Θ is the rotation angle. x y are the object space coordinate system (x, y) before transform and X Y are transform space coordinate system. This transform is for proving the each algorithm accuracy. The mathematical pattern of Helmerts transform is used coordinate transform functions with Least-Square method to obtain the error value from the two group data which are get from one object in two times observation, measurement or computation with rotation angle and displacement. If the error value σ more small than the two group observation values more accuracy. In this thesis, we have use a twenty targets array board which has described in the chapter two, then using the PHOENICS system have snap two image frames for the target board in approach 90 degree rotation in object space. One of two images can be set a original image and another one is called rotational image. The two group real observation values is get from the mechanic method in 90 degree phase difference. Let one of two group observation values to fit another group observation values through the Helmerts transform. The error value σ is equal to 2.887. The distance between the two neighbouring target is equal 127 mm.

The follow tables are presented all of transform results of error. The error σ results in table 1 are Moment method , Ellipse Bestfitting and Circle Bestfitting method's position value of the each target centre in the target image transform to Mechanic method's observation value of these target centre positions.

Table 6.1

	M1	M2	F1	F2	F3	F4	Fc
MechM	3.913	4.105	3.79	3.8	3.806	3.815	3.79
MechM90	5.673	5.657	5.625	5.676	5.626	5.627	5.667

where

M1: is Moment method for target centre detection in target image with threshold value equal to 0.

M2: threshold value is equal to 100. Background pixels gray value are equal to 100. Moment method.

F1: is the Ellipse Bestfitting method for centre of target detection in digital image without the threshold value

F2: is the Ellipse Bestfitting method for centre of target detection in digital image under the threshold value is equal to 100. Background pixels gray value are equal to 0

F3: is the Ellipse Bestfitting method for centre of target detection in digital image under the threshold value is equal to 100. Background pixels gray value are equal to 100

F4: is the Ellipse Bestfitting method under the threshold value is equal to mean value of all pixels in the target window and Background pixels gray value are equal to mean value.

Fc: is Circle Bestfitting method without threshold.

MechM: are mean value of rotation observation value to fit original value of the target centre from the Mechanic method

MechM90: are mean value of original observation value to fit rotation observation value of the target centre from the Mechanic method

Table 2 are present error value of the original target image and rotation target image in Moment method and Ellipse Bestfitting without the threshold value after through some of image processing methods to process to fit the real target board. These image processing method involved Sharpping (Sh1) method with 4 neighbouring mask, Sharpping (Sh2) method with 8 neighbour mask, Sobel operator (Sob), Average Method (Ave) and Median Filter (Med).

Table 2

Moment

	Sh1	Sh2	Sobel	Average	Mediam
mechM	3.861	3.893	3.99	3.864	3.9
mechM90	5.685	5.698	5.717	5.683	5.688

Fitting

	Sh1	Sh2	Sobel	Average	Median
mechM	3.825	3.826	4.06	3.795	3.787
mechM90	5.74	5.735	5.768	5.663	5.693

Table 3 are present original image after through Sh1, Sh2, Sob, Ave and Med image processing to compare each other result with Moment method and Bestfitting method.

Table 3

Moment

	sh1	sh2	sob	Ave	Med
Sh1	0	0.056	0.246	0.032	0.094
Sh2		0	0.219	0.045	0.08
Sob			0	0.231	0.156
Ave				0	<u>**0.026**</u>
Med					0

Ellipse Bestfitting

	sh1	sh2	sob	Ave	Med
Sh1	0	0.082	0.993	0.157	0.078
Sh2		0	0.989	<u>**0.054**</u>	<u>**0.054**</u>
Sob			0	1.018	0.094
Ave				0	0.122
Med					0

Table 4 are present the both methods (Moment and Ellipse Bestfitting) under various image processing methods to compare each other results

Table 4

Sh1m	Sh2m	Sobm	Avem	Medm
Sh1f	Sh2f	Sobf	Avef	Medf
<u>**0.159**</u>	0.178	0.956	0.177	0.252

6 CONCLUSIONS

The objective of the thesis was the investigation of the effect of image processing algorithms on the position of targets on digital images, as required for photogrammetric modeling in machine and computer vision. For this purpose images of typical target fields were generated and a number of image processing algorithms were implemented on these images. Three algorithms for the accurate determination of the centres of elliptical targets on these images were applied to original images as well as images modified by image processing operations and the results were compared and evaluated. The three target centring methods are Moment Method, Bestfitting Ellipse method, Radius of Gyration (GR) method.

Comparison of the results of the target centring algorithms on the original images allows the following conclusions:

1. The Bestfitting Ellipse method proved marginally more accurate than the Moment method. This technique relies on the detection of target edges followed by a bestfit of an allipse to the detected edge. The superior accuracy of this method over the centre of gravity model can be explained by the lower sensitivity of this technique to assymetric gray scale distributions in the target area.

2. If the gray scale distribution of the target image

shows substantial assymetrical deviations from the ideal elliptical shape then the GR method is more accurate than the other two methods.

After processing the images using Average, Median, Sharpening and Sobel operators to improve the quality of the images, and thus the targets, target centre calculations were repeated employing the same methods as before.

The result of this investigation showed that:

1. None of the implemented image processing methods resulted in systematic point shifts

2 Averaging results in an improved centre detection accuracy. This can be attributed to two functions of the averaging algorithm, these are the capability to remove image noise and to improve image gray intensity to be more symmetric. Subsequently averaging tends to flatten the target edge gradient and make the target image more symmetrical.

3 None of the applied image processing algorithms showed an improvement for the Bestfitting Ellipse method, as this technique relies on a high contrast edge for the edge detection component of the method, and none of the applied methods had a positive effect on the edge definition.

REFERENCES

[1] A. Beghdadi & A.Le Negrate "Contrast Enhancement Technique Based on Local Detection of Edge" Computer Vision, Graphics, and Image Processing 46, 162-174 (1989)

[2] A. Dachs & G. De Jager "Filtering in Hough Space to Enhance or Detect Linear Features" , Comsig 90, Johannesburg, 1990

[3] A. Huertas & G. Medioni "Detection of Intensity Changes with Sub-Pixel Accuracy Using Laplacian-Gaussian Masks." IEEE Trans. Pattern Analysis and Machine Intelligence, vol. PAMI-8, No. 5, Sept, 651-664 1986.

[4] A.J. Tabatabai & O.R. Mitchell " Edge Location to Subpixel Accuracy in Digital Imagery". IEEE Trans. Pattern Analysis and Machine Intelligence. 1981.

