

56547

ESTIMATION OF MOBILE ROBOT LOCATION BY USING A SINGLE
CCD CAMERA

A THESIS SUBMITTED TO
THE GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES
OF
THE MIDDLE EAST TECHNICAL UNIVERSITY

BY

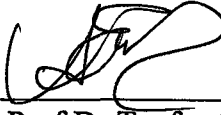
KAĞAN ÖZDEMİR

56547


IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR THE DEGREE OF MASTER OF SCIENCE
IN
THE DEPARTMENT OF MECHANICAL ENGINEERING

SEPTEMBER 1996


Approval of the Graduate School of Natural and Applied Sciences


Prof. Dr. Tayfur ÖZTÜRK
Director

I certify that this thesis satisfies all the requirements as a thesis for the degree of Master of Science.


Prof. Dr. Ediz PAYKOÇ
Head of Department

This is to certify that we have read this thesis and that in our opinion it is fully adequate, in scope and quality, as a thesis for the degree of Master of Science.

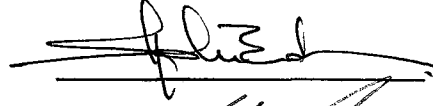

Prof. Dr. Abdülkadir ERDEN
Supervisor

Examining Committee Members:

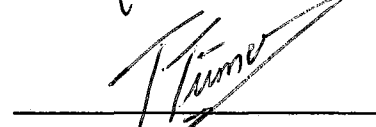
Prof. Dr. Metin AKKÖK



Prof. Dr. Abdülkadir ERDEN



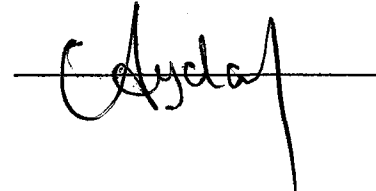
Prof. Dr. Turgut TÜMER



Prof. Dr. Faruk ARINÇ



Assoc. Prof. Dr. Aydan ERKMEN



ABSTRACT

ESTIMATION OF MOBILE ROBOT LOCATION BY USING A SINGLE CCD CAMERA

ÖZDEMİR, Kağan

M.S., Department of Mechanical Engineering

Supervisor : Prof. Dr. Abdülkadir ERDEN

September 1996, 94 pages

In this work, it is aimed to calculate the position of a mobile robot with respect to a door at indoor environment. A single CCD camera is used and a camera viewing parameters calculated as a first step. Newton Raphson Method and Liu and Huang's method and their modified form are used for this purpose. The implementation software is written in C++ programming language under DOS operating system. Extensive number of experiments are performed to test the performance of the algorithms.

Keywords: Mobile Robots, Computer Vision, Robot Vision, Perception of the Environment, Distance Estimation.

ÖZ

TEK CCD KAMERA KULLANARAK BİR GEZER ROBOTUN KONUMUNUN BELİRLENMESİ

ÖZDEMİR, Kağan

Yüksek Lisans, Makina Mühendisliği Bölümü

Tez Yöneticisi : Prof. Dr. Abdülkadir ERDEN

Eylül 1996, 94 sayfa

Bu çalışmada, tek kamera kullanarak bir gezer robotun çevresindeki bir kapıya göre konumunun hesaplanması amaçlanmıştır. Kamera parametrelerinin belirlenmesi için Newton Raphson ve Liu ve Huang yöntemleri kullanılmış ve bu iki yöntem gezer robot konumunun hesaplanmasına uyarlanmıştır. Programlar C++ programlama dilinde ve DOS işletim sisteminde yazılmıştır. Algoritmaların test edilmesi için deneyler yapılmış ve deney sonuçları tartışılmıştır.

Anahtar Kelimeler: Gezer Robotlar, Komputer görme, Robot görme, Çevrenin algılanması, Mesafe ölçümü.

ACKNOWLEDGEMENTS

I highly appreciate the aids I got from various individuals in this thesis work. Firstly I would like to thank to my supervisor Prof. Dr. Abdülkadir Erden who guided and supervised me throughout the work and establish the required medium for me. I wish to thank to project members Mr. Alper Alansal and Mr. Erhan Çokal for their invaluable thoughts and helps. Special thanks to Mr. Ömer Köksal, Mrs. Zuhale Erden who supported me with their ideas. Also I wish to thank Mr. Cem Kendi who shared his genius, assisting ideas with us. Finally I wish to thank to my family, Mr. Kemal Özdemir, Mrs. Binnur Özdemir, Mr. Okan Özdemir, Mr. Volkan Özdemir and my grandmother Mrs. Hatice Tanık who really supported me with patience.