[5] A.J. Tabatabai & R.Mitchell "Edge Location to Subpixel Values in Digital Imagery." IEEE Trans. Pattern Analysis and Machine Intelligence, Vol. PAMI-6, No.2, March 1984,

[6] A.J. Tabatabai "Edge Location and Data Compression for Digital Imagery" Ph.D. dissertation, School of Elect. Engrg, Purdue University, Dec. 1981

[7] A.P. Reeves " The General Theory of Moments for Shape Analysis and the Parallel Implementation of Moment Operations. 1981. TR-EE 81-37, Purdue University, W.Lafayette, IN

[8] C. Kimme, D. Ballard & J. Sklansky "Finding Circles by an Array of Accumulators" Sheet Communications Graphics and Image Processing, 1975

[9] D.H. Ballard "Generalizing the Hough Transform to Detect Arbitrary Shapes", Pattern Recognition, Vol 13, No 2, pp 111-122, 1981

[10] D.B. Salzman "A Method of General Moments for Orienting 2D Projections of Unknown 3D Objects" Computer Vision, Graphics, and Image Processing 50, 129-156 1990

[11] E.M. Mikhail " Observations and Least Squares". IEP, New York, Distributed by University Press of America, 4720 Boston Way, Lanham, MD 20706, U.S.A 1976.

[12] E.M. Mikhail, M.L. Akey, & O.R. Mitchell "Detection and Sub-Pixel Location of Photogrammetric Targets in Digital Images". Photogrammrtric, 39, 63-83, 1984.

[13] G. De Jager & B.J. Olivier" Ellipse Detection Using the Hough Transform" Department of Electrical and Electronic Engineering, University of Cape Town. 1990

[14] H.C. Andrews & B.R. Hunt "Digital Image Restoration" Prentice-Hall. INC, Engiewood Cliffs, New Jersey 07632 P.Stucki "Advance in Digital Image Processing"1979 Plenum Press, New York

[15] H.K. Muammar & M. Nixon,"A 3-Stage Procedure for Ellipse Extraction"

[16] H. Rüther "Near Real Time Photogrammetry on a Personal Computer PHOENICS" Conference of Southern African Surveyors 1989

[17] H.Rüther & N. Parkyn "Near Real Time Photogrammetry on a Personal Computer" Photogrammetric Record, 13(75):415-422, April,1990

[18] I. Tchoukanov, R. Safaee-Rad, B. Benhabib and K.C. Smith, "Sub-Pixel Edge Detection for Accurate Estimation of Elliptical Shape Parameters." CSME Mechanical Engineering Forum 1990, Toronto, Ontario, Vol. III, pp. 313-318

[19] J.B. Martens,"Deblurring Digital Images by Means of Polynomial Transforms" Computer Vision, Graphics, and

Image Processing 50, 157-176, 1990

[20] J.C. Trinder " Precision of Digital Target Location." Photogrammetric Engineering and Remote Sensing, Vol. 55, No.6, June, 1989, pp 883-886

[21] J.D. Thurgood & E.M. Mikhail " Subpixel mensuration of Photogrammetric Targets in Digital Images." Tech. Rep, School of Civil Engineering, Purdue University, No. CE-PH-82-2, 1982,

[21] J. Illingworth & J. Kittler "A Survey of the Hough Transform" Computer Vision, Graphics, and Image Processing 44, 87-116 (1988)

[22] J.S. Huang & D.H. Tseng "Statistical Theory of Edge Detection" Computer Vision, Graphics, and Image Processing 43, 337-346 1988

[23] L. Hertz & R.W. Schafer "Multilevel Threshoulding Using Edge Matchimng" Computer Vision, Graphics, and Image Processing 44, 279-295 (1988)

[24] M. Nagao & T.Matsuyama "A Structural Analysis of Complex Aerial Photographs"1980 Plenum press, New York

[25] R.C. Gonzalez & P. Wintz "Digital Image Processing" Second Edition, 1987

[26] R. Ehrlich & B. Weinberg "An Exact Method for Characterization of Grain Shape" Journal of Sedimentary Petrology, Vol 40, No.1, PP 205-212

[27] R. Safaee-Rad, I. Tchoukanov, B. Benhabib and K.C. Smith, "Accurate Parameter Estimation of Quadratic Curves from Grey-Level Images" CVGIP: Image Understanding, Vol. 54. No.2, September, pp. 259-274, 1991

[28] S.D. Yanowitz & A.M. Bruckstein "A New Method for Image Segmentation", Computer Vision, Graphics, and Image Processing, 46, 82-95 1989

[29] S. K. Ghosh "Analytical Photogrammetry" Printed in Great Britain by A.Wheaton & Co. Ltd, Exeter 1988

[30] S.Y. Xu & Y.H. Zhou "Digital Image Processing" etc.Electronics Industry Press 1989, China

[31] T. Nagata, H. Tamura and K. Ishihashi "Detection of an Ellipse by Use of a Recursive Least-Square Estimator. J. Robotic System 2, 1985 163-177

[32] T.W. Hill "On determining a distribution function known only by it's moments and/or moment generating function" Ph.D. dissertation, Arizona State University

[33] V. Torre & T.A. Poggio "On Edge Detection" IEEE

Transactions on Pattern Analysis and Machine Intelligence, Vol. PAMI-8, No. 2, March 1986

[34] W.H. Tsai "Moment-Preserving Thresholding: A New Approach" Computer Vision, Graphics, and Image Processing 29, 377-393 1985

[35] Wiley "Method for Statistical Data Analysis of Multivariate Observations, New York, 1977

[36] P. Mansbach "Calibration of a Camera and Light Source by Fitting to a Physical Model" Computer Vision, Greapics, and Image Processing 35, 200-219 1986

[37] P.V.C. Hough "Method and means for recognizing compiex patterns, U.S. Patent 3069654 1962

[38] R.O. Dutda & P.E. Hart "Use of the Hough transform to detect lines and curves in pictures" Communs Ass Comput. Mach 15,11-15 1975

[39] C.Kimme, D.H. Ballard & J. Sklansky "Finding circles by an array of accumulators, Communs Ass, Comput. Mach 18, 120- 122(1975)

[40] S.D. Shapiro "Transformation for the Computer Detection of Curves in Noise Pictures" Computers Vision, graphics, and Image Processing 4, 328-338 1975

[41] S. Tsuji & F. Matsumoto "Detection of Ellipic and Linear Edge by Searching two Parameter Spaces" Proc.5th Int. Joint Conf. Artificial Intelligence, Vol. 2, pp. 700-705, Cambridge, MA, August 1977

[42] J. Sklansky "On the Hough transform technique for Curve detection" IEEE Trans, Comput, C27, 923-926 1978

[43] A.E. Cowart, W.E. Snyder & W.H. Ruedger " The Detection of Unresolved Targets Using the Hough Transform" CVGIP 21, 222-238 1983

[44] L.S. Davis "Hierarchical Generalized Hough Transforms and Segment Based Generalized Hough Transforms" PR 15, 227-285, 1983

APPENDIX A

The follow program is for the target centre detection.

```
*****************************************************************
*                BESTFITTING OF ELLIPSE                         *
*                WRITTEN IN TRUE BASIC                          *
*                IN     1991                                    *
*                R.G. XUE                                       *
*                H. Rüther                                      *
*                UNIVERSITY OF CAPE TOWN                        *
*                DEPARTMENT OF SURVEYING ENGINEERING &          *
*                DEPARTMENT OF ELECTRONICS ENGINEERING          *
*****************************************************************

!DIM A2(100,8)
!OPEN    #6:NAME    "c:\hello\POSITION.TRU",ACCESS    INPUT,
     ORGANIZATION TEXT,CREATE NEWOLD
!LET NO = 0!DO WHILE MORE #6
!   INPUT #6:A$!LET NO = NO + 1
!   LET P3 = POS(A$,"  ")
!   LET NUM = VAL(A$(1:P3))
!   FOR I = 1 TO 7
!       LET A$ = (A$(p3+1:MAXNUM))
!       LET A$ = TRIM$(A$)
!       LET P3 = POS(A$," ")
!       LET Tem = VAL(A$(1:P3))
!       LET A2(NO,I)=TEM!   NEXT I
!   LET A2(NO,8) =  VAL(A$(P3+1:MAXNUM))
!LOOP
!mat redim a2(no,8)
!DIM WX(1)
!DIM WY(1)
!DIM WD(1)
!DIM WR(1)
!MAT REDIM WX(NO)
!MAT REDIM WY(NO)
!MAT REDIM WD(NO)
!MAT REDIM WR(NO)
!FOR I = 1 TO NO
!    LET WX(I) = A2(I,1)
!    LET WY(I) = A2(I,2)
!    LET WD(I) = A2(I,5)
!    LET WR(I) = A2(I,6)
!NEXT I
!FOR E = 1 TO NO
!LET IT = 0
!    LET X    = WX(E)
!    LET Y    = WY(E)
!    LET DX   = WD(E)
!    LET ROWS = WR(E)
!    LET X    = X
!    LET Y    = Y
!    LET DX   = DX
!    LET ROWS = ROWS
!    DIM WINDOW(1,1)
```

```
!     DIM IM(1,1)
!     DIM LX(1)
!     DIM LY(1)
!     DIM POINT(1,1)
!     LET AN$  = "Y"
!     LET PAN$ = "N"
!     LET RSTART = (Y)*512 + X
!     MAT REDIM WINDOW(DX,ROWS)
!     LET R = RSTART!     OPEN #1 : NAME "c:\pip\ikytr.", ACCESS
         INPUT, ORGANIZATION BYTE
!     SET #1: RECSIZE 1
!     SET #1: RECORD RSTART
!     OPEN #3 : NAME "c:\hello\WINDOW.DAT", ACCESS OUTPUT,
        ORGANIZATION TEXT, CREATE NEWOLD
!     ERASE #3
!     LET I = 0
!     LET FORM$ = "####"
!     IF UCASE$(PAN$) = "Y" THEN
         OPEN #2 : PRINTER
         PRINT #2 :
         PRINT #2 :
!        PRINT #2 : "Image name        : "; FILE$
!        PRINT #2 : "Top left corner : x ="; X; "    y ="; Y
!        PRINT #2 :
!        DO WHILE R < RSTART + ROWS*512
!           FOR J = 1 TO DX
!               READ #1: A$
!               PRINT #2, USING FORM$ : ORD(A$);
!               PRINT USING FORM$ : ORD(A$);
!           NEXT J
!           LET R = R+512
!           SET #1: RECORD R
!        LOOP
!     ELSE
!        LET COUNT = 1
        DO WHILE R < RSTART + ROWS*512
           FOR J = 1 TO DX
               READ #1: A$
               LET WINDOW(J,COUNT)=ORD(A$)
               PRINT #3:WINDOW(J,COUNT)
!           NEXT J
!           LET COUNT = COUNT + 1
            LET R = R+512
!           SET #1: RECORD R
!        LOOP
!     END IF
!     CLOSE #1
!     CLOSE #2
!     CLOSE #3
!     OPEN #3: NAME "c:\hello\WINDOW.DAT", ORGANIZATION TEXT,
        CREATE NEWOLD
      MAT REDIM IM(DX,ROWS)
      MAT REDIM LX(DX)
!     MAT REDIM LY(ROWS)
!     MAT REDIM POINT(2*(DX+ROWS),2)
!     MAT REDIM WINDOW(ROWS,DX)
```

```
!     MAT WINDOW = 0
!     MAT LY = 0
!     MAT LX = 0
!     FOR J = 1  TO ROWS
!         FOR I = 1 TO DX
               INPUT #3:A$
        LET WINDOW(J,I)=VAL(A$)
          NEXT I
!     NEXT J
!     FOR J = 1 TO ROWS
         FOR I = 1 TO DX - 1
!              PRINT WINDOW(J,I);"  ";
!          NEXT I
!          PRINT WINDOW(J,DX)
!     NEXT J
!     FOR J = 1 TO ROWS
!         FOR I = 1 TO DX
!             LET LX(I) = WINDOW(J,I)
!            NEXT I
!            DIM D1(1)
!            MAT REDIM D1(ROWS)
!            LET SUMX = 0
!            LET  SUMY = 0
!            FOR I = 1 TO DX
!            IF LX(I) > 0 THEN
!            LET SUMX = SUMX + LX(I)
!            END IF
!            NEXT I
!            LET D1(J) = SUMX/DX!PRINT D1(J)
!      NEXT J
!FOR J = 1 TO ROWS
!  LET SUMY = SUMY + D1(J)
!   NEXT  J
!   LET D2 = SUMY/ROWS
!   LET D2  = ((D2 + 21)/2)+0.99
!   FOR J = 1 TO ROWS
!     FOR I = 1 TO DX
!      IF WINDOW(J,I) < D2 THEN
!      LET WINDOW(J,I) = D2
!     END IF
!  NEXT I
!  NEXT J
  FOR J = 1 TO ROWS!FOR I = 1 TO  DX
!            LET LX(I) = WINDOW(J,I)
!           NEXT I
!           DIM D3(1),D5(1)
!           MAT REDIM D3(ROWS),D5(ROWS)
!           LET SUMX1 = 0