to my grandmother *HATİCE TANIK*



TABLE OF CONTENTS

ABSTRACT	iii
ÖZ	iv
ACKNOWLEDGMENTS	v
TABLE OF CONTENTS	vii
LIST OF TABLES	x
LIST OF FIGURES	xii
CHAPTER	
1. INTRODUCTION	1
1.1 Mobile Robots	1
1.2 Definition of The Need	3
1.3 Scope of This Study	4
1.4 Outline of The Thesis	5
2. RELATED WORKS	7
2.1 Previous work	8
2.2 Need for a Now Research	10
3. DESIGN OF THE VISION SYSTEM	12
3.1 General	12
3.2 Low Level Image Processing	12

3.2.1 Image Capture	12
3.2.2 Smoothing Algorithm	14
3.2.3 Threshold Selection Techniques	14
3.3 Intermediate Level Image Processing	15
3.3.1 Edge Detection	15
3.3.2 Edge Tracing	15
3.4 High Level Image Processing (Interpretation of Images)	16
3.5 Determination of Mobile Robot Location	19
4. METHODS FOR DETERMINATION OF A	
MOBILE ROBOT LOCATION	20
4.1 General	20
4.2 Camera Model	21
4.3 Coordinate Systems and Transformations	24
4.4 Newton Raphson Method	32
4.5 Liu and Huang's Method	40
4.6 Computer Implementation	50
5. EXPERIMENTS AND RESULTS	55
5.1 General	55
5.2 Experimental Setup	55
5.3 Assumptions and Experimental Limitations	57
5.4 Experimental Parameters and Types of Experiments	58
5.5 Presentation of the Experimental Results	63
6. CONCLUSION	82
REFERENCES	85

APPENDIX

A. GENERAL PERSPECTIVE PROJECTION EQUATIONS _____ 89

B. LINES IN SPACES AND PLANES _____ 92



LIST OF TABLES

TABLES

5.1 Experiment Types and Sets _____	59
5.2 Experimental Results according to Liu and Huang's	
Method for Experiment Type 1 and Set 1. _____	64
5.3 Experimental Results according to Newton Raphson	
Method for Experiment Type 1 and Set1. _____	64
5.4 Absolute %Errors in Y_o for Experiment Type 1 and Set 1 _____	64
5.5 Experimental Results according to Liu and Huang's	
Method for Experiment Type 1 and Set 2. _____	65
5.6 Experimental Results according to Newton Raphson	
Method for Experiment Type 1 and Set 2. _____	65
5.7 Absolute %Errors in Y_o for Experiment Type 1 and Set 2. _____	65
5.8 Experimental Results according to Liu and Huang's	
Method for Experiment Type 2 and Set 3. _____	69
5.9 Experimental Results according to Newton Raphson	
Method for Experiment Type 2 and Set 3. _____	69
5.10 Absolute %Errors in Y_o for Experiment Type 2 and Set 3 _____	69
5.11 Experimental Results according to Liu and Huang's	
Method for Experiment Type 2 and Set 4. _____	70
5.12 Experimental Results according to Newton Raphson	
Method for Experiment Type 2 and Set4. _____	70

5.13	Absolute %Errors in Y_o for Experiment Type 2 and Set 4	70
5.14	Experimental Results according to Liu and Huang's Method for Experiment Type 3 and Set 5.	73
5.15	Experimental Results according to Newton Raphson Method for Experiment Type 3 and Set 5.	73
5.16	Absolute %Errors in Y_o for Experiment Type 3 and Set 5.	73



LIST OF FIGURES

FIGURES

3.1 Structure of the Software	13
3.2 A door Image with its Output after Each Process from 7 m.	17
3.3 A Door Image with its Output after Each Process from 5 m.	18
4.1 A Camera Model	22
4.2 Coordinate System	25
4.3 Camera Viewing in 3-D Scene and the Imaging Model	27
4.4 Definitions of Rotations	28
4.5 System Geometry	36
4.6 Mobile Robots Location at Indoor Environments	36
4.7 Flowchart of the Newton Raphson Method Solution	41
4.8 System Strategy	52
4.9 Several Image Processing Outputs and Coordinates of Lines in the Image Plane.	53
5.1 General view of Test Vehicle (1)	56
5.2 General view of Test Vehicle (2)	56
5.3 Experiment Type 1	60
5.4 Experiment Type 2	61
5.5 Experiment Type 3	62
5.6 Yo Errors for Experiment Type1 and Set1	66
5.7 Yo Errors for Experiment Type1 and Set2	66

5.8 Absolute % Errors in Y_o for Experiment Type1 and Set1	67
5.9 Absolute. % Errors in Y_o for Experiment Type1 and Set2	67
5.10 Y_o Errors for Experiment Type 2 and Set 3.	71
5.11 Y_o Errors for Experiment Type 2 and Set 4.	71
5.12 Absolute % Errors in Y_o for Experiment Type 2 and Set 3	72
5.13 Absolute % Errors in Y_o for Experiment Type 2 and Set 4	72
5.14 Y_o Errors for Experiment Type 3 and Set 5	75
5.15 Absolute %Errors in Y_o for Experiment Type3 and Set5	75
5.16 X_o and Orientation Errors according to (a) LHM;	
(b) NRM for Experiment Type 1 and Set 1.	76
5.17 X_o and Orientation Errors according to (a) LHM;	
(b) NRM for Experiment Type 1 and Set 2.	77
5.18 X_o and Orientation Errors according to (a) LHM;	
(b) NRM for Experiment Type 2 and Set 3.	78
5.19 X_o and Orientation Errors according to (a) LHM;	
(b) NRM for Experiment Type 2 and Set 4.	79
5.20 X_o and Orientation Errors according to (a) LHM;	
(b) NRM for Experiment Type 3 and Set 5.	80
B.1 Line in Space	92
B.2 Line in Plane and Normal Vector	94

CHAPTER 1

INTRODUCTION

1.1 Mobile Robots

Mobile robots are among the most popular subject in robotics research. A mobile robot is defined as an autonomous vehicle that can move within its environment requiring little human intervention. In order to achieve a high degree of autonomy, a mobile robot should be able to sense its surroundings, build or update information about its environment, plan and execute actions and adapt its behaviour to environmental changes. Implementation of these tasks require many background concepts like sensor integration and interpretation, real world modeling and mapping, actuator and a sensor based control architecture, path planning, navigation, decision making, task level planning and control of robotic system.

Study of mobile robot is a multi-disciplinary subject, utilizing techniques derived from optics, electronics, mechanical engineering, computer science and artificial intelligence. Design and development of mobile robots require a vast technological background and engineering experience on the above mentioned concepts. Many research institutions,

universities and industrial companies have emphasized research topics on autonomous mobile robots.

Today, there are various application fields of mobile robots. Briefly these are:

- Service application such as autonomous transport of luggage and meal in a hotel [1];
- Hazardous environments and applications such as dangerous zones in nuclear power stations [2];
- Space exploration where severe environment limits human exploration[3];
- Hospital transportation where meal trays and laboratory supplies are transported [4];
- Security guarding which is tiresome and requires careful observation dayaround[5];
- Factory automation where robot vehicles travel on the production floor and carry materials and products for production operations [6,7];
- Cleaning under poor conditions[7];

Review of literature about autonomous mobile indoor robots, reveal that they have some common problems of research. Perception of the environment through sensing is an important and popular research topic of mobile robots. A mobile robot running without human intervention or with little human intervention must feel its surrounding world and obtain some information from there. For that reason it is required to implement various sensing methods. For example, ultrasonic sensing, tactile sensing, infrared light, laser, computer vision etc.

Perception of environment by computer vision is a sophisticated and complex phenomenon. Robots with computer vision systems obtain images of an unknown 3D environment by using one or several optical cameras, interpret raw images by image processing techniques and then accordingly develop a collision free path.

1.2 Definition of The Need

A vision based mobile robot project has started at METU three years ago. Previously two mobile robots were developed at the Mechatronics Design Laboratory; MODROB_C and MODROB_D¹. They did not use camera as prototypes for vision based systems[8,9].

¹ Visit://http://www.me.metu.edu.tr/~erden/m_robots for more information on MODROB_C and MODROB_D

In this project, a vision based system with a single CCD (Charge Coupled Devices) camera is developed for mobile robots to recognize a door at indoor environments and estimate its position with respect to a door. Computer vision is needed because it is the most versatile artificial sense available to mobile robots and conveys extremely rich information. It also senses the environment in a remote manner.