            FOR I = 1 TO DX
!           LET SUMX1 = SUMX1 + LX(I)
!           NEXT I
!           LET D3(J) = SUMX1/DX
!           IF D3(J) - D2 >  1 THEN
!           LET D5(J) = D3(J)
!           ELSE
!            LET D5(J) = 0
```

```
!          END IF
!          NEXT J
!FOR I = 1 TO DX
!FOR J = 1 TO  ROWS
!          LET LY(J) = WINDOW(J,I)
!          NEXT J
!          DIM D4(1),D6(1)
           MAT REDIM D4(DX),D6(DX)
           LET SUMY1 = 0
           FOR J = 1 TO ROWS
           LET SUMY1 = SUMY1 + LY(J)
           NEXT J
!          LET D4(I) = SUMY1/ROWS
!          IF D4(I) - D2 >  1 THEN
!          LET D6(I) = D4(I)
!          ELSE
!           LET D6(I) = 0
!          END IF
!          NEXT I
!     LET XLOW  = 1.00
!     LET XHIGH = DX
      LET YLOW  = 1.00
      LET yhigh = ROWS
!     PRINT # 0 , Using  " Hor ### ###   Ver ###
          ###":xlow,xhigh,ylow,yhigh!MAT POINT = 0
!     LET k = -1
!FOR J = 3 TO YHIGH - 2!IF D5(J)  > 0.1 THEN
!         FOR R = 1 to XHIGH        !.....store the row in l( )
!             LET LX(R) =  WINDOW(J,R)
!         NEXT R
!ELSE
!         MAT LX = D2
!END IF
!         LET K = K + 2
!         LET SUM1  = 0
!         LET SUM2  = 0
!         LET SUM3  = 0
!         LET SUM12 = 0
    FOR R = 1 TO XHIGH
!         IF LX(R) > 0 THEN
!          LET SUM1   = SUM1  + LX(R)
!          LET SUM2   = SUM2  + LX(R)^2
!          LET SUM3   = SUM3  + LX(R)^3
!           LET SUM12  = SUM12 + (R-0.5) * LX(R)
!        END IF
!    NEXT R
!    LET M1  = SUM1
!    LET M2  = SUM2
!    LET M3  = SUM3
!    LET M12 = SUM12
!    LET M1  = M1 / DX
! LET M2  = M2 / DX
! LET M3  = M3 / DX
! LET M12 = M12/ DX
! IF ABS(M2 -M1) > 0 THEN
!    LET sigma = ABS((m2 - m1^2))^0.5
```

```
!    IF SIGMA > 0 THEN
!      LET c      = ( 3 * m1 * m2 - 2 * m1^3 - m3) /
!           sigma^3
!      LET alfa   = c / (4 + c^2)^0.5
!      IF ABS(alfa) = 1 THEN
!          LET alfa=0.99
!     ELSE
!    LET h1     = m1 - ( (1 + alfa) / (1 - alfa) )^0.5 *
!      sigma      !...H1 IS A BRIGHTNESS VALUE
!    LET h2     = m1 + ( (1 - alfa) / (1 + alfa) )^0.5 *
!       sigma      !... H2 IS ANOTHER BRIGHTNESS VALUE
!     LET x2     = ( (1 + alfa) * DX/2 + 2*(2*m12 -
!        DX*h1) / (1 + alfa) / (h2 - h1) )/2
!      LET x1     = x2 - (1 + alfa) * DX/2
!      LET point(k,1)    = x1     !......store edges in
!         array point( , )
!       LET point(k,2)    = j-0.5
!       LET point(k+1,1) = x2
!       LET point(k+1,2) = j-0.5
       END IF
!    ELSE
!               LET K = K - 2
!            END  IF
!         END IF
!     NEXT J
!     FOR I = 2 to XHIGH - 2              !......for each column
                                                 target, locate
                                                 edges
!      IF D6(I)  > 0.1 THEN
!         FOR r = 1 to YHIGH              !.....store the column
                                                in l( )
!             LET LY(R) = WINDOW(R,I)
!         NEXT r
!         ELSE
!           MAT LY = D2
!         END IF
!         LET  k = k + 2
!         LET SUM1  = 0
!         LET SUM2  = 0
          LET SUM3  = 0
!         LET SUM12 = 0
!         FOR R = 1 TO YHIGH
!             IF LY(R) > 0 THEN
!                LET SUM1   = SUM1  + LY(R)
!                LET SUM2   = SUM2  + LY(R)^2
!                LET SUM3   = SUM3  + LY(R)^3
!                LET SUM12  = SUM12 + (R-0.5) * LY(R)
!             END IF
!         NEXT R
!         LET M1  = SUM1
!         LET M2  = SUM2
!         LET M3  = SUM3
!         LET M12 = SUM12
!         LET M1  = M1 / ROWS
!         LET M2  = M2 / ROWS
!         LET M3  = M3 / ROWS
```

```
!          LET M12 = M12/ ROWS
!          IF ABS(M2-M12) > 0 then
!             LET sigma = ABS((m2 - m1^2))^0.5
!             IF SIGMA > 0 THEN
!                LET c      = ( 3 * m1 * m2 - 2 * m1^3 - m3) /
                   sigma^3
!                LET alfa   = c / (4 + c^2)^0.5
               IF alfa = 1 THEN
               LET ALFA = 0.99
               ELSE
               LET h1     = m1 - ( (1 + alfa) / (1 - alfa) )^0.5
                  sigma     !...H1 IS A BRIGHTNESS VALUE
!               LET h2     = m1 + ( (1 - alfa) / (1 + alfa) )^0.5
                  sigma     !...H2 IS ANOTHER BRIGHTNESS VALUE
!                LET x2     = ( (1 + alfa) * ROWS/2 + 2*(2*m12 -
                      ROWS*h1) / (1 + alfa) / (h2 - h1) )/2
!                LET x1     = x2 - (1 + alfa) * ROWS/2
!                LET point(k,1)    = I-0.5    !....store edges
                                               in array point( , )
!                LET point(k,2)    = x1
!                LET point(k+1,1) = I-0.5
!                LET point(k+1,2) = x2
!              END IF
!             ELSE
!                LET k = k - 2
!             END IF
!          END IF