1.3 Scope of This Study

Main aim of this project is to use a single CCD camera to perceive the environments. The system is designed to "see" a door indoors and navigate through the door. The mobile robot is assumed to move parallel or perpendicular to the wall and sharp color difference between the door and the background is expected. No additional sensory devices are planned to be utilized for navigation. Various image processing and analysis algorithms are developed and computer codes are implemented for the prototype vision system. This project was divided into three parts. These are:

- Automatic picture grasping and low level image processing [10]
- Recognition of the door and determination of its status (open,close) [11]
- Determination of mobile robot location by using a single CCD camera.

The objective of the third part of the project is to calculate the camera viewing parameters from 2D image of a door shape in 3D space by using a single CCD camera. In order to determine the mobile robot's current position and movement direction in real time, the camera viewing parameters must be needed. For determination of camera location, two algorithms will be presented. These are: Newton Raphson method and Liu and Huang's method. Each method is studied and applied in detail to estimate mobile robot position and orientation.

1.4 Outline of The Thesis

Outline of this thesis organisation is given as below:

Chapter 2 is an overview of published information the available papers related with this work.

Chapter 3 describes system design which is based on Vision and this chapter gives some information about developed various image processing and analysis algorithms.

Chapter 4 explains two methods for determination of a mobile robot location with respect to a recognized door. The camera model and transformations are given also.

Chapter 5 states experimental parameters and results. Properties of experimental setup and types of experiments are explained.

Chapter 6 discusses outcomes of the study and related conclusion and comments about further work.



CHAPTER 2

RELATED WORKS

This chapter is an overview of the literature on determination of camera viewing parameters, object configurations and robot location by using a single camera.

One of the basic tasks of a mobile robot is to reach its destination quickly and plan to a collision-free path [12,13]. This general task results in two basic requirements. The first is the availability of one or more sensor systems which perceive information about the environment. Distance measuring sensors based on laser, ultrasound, infrared and camera (vision) technologies are used most often for this purposes. The second requirement is a control link between these sensory systems and the robot's action.

For mobile robot navigation, a robot vehicle needs to determine its position relative to objects in its environment. If the robot is going to follow a pre-defined path then the information need to be obtained relative to the path. If the path is not pre-defined then the robot needs to know its position relative to the obstacles. Computer vision techniques are

commonly used for perception of its environment and determination of the robot positions. As an initial step camera viewing parameters must be calculated from image data to calculate robot position and orientation.

2.1 Previous work

Haralick [14] shows that it is possible to determine the camera viewing parameters from observed perspective projection of a rectangle. If the information about size of a rectangle is known, the camera parameters can be simply determined. This study is only theoretical and no experimental results are published yet.

A technique is developed for calculating external and internal camera parameters for machine vision using TV cameras and lenses [15]. The two-stage technique is aimed at efficient computation of camera external position and orientation relative to object reference coordinate system as well as the effective focal length and radial lens distortion parameters. The two stage technique has advantage in terms of accuracy and speed.

Using wire frame objects from their two dimensional perspective projections is another way of estimating the camera external parameters[16,17]. A wire frame object consists of a set of three dimensional (3D) arcs where each arc is a sequence of conics and line segments lying in the same plane[16,17]. Given a picture taken by a camera focusing on one wire frame object, this method shows how to

determine the objects positions relative to the camera. This problem is expressed as a nonlinear optimization search procedure on the six external camera viewing parameters: three translational and three rotational parameters. A solution is found when viewing parameters are determined which are consistent with the world knowledge having possible curves and observed image data.

Bülent E. Platin et al. developed a method which is based on the perspective projection by a single camera. The method reconstruct the object configuration in 3D space by using a circular target [18]. Its elliptic projection on the image plane is studied.

Zen Chen [20] considers cubes in 3D space to determine the camera position and orientation relative to an object. Most of the equations used to determine position are simple and non-trigonometric. A cube with known dimension is chosen to be a calibration object. The basic theory for 3D position determination utilizes the 3D line equations and the 3D plane equations of the cube in terms of the 2D coordinates of the vanishing points obtained from the perspective projection of the cube.

A new method for a fundamental problem of determining the position and orientation of a 3D object using single perspective image plane is introduced by Yuwan Wu [21]. The technique is focused on the interpretation of angle constraint information. This method provides a general analytic technique for dealing with a class of problem of shape from inverse perspective projection by using "Angle to Angle

Correspondence Information". Different trihedral angle configurations are generated for testing the approach of finding object orientation by angle to angle constraints. It is reported that this method works effectively in noisy environment.

Application of the landmark technique in mobile robot navigation is discussed by Courney and Magee [22]. The main aim of the work was to locate an autonomous robot vehicle in a three dimensional space. A diamond formed mark by four thick lines on a contrasting background is used in order to estimate the mobile robot positions .