!      NEXT I
!      LET numpoint = k - 1
!      DIM F3(1)
!      DIM F1(1)
!      MAT REDIM F1(2*(DX+ROWS))
!      MAT REDIM F3(2*(DX+ROWS))
!MAT F1 = 0
!MAT F3 = 0
!      FOR Z = 1 TO 2*(DX+ROWS)
!          LET F1(Z) = POINT(Z,2)
!      NEXT Z
!      FOR Z =1 TO 2*(DX+ROWS)
!          LET F3(Z) = POINT(Z,1)
!      NEXT Z
!      PRINT # PR :"BEST FITTING ELLIPSE"
!      IF ABS( F1(I)) > 0 THEN
!        IF ABS(F3(I)) > 0 THEN
!      DIM A(1,1),B(1,1),L1(1),N1(5,5),C1(5),P(1,1),U(1),V(1),
          P2(1,1),Q(1,1),M(1,1)
!      LET enough = 100
!      LET tb = 0.001
!      LET Np = DX+ROWS
!      LET Nu = 5
!      FOR I = 1 to Np
!      NEXT I
!      MAT REDIM A(Np,5),B(5,Np),L1(Np),P(Np,Np),U(5),V(Np),
          P2(5,Np)
     MAT P=0
     CALL Prov
```

```
    PRINT # Pr, USING "###   ####.###  ####.### #####.###
                ####.###  #####.######":IT,X0,Y0,A1,B1,Phi
!    CALL L2_NORM
!    PRINT # Pr, USING "###   ####.###  ####.### #####.###
                ####.###  #####.######":IT,X0,Y0,A1,B1,Phi
!    LET Fin = 0
!    LET I1 = 0
!    DO until  I1 = Nu  or  Fin = 1
!       LET I1 = I1 +1
!       IF ABS(U(I1)) > TB  THEN
!          LET IT = IT+1
!          IF IT >Enough THEN
!             PRINT # 0, USING "Program run abandoned after ##
               Iterations": Enough
!             PRINT # 0, USING "for IMAGE ##":Im(I,J)
!             SOUND 300,.4
!             LET Fin = 1
!          END IF
!          LET I1 = 0
          PRINT # Pr
          PRINT # Pr       :"          X0        Y0        A
                  B         Phi     "
          CALL L2_NORM
         PRINT # Pr, USING "###   ####.###  ####.### #####.###
                ####.###  #####.######":IT,X0,Y0,A1,B1,Phi
      END IF
   LOOP
   PRINT # Pr: " "
   PRINT # Pr: "Final Results "
!    PRINT # Pr, USING "###   ####.###  ####.### #####.###
                   ####.########.######":IT,X0,
                   Y0,A1,B1,Phi*180/pi
!    MAT V= A * U
!    MAT V= V - L1
!    LET Sum = 0
!    FOR I=1 TO Np
!       LET Sum = Sum + V(I)^2 * P(I,I)
!    NEXT I
!    LET P3=0
!    FOR I=1 TO Np
        LET V1=V(I)*A(I,1)*P(I,I)
        LET V2=V(I)*A(I,2)*P(I,I)
        LET P3=P3+V1^2+V2^2
!        PRINT V1,V2
!    NEXT I
!   END IF
!END IF
!    DIM RES(1,1)
!    MAT REDIM RES(E,2)
!    LET RES(E,1) = X0 + X - 0.5
!    LET RES(E,2) = Y0 + Y - 0.5
!    CLOSE #3!SET WINDOW 0,X0+10,0,Y0+10!MAT PLOT POINTS:
          POINT
!PLOT POINTS:X0,Y0
!PRINT D2
 GET KEY ZZ
```

```
!SET MODE "EGARES"
!MAT PRINT RES
!NEXT E
!FOR I = 1 TO NO
!PRINT # PR,USING " ###.### ###.###      ###.###   ###.###
  ###.###   ###.###":RES(I,1),RES(I,2),A2(I,7),A2(I,8),
  RES(I,1)-A2(I,7), RES(I,2)-A2(I,8)
!NEXT I
!OPEN #8:NAME "C:\HELLO\RESULT.DAT",ACCESS OUTPUT,ORGANIZATION
       TEXT, CREATE NEWOLD!ERASE #8
 PRINT #8       :" No       NEW Xo  NEW Yo       OLD Xo    OLD Yo
              dXo       dYo        :"
!DIM RESULT(1,1)
!MAT REDIM RESULT(NO,7)
!FOR E = 1 TO NO
!LET RESULT(E,1)=RES(E,1)
!LET RESULT(E,2)=RES(E,2)
!LET RESULT(E,3)=A2(E,7)
!LET RESULT(E,4)=A2(E,8)
!LET RESULT(E,5)=RES(E,1) - A2(E,7)
!LET RESULT(E,6)=RES(E,2) - A2(E,8)!PRINT #8,USING "       ###
          ###.###  ###.###      ###.###  ###.###      ###.###
          ###.###":E,RESULT(E,1),RESULT(E,2),RESULT(E,3),
          RESULT(E,4),RESULT(E,5),RESULT(E,6)
!NEXT E

 SUB Prov
!    LET X0 = DX/2
!    LET Y0 = ROWS/2
!    LET A1 = DX/2
!    LET B1 = ROWS/2
!    LET PHI= 0.1
!    IF ABS(A1-B1) < 0.5 then
!       LET A1 = A1 + 1
     END IF
 END SUB

!SUB L2_NORM
!    FOR I=1 TO Np
!        IF ABS(F1(I)) > 0 THEN
!           IF ABS(F3(I)) > 0 THEN
!              LET MM=(F3(I)-X0)*COS(PHI)+(F1(I)-Y0)*SIN(PHI)
!              LET NN=(F1(I)-Y0)*COS(PHI)-(F3(I)-X0)*SIN(PHI)
              LET s = sin(phi)
!             LET co= cos(phi)
!             LET A(I,1) = 2*(A1^2*NN*s - B1^2*MM*co)  !...X0
!             LET A(I,2) =-2*(B1^2*MM*s + A1^2*NN*co)  !...Y0
!             LET A(I,3) = 2* A1 * (NN^2-B1^2)         !...A
!             LET A(I,4) = 2* B1 * (MM^2-A1^2)         !...B
!             LET A(I,5) = (2*NN*MM*(B1^2-A1^2))        !...
!             LET L1(I)   =-( B1^2*MM^2 + A1^2*NN^2 -
                           A1^2*B1^2)
!...L-term
!             LET P(I,I) = 1/(A(I,1)^2+A(I,2)^2)
!          END IF
!       END IF
```

```
!    NEXT I
!    MAT B  = TRN(A)
!    MAT P2 = B * P                    !...AtP
!    MAT N1 = P2 * A                   !...AtPA
!    MAT C1  = P2 * L1                 !...AtPl
!    MAT N1=INV(N1)
!    MAT U=N1*C1
!    LET X0 =X0 +U(1)
!    LET Y0 =Y0 +U(2)
!    LET A1 =A1 +U(3)
!    LET B1 =B1 +U(4)
!    LET PHI=PHI+U(5)
!    PRINT # Pr, USING " dx   ####.###  ####.### #####.###
          ####.###  #####.######":U(1),U(2),U(3),U(4),U(5)
!END SUB
 END
```