Another technique of calculating the robot position is line correspondences from 3D world to 2D image. Ayache [23] discussed the notations of geometric representations of 3D lines and planes for estimating camera parameters and robot locations. Xiang Wang [24] calculated camera external and internal parameters by using 3D to 2D line or point correspondences. The Author considered the problem of combining use of points or lines in a sequence of images to recover 3D structure.

X. Zhuang [25] focused on the robust 3D pose estimation. From various objects of interest in the view of a camera, corresponding point data set is obtained. The robust pose estimation can be formulated as Gaussian error noise model and MF estimator to solve each unknown rotation and translation parameters [25,26].

2.2 Need for a New Research

The general task of this mobile robot is to recognize the door in its environment, then move towards the door and pass through the door if it is open. Also this robot is required to operate autonomously in unknown surroundings, in particular buildings without explicit cues or markers. For these purposes and navigation, a mobile robot need to estimate its location relative to the natural objects in the environment. According to the mobile robot task, the door is used as a natural object, in robot environment. To estimate robot location with respect to the door, the camera viewing parameters must be known.

CHAPTER 3

DESIGN OF THE VISION SYSTEM

3.1 General

Main components and simplified structure of the developed software are illustrated in Figure 3.1. Each of the modules is isolated from the others and alternative algorithms are studied thoroughly to select the best one for our case. Code implementations of several alternative algorithms are performed to test their relative performance whenever it is necessary. Image processing and analysis algorithms which are implemented in this work are discussed in the following section.

3.2 Low Level Image Processing

3.2.1 Image Capture

DT2851 frame grabber is used for this purpose. The captured image is 256 colored gray image. The image size (256x256 pixels) is used.

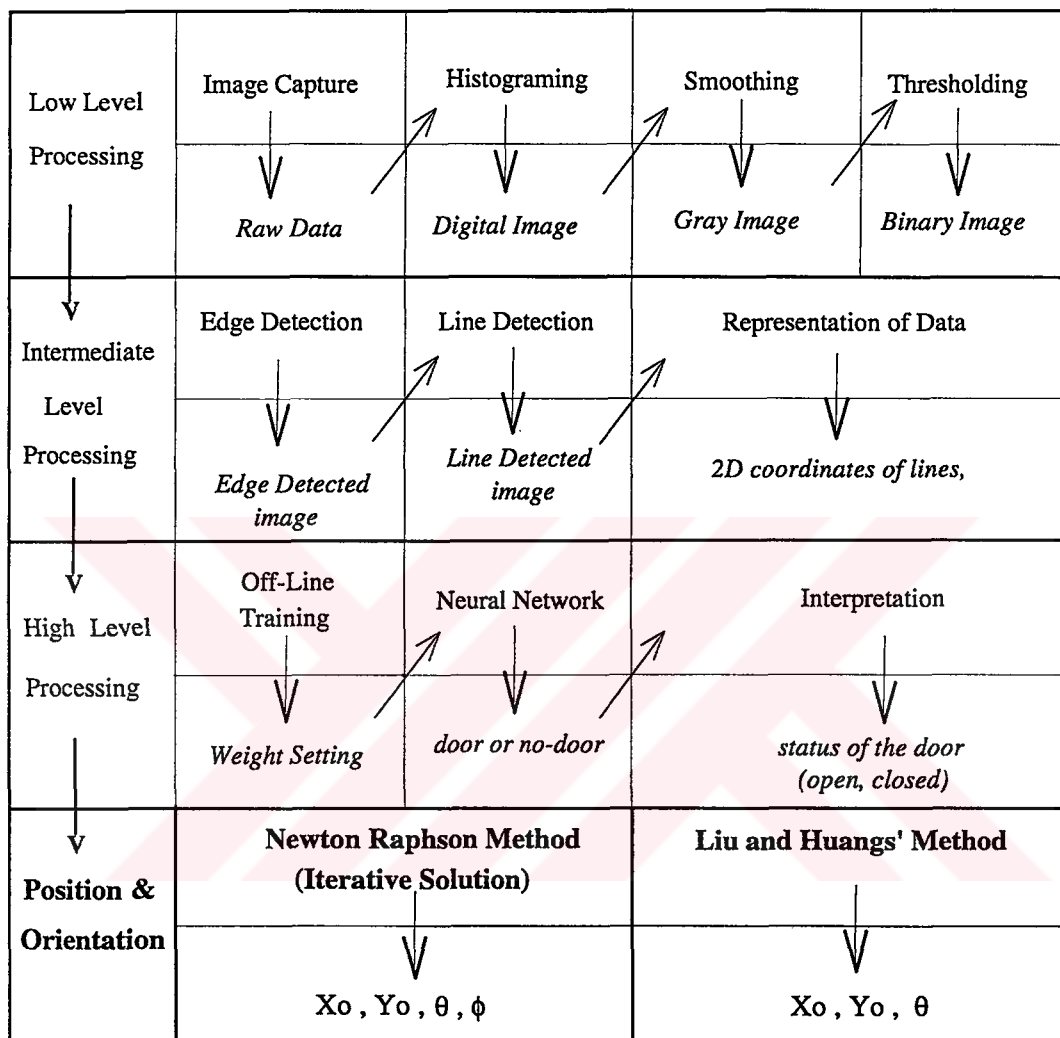


Figure 3.1 Structure of the Software. Italic terms indicate output of each stages.

3.2.2 Smoothing Algorithm

The histogram data structure is obtained by number of pixels versus gray-level values (0-255) and the histogram data is generated by using 256x256 pixels digitized image. By using this generated histogram data we have to find a threshold value for our aim which is to slice the gray image into a binary image. This process should be automatic and efficient, since the image processing during the operation should be completely on-line [10].

Due to illumination and other objects in the environment, the peak number and roughness of the histogram increases. Therefore applying a thresholding method and determining a threshold value to separate objects and background becomes difficult. In order to overcome this difficulty a smoothing algorithm was developed within this project [10].

3.2.3 Threshold Selection Techniques

The selection of an appropriate threshold is the single major problem for segmentation. A lot of techniques which have been proposed, most are based on the analysis of the gray-level histogram [10].

To determine the threshold value of the histogram the following four methods which are popular in the literature were chosen and their algorithms developed and implemented [10]:

- Ostu Method
- Entropy Method
- Minimum Kullback Method
- Moment Preserving Method

Details about these methods are given in other thesis study. [10].

3.3 Intermediate Level Image Processing

The intermediate level image processing includes edge detection, edge tracing for line detection and representation procedures. These stages of the system aims to extract data for high level image processing.

3.3.1 Edge Detection

A great number of edge detection techniques are presented in the literature. Laplacian Gradient Operator is used because of its simple, fast, and noise sensitive characteristics [10].

3.3.2 Edge Tracing

Hough Transformation [27] is applied first as line detection algorithm based on a previous experience, however end points can not be

located satisfactorily and processing time is long. Therefore a novel line detection algorithm based on edge tracing method is developed and implemented [10]. Representation of line segments includes calculation of length and angle of each segment and orientation by using the coordinates of the end points which are also used as the inputs of training process of multilayer feed forward neural network. Details of these processes are given in other thesis [11]. Figure 3.2, and Figure 3.3 indicate a door image with its outputs after thresholding, edge detection and line detection stages from different distancens.

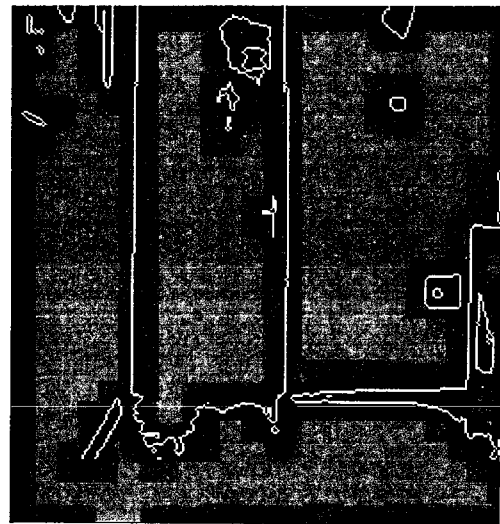
3.4 High Level Image Processing (Interpretation of Images)

Recognition of a door and interpretation of its status (open or closed) are the main goals of this part of the study. Back-propagation error learning algorithm for the training of multilayered neural network is applied for classification of randomly captured image. Extensive number of experiments are performed to obtain information which is necessary for knowledge base to interpret door images[11].

A decision making algorithm is developed to collect all of the available information and reach a final decision. Related data are further interpreted to calculate position and orientation of the robot with respect to the door. The door position is estimated by using intersection of door-vertical lines and base line of the image. These two coordinates are the only useful data for position determination after interpretation process of the door [11].