The Moment method for centre of target detection routing is

following

```
*************************************************************
*  MOMENT METHOD FOR THE TARGET'S CENTRE DETECTION          *
*      WRITTEN IN TRUE BASIC                                *
*       ON THE 1991                                         *
*       R.G.XUE                                           * *
*************************************************************
!DIM A2(100,8)
!OPEN #6:NAME "c:\hello\pt.dat",ACCESS INPUT,ORGANIZATION
         TEXT,CREATE NEWOLD
!LET NO = 0
!DO WHILE MORE #6
!   INPUT #6:A$
!   LET NO = NO + 1
!   LET P3 = POS(A$,"  ")
!   LET NUM = VAL(A$(1:P3))
!   FOR I = 1 TO 7
!       LET A$ = (A$(p3+1:MAXNUM))
!       LET A$ = TRIM$(A$)
!       LET P3 = POS(A$," ")
!       LET Tem = VAL(A$(1:P3))
!       LET A2(NO,I)=TEM
!   NEXT I
!   LET A2(NO,8) =  VAL(A$(P3+1:MAXNUM))
!LOOP
!MAT redim a2(no,8)
!DIM WX(1)
!DIM WY(1)
!DIM WD(1)
!DIM WR(1)
!MAT REDIM WX(NO)
!MAT REDIM WY(NO)
!MAT REDIM WD(NO)
!MAT REDIM WR(NO)
```

```
!FOR I = 1 TO NO
!     LET WX(I) = A2(I,1)
!     LET WY(I) = A2(I,2)
!     LET WD(I) = A2(I,5)
!     LET WR(I) = A2(I,6)
!NEXT I
!FOR E = 1 TO NO
!     LET IT = 0
!     LET X    = WX(E)
!     LET Y    = WY(E)
!     LET DX   = WD(E)
!     LET ROWS = WR(E)
!     DIM WINDOW(1,1)
!     DIM IM(1,1)
!     DIM LX(1)
!     DIM LY(1)
!     DIM POINT(1,1)
!     LET AN$  = "Y"
!     LET PAN$ = "N"
!     LET RSTART = (Y)*512 + X + 1
!     MAT REDIM WINDOW(DX,ROWS)
!     PRINT
!     LET R = RSTART
!     OPEN #1 : NAME "a:target1.cfi", ACCESS INPUT,
                ORGANIZATION BYTE
!     SET #1: RECSIZE 1
!     SET #1: RECORD RSTART
!     OPEN #3 : NAME "c:\hello\WINDOW.DAT", ACCESS OUTPUT,

                ORGANIZATION TEXT, CREATE NEWOLD
!     ERASE #3
!     LET I = 0
!     LET FORM$ = "####"
!     IF UCASE$(PAN$) = "Y" THEN
!        OPEN #2 : PRINTER
!        PRINT #2 :
!        PRINT #2 :
!        PRINT #2 : "Image name       : "; FILE$
!        PRINT #2 : "Top left corner : x ="; X; "    y ="; Y
!        PRINT #2 :
!        DO WHILE R < RSTART + ROWS*512
!           FOR J = 1 TO DX
!               READ #1: A$
!               PRINT #2, USING FORM$ : ORD(A$);
!               PRINT USING FORM$ : ORD(A$);
!           NEXT J
!           LET R = R+512
!           SET #1: RECORD R
!        LOOP
!     ELSE
!        LET COUNT = 1
!        DO WHILE R < RSTART + ROWS*512
```

```
 !     NEXT I
 !     MAT B   = TRN(A)
 !     MAT P2 = B * P                          !...AtP
 !     MAT N1 = P2 * A                         !...AtPA
 !     MAT C1  = P2 * L1                       !...AtPl
 !     MAT N1=INV(N1)
 !     MAT U=N1*C1
 !     LET X0 =X0 +U(1)
 !     LET Y0 =Y0 +U(2)
 !     LET A1 =A1 +U(3)
 !     LET B1 =B1 +U(4)
 !     LET PHI=PHI+U(5)
 !     PRINT # Pr, USING " dx   ####.###  ####.### #####.###
            ####.###  #####.######":U(1),U(2),U(3),U(4),U(5)
 !END SUB
  END
```