Binary Image

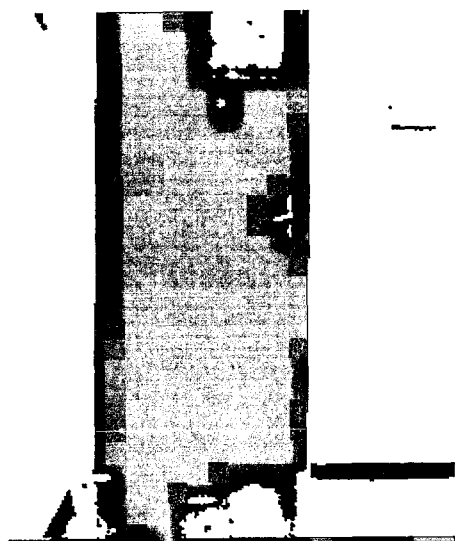


Edge Detected Image

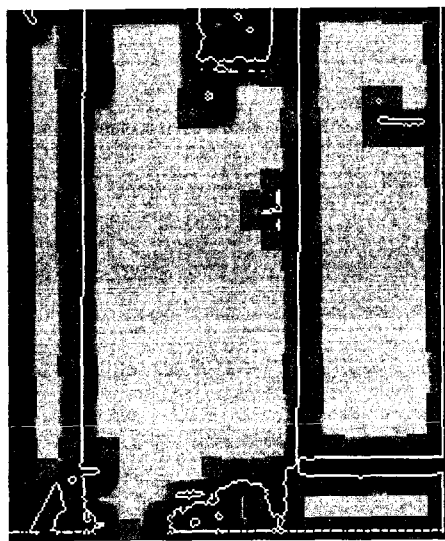


Line Detected Image

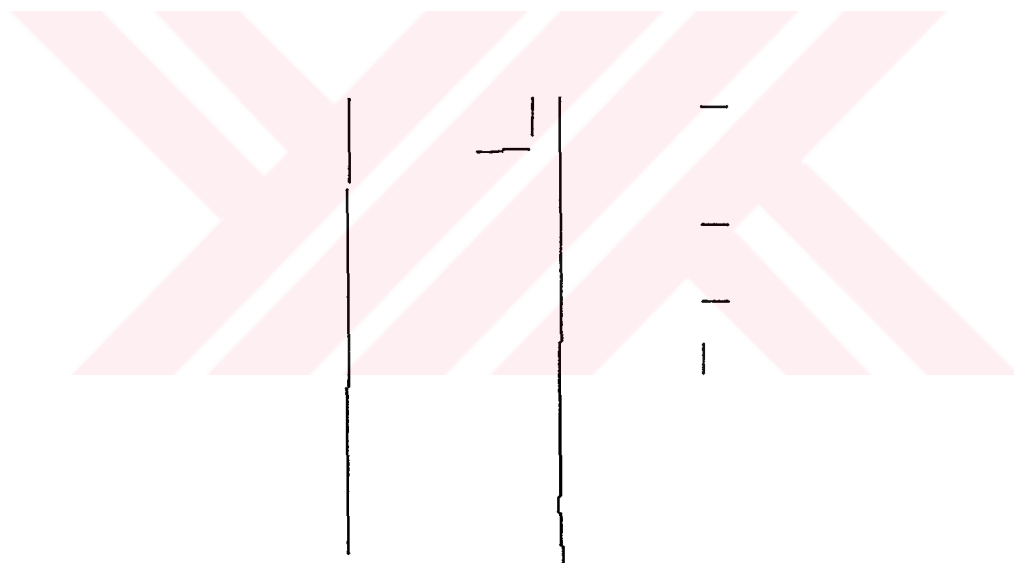
Figure 3.2. A door image with its output after each process from 7 m



Binary Image



Edge Detected Image



Line Detected Image

Figure 3.3. A door image with its output after each process from 5 m

3.5 Determination of Mobile Robot Location

The mobile robot position and orientation in 3D space are calculated by using general perspective projection equations after finding coordinates of corner points of the door in 2D image plane. In order to determine the mobile robot's current position and moving direction with respect to the door, viewing parameters of the camera must be determined as a first step.

In this study, two methods are used which input 2D image coordinates of straight line or two point correspondences, to decide translation and rotation angles of the camera. These methods are modified to mobile robot position and orientation determination. These two methods are explained in the next chapter.