The Moment method for centre of target detection routing is following

```
***************************************************************
*  MOMENT METHOD FOR THE TARGET'S CENTRE DETECTION            *
*       WRITTEN IN TRUE BASIC                                 *
*        ON THE 1991                                          *
*        R.G.XUE                                              *
***************************************************************
!DIM A2(100,8)
!OPEN #6:NAME "c:\hello\pt.dat",ACCESS INPUT,ORGANIZATION
            TEXT,CREATE NEWOLD
!LET NO = 0
!DO WHILE MORE #6
!   INPUT #6:A$
!   LET NO = NO + 1
!   LET P3 = POS(A$,"  ")
!   LET NUM = VAL(A$(1:P3))
!   FOR I = 1 TO 7
!       LET A$ = (A$(p3+1:MAXNUM))
!       LET A$ = TRIM$(A$)
!       LET P3 = POS(A$," ")
!       LET Tem = VAL(A$(1:P3))
!       LET A2(NO,I)=TEM
!   NEXT I
!   LET A2(NO,8) =  VAL(A$(P3+1:MAXNUM))
!LOOP
!MAT redim a2(no,8)
!DIM WX(1)
!DIM WY(1)
!DIM WD(1)
!DIM WR(1)
!MAT REDIM WX(NO)
!MAT REDIM WY(NO)
!MAT REDIM WD(NO)
!MAT REDIM WR(NO)
```

```
!FOR I = 1 TO NO
!     LET WX(I) = A2(I,1)
!     LET WY(I) = A2(I,2)
!     LET WD(I) = A2(I,5)
!     LET WR(I) = A2(I,6)
!NEXT I
!FOR E = 1 TO NO
!     LET IT = 0
!     LET X    = WX(E)
!     LET Y    = WY(E)
!     LET DX   = WD(E)
!     LET ROWS = WR(E)
!     DIM WINDOW(1,1)
!     DIM IM(1,1)
!     DIM LX(1)
!     DIM LY(1)
!     DIM POINT(1,1)
!     LET AN$  = "Y"
!     LET PAN$ = "N"
!     LET RSTART = (Y)*512 + X + 1
!     MAT REDIM WINDOW(DX,ROWS)
!     PRINT
!     LET R = RSTART
!     OPEN #1 : NAME "a:target1.cfi", ACCESS INPUT,
                    ORGANIZATION BYTE
!     SET #1: RECSIZE 1
!     SET #1: RECORD RSTART
!     OPEN #3 : NAME "c:\hello\WINDOW.DAT", ACCESS OUTPUT,

               ORGANIZATION TEXT, CREATE NEWOLD
!     ERASE #3
!     LET I = 0
!     LET FORM$ = "####"
!     IF UCASE$(PAN$) = "Y" THEN
!        OPEN #2 : PRINTER
!        PRINT #2 :
!        PRINT #2 :
!        PRINT #2 : "Image name      : "; FILE$
!        PRINT #2 : "Top left corner : x ="; X; "    y ="; Y
!        PRINT #2 :
!        DO WHILE R < RSTART + ROWS*512
!           FOR J = 1 TO DX
!               READ #1: A$
!               PRINT #2, USING FORM$ : ORD(A$);
!               PRINT USING FORM$ : ORD(A$);
!           NEXT J
!           LET R = R+512
!           SET #1: RECORD R
!        LOOP
!     ELSE
!        LET COUNT = 1
!        DO WHILE R < RSTART + ROWS*512
!           FOR J = 1 TO DX
!               READ #1: A$
!               LET WINDOW(J,COUNT)=ORD(A$)
!               PRINT #3:WINDOW(J,COUNT)
```

```
!           NEXT J
!           LET COUNT = COUNT + 1
!           LET R = R+512
!           SET #1: RECORD R
!        LOOP
!     END IF
!     CLOSE #1
!     CLOSE #2
!     CLOSE #3
!     OPEN #3: NAME "c:\hello\WINDOW.DAT", ORGANIZATION TEXT,
                  CREATE NEWOLD
!     MAT REDIM IM(DX,ROWS)
!     MAT REDIM LX(DX)
!     MAT REDIM LY(ROWS)
!     MAT REDIM POINT(2*(DX+ROWS),2)
!     MAT REDIM WINDOW(ROWS,DX)
!     MAT WINDOW = 0
!     MAT LY = 0
!     MAT LX = 0
!     FOR J = 1  TO ROWS
!         FOR I = 1 TO DX
!             INPUT #3:A$
!             LET WINDOW(J,I)=VAL(A$)
!         NEXT I
!     NEXT J
!     FOR J = 1 TO ROWS
!         FOR I = 1 TO DX
!             PRINT WINDOW(J,I);"  ";
!         NEXT I
!         PRINT WINDOW(J,DX)
!     NEXT J
!         DIM D1(1)
!         MAT REDIM D1(ROWS)
!         DIM M1(1)
!         MAT REDIM M1(ROWS)
!         LET SUMX = 0
!         LET SUMX1= 0
!         LET SUMY = 0
!         LET SUMY1= 0
!         DIM D2(1)
!         MAT REDIM D2(DX)
!         DIM M2(1)
!         MAT REDIM M2(DX)
!     LET SUMX2 = 0
!     LET SUMX3 = 0
!     LET SUMY2 = 0
!     LET SUMY3 = 0
!     FOR J = 1 TO ROWS
!         FOR I = 1 TO DX
!             LET LX(I) = WINDOW(J,I)
!         PRINT LX(I)
!             LET SUMX = SUMX + (LX(I) * (I-1))
!             LET SUMX1= SUMX1+ LX(I)
!         NEXT I
!         LET D1(J) = SUMX
!         LET M1(J) = SUMX1
```

```
!         LET SUMX2 = SUMX2 + D1(J)
!         LET SUMX3 = SUMX3 + M1(J)
!     NEXT J
!     LET X0 = (SUMX2/SUMX3)
!     FOR I = 1 TO DX
!         FOR J = 1 TO ROWS
!             LET LY(J) = WINDOW(J,I)
!         PRINT LY(J)
!             LET SUMY = SUMY + (LY(J) * J)
!             LET SUMY1= SUMY1 +LY(J)
!         NEXT J
!         LET D2(I) = SUMY
!         LET M2(I) = SUMY1
!         LET SUMY2 = SUMY2 + D2(I)
!         LET SUMY3 = SUMY3 + M2(I)
!     NEXT I
!     LET Y0 = (SUMY2/SUMY3)
!     DIM RES(1,1)
!     MAT REDIM RES(E,2)
!     LET RES(E,1) = X0 + X
!     LET RES(E,2) = Y0 + Y
!     CLOSE #3
!     MAT D1 = 0
!     MAT M1 = 0
!     MAT D2 = 0
!     MAT M2 = 0
!     FOR J = 1  TO ROWS
!         FOR I = 1 TO DX
!             INPUT #3:A$
!             LET WINDOW(J,I)=VAL(A$)
!         NEXT I
!     NEXT J
!     FOR J = 1 TO ROWS
!         FOR I = 1 TO DX
!             PRINT WINDOW(J,I);"  ";
!         NEXT I
!         PRINT WINDOW(J,DX)
!     NEXT J
!         DIM D1(1)
!         MAT REDIM D1(ROWS)
!         DIM M1(1)
!         MAT REDIM M1(ROWS)
!         LET SUMX = 0
!         LET SUMX1= 0
!         LET SUMY = 0
!         LET SUMY1= 0
!         DIM D2(1)
!         MAT REDIM D2(DX)
!         DIM M2(1)
!         MAT REDIM M2(DX)
!     LET SUMX2 = 0
!     LET SUMX3 = 0
!     LET SUMY2 = 0
!     LET SUMY3 = 0
!     FOR J = 1 TO ROWS
!         FOR I = 1 TO DX
```

```
!               LET LX(I) = WINDOW(J,I)
!           PRINT LX(I)
!               LET SUMX = SUMX + (LX(I) * (I-1))
!               LET SUMX1= SUMX1+ LX(I)
!           NEXT I
!           LET D1(J) = SUMX
!           LET M1(J) = SUMX1
!           LET SUMX2 = SUMX2 + D1(J)
!           LET SUMX3 = SUMX3 + M1(J)
!       NEXT J
!       LET X0 = (SUMX2/SUMX3)
!       FOR I = 1 TO DX
!           FOR J = 1 TO ROWS
!               LET LY(J) = WINDOW(J,I)
!           PRINT LY(J)
!               LET SUMY = SUMY + (LY(J) * J)
!               LET SUMY1= SUMY1 +LY(J)
!           NEXT J
!           LET D2(I) = SUMY
!           LET M2(I) = SUMY1
!           LET SUMY2 = SUMY2 + D2(I)
!           LET SUMY3 = SUMY3 + M2(I)
!       NEXT I
!       LET Y0 = (SUMY2/SUMY3)
!       DIM RES(1,1)
!       MAT REDIM RES(E,2)
!       LET RES(E,1) = X0 + X
!       LET RES(E,2) = Y0 + Y
!       CLOSE #3
!       MAT D1 = 0
!       MAT M1 = 0
!       MAT D2 = 0
!       MAT M2 = 0
!NEXT E
!FOR I = 1 TO NO
!       PRINT # PR,USING " ###.### ###.###      ###.###  ###.###
           ###.###  ###.###":RES(I,1),RES(I,2),A2(I,7), A2(I,8),
              RES(I,1)-A2(I,7),RES(I,2)-A2(I,8)
!NEXT I
!OPEN #8:NAME "C:\hello\RESULTM.DAT",ACCESS OUTPUT,
               ORGANIZATION TEXT, CREATE NEWOLD
!ERASE #8
!PRINT #8       :" No      NEW Xo  NEW Yo      OLD Xo   OLD Yo
               dXo      dYo         :"
!DIM RESULTM(1,1)
!MAT REDIM RESULTM(NO,7)
!FOR E = 1 TO NO
!       LET RESULTM(E,1)=RES(E,1)
!       LET RESULTM(E,2)=RES(E,2)
!       LET RESULTM(E,3)=A2(E,7)
!       LET RESULTM(E,4)=A2(E,8)
!       LET RESULTM(E,5)=RES(E,1) - A2(E,7)
!       LET RESULTM(E,6)=RES(E,2) - A2(E,8)
!       PRINT #8,USING " ###     ###.### ###.###     ###.###
                  ###.###     ###.###  ###.###":E,RESULTM(E,1),
                     RESULTM(E,2),RESULTM(E,3),RESULTM(E,4),
```

```
                    RESULTM(E,5),RESULTM(E,6)
!NEXT E
!END
```

Appendix B

In this thesis, all of the program have write in True Basic. We have use the PIP software package to design a drawing circle. This fill circle is design as ideal condition circular target that the gray scale values are equal to 255 in target border and gray indensity equal to 0. as in Fig C.1. The centre of the circle is easy got by hand calculation. Putting the circle image in my program, then, compare the program result with the hand calculation result, can be proved the correctness of the program. Through the compare, the programs are hundred percent correct. We have got same results.

Top of the corner: x = 141, y = 240

000	000	000	000	000	000	000	000	000	000	000	000	000	000
000	000	000	000	000	255	255	255	255	000	000	000	000	000
000	000	000	255	255	255	255	255	255	255	255	000	000	000
000	000	255	255	255	255	255	255	255	255	255	255	000	000
000	255	255	255	255	255	255	255	255	255	255	255	255	000
000	255	255	255	255	255	255	255	255	255	255	255	255	000
000	255	255	255	255	255	255c	255	255	255	255	255	255	000
000	255	255	255	255	255	255	255	255	255	255	255	255	000
000	255	252	255	255	255	255	255	255	255	255	255	255	000
000	000	255	255	255	255	255	255	255	255	255	255	000	000
000	000	000	255	255	255	255	255	255	255	255	000	000	000
000	000	000	000	000	255	255	255	255	000	000	000	000	000
000	000	000	000	000	000	000	000	000	000	000	000	000	000

center of the target are equal to x = 147.5 , y = 246

Moment method x = 147.5 Ellipse Bestfitting x = 147.5

y = 246 y = 246

This result expressed the program is complete right

Test two:

Such in chapter two for big and small displacements, the Moment method have get results as shown in table 1. If the mean difference of the displacement smaller, the method calculation will more accurecy.

Table C.1

Moment

X	Y	Δx	Δy	v_x	v_y
252.875	308.48				
252.649	308.445	0.226	0.035		
252.405	308.447	0.224	0.002	0.002	0.033
252.116	308.471	0.289	0.024	0.065	0.022
251.857	308.473	0.259	0.002	0.03	0.022
Mean Difference of the displacement				0.032	0.026

Ellipse Bestfitting

X	Y	Δx	Δy	v_x	v_y
252.604	308.251				
252.336	308.222	0.298	0.029		
252.063	308.211	0.273	0.011	0.025	0.018
251.71	308.23	0.353	0.019	0.08	0.008
251.426	308.223	0.238	0.007	0.115	0.012
Mean Difference of the displacement				0.073	0.013

where Δx, Δy are distance of the displacement. v_x, v_y are the difference value of the displement.

3	3	34	69	107	117	104	80	37	20
19	28	64	130	183	114	204	156	96	38
7	49	137	207	234	242	237	202	151	71
20	83	180	234	255	255	255	245	175	91
27	95	191	246	255	255	255	255	224	147
38	123	206	255	255	255	255	255	246	168
40	128	219	253	255	255	255	255	251	172
35	116	218	255	255	255	255	255	242	158
34	106	203	255	255	255	255	255	204	118
23	92	184	225	255	255	255	255	188	101
23	47	120	187	223	243	231	195	132	65
9	35	83	127	171	196	185	142	90	46
0	9	33	69	99	111	106	94	45	16

(a) Original target image gray scale distribution

15	29	51	73	95	103	98	81	58	35
23	49	83	118	144	156	149	126	92	57
37	72	115	158	188	200	193	167	127	82
49	92	143	188	217	229	222	198	157	107
59	100	161	206	233	244	239	218	179	127
66	118	172	215	240	249	246	228	192	141
69	122	176	218	242	251	248	231	196	145
68	119	173	216	241	249	245	227	190	139
62	110	163	206	233	243	237	215	176	125
52	94	143	186	215	226	219	195	154	105
40	74	116	155	184	196	189	164	125	82
27	52	84	117	142	154	147	125	93	59
17	32	54	73	98	106	102	85	61	38

(b)The target
image gray scale distribution changed after using average method
Processed original target

5	59	151	26	51	175	158	91	66	96	34
19	87	64	81	135	255	255	255	255	109	64
54	154	97	255	255	255	255	255	255	255	45
2	101	41	255	255	255	255	255	255	189	43
44	149	13	255	255	255	255	255	255	255	255
3	111	163	255	255	255	255	255	255	255	255
13	126	157	255	255	255	255	255	255	255	255
20	155	61	255	255	255	255	255	255	255	234
9	99	49	255	255	255	255	255	255	255	15
94	131	88	255	255	255	255	255	255	255	103
77	35	146	100	255	255	255	255	255	106	43
74	66	9	120	158	255	255	255	200	75	15
50	90	122	91	4	77	79	76	153	71	125

(c) Using shapping image processing
method processed the original target image by 8-neighboors mask