CHAPTER 4

METHODS FOR DETERMINATION OF A MOBILE ROBOT LOCATION

4.1 General

Determining camera location from image to space point correspondences (i.e., 2-D and 3-D correspondences) is an important problem in vision based systems. It may be applied to aircraft location, estimation of robot position and orientation, and so on. Roughly, this problem may be defined as a problem of estimating three dimensional location from an image which include a set of recognized objects or landmarks.

In this chapter, two algorithms for determination of camera location will be presented. These are: Newton Raphson method [26] and Liu and Huang's method [29,30]. Each method describes how to apply it to mobile robot position and orientation determination. First of all, camera model will be explained and then the coordinate systems and transformations required for a solution will be defined.

4.2 Camera Model

A camera is illustrated as in Figure 4.1 using a perspective projection model where the perspective center is the optic center O_c . The figure also shows the 3D cartesian coordinate systems which is defined and used in the following formulations [19]:

- The world coordinate system ($O-XYZ$) is positioned at some convenient point. Its orientation is arbitrary.
- The sensor coordinate system ($O_s - X_s Y_s Z_s$) is fixed to the sensor such that O_s is at the geometric center of the sensor, Y_s is perpendicular to the sensor plane and X_s is parallel to the horizontal axis of the sensor.
- The camera coordinate system ($O_c - X_c Y_c Z_c$) is fixed to the lens such that O_c is at the optic center and Y_c is the optical axis.

The camera parameters can be divided into two groups: these are the "external" and "internal" parameters. For a CCD camera, the external parameters define position and orientation of the sensor with respect to the world coordinate system. The internal parameters of a camera are specified by the camera constant or image plane distance, coordinates of the principal point and distortion characteristics of the lens.

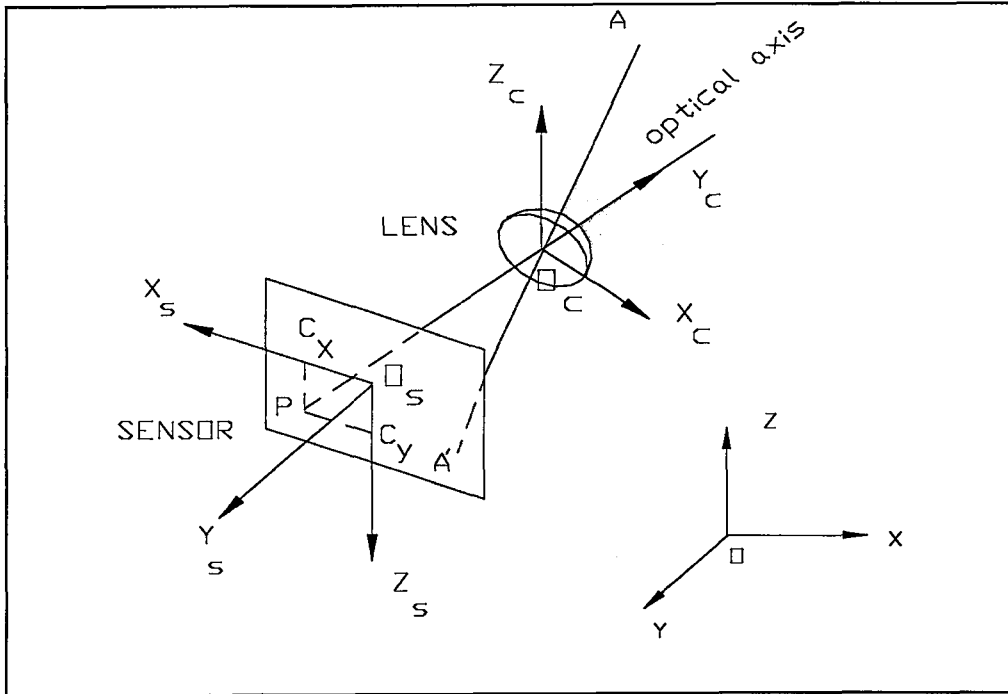


Figure 4.1 A Camera Model

The camera model parameters can be described by an 11 element set in which the first five are camera internal parameters and the later six are the camera external parameters:

$$g = \{ C_x, C_y, k_1, k_2, f, X_0, Y_0, Z_0, \theta, \phi, \xi \} \quad (4.1)$$

where

C_x, C_y : positions of the optical center on the sensor plane.

k_1, k_2 : first and second items of radial lens distortion coefficient.

f : camera constant, the distance between the image plane and the camera lens.

X_0, Y_0, Z_0 : positions of the camera coordinate center with respect to the world coordinate system.

θ, ϕ, ξ : orientations of the camera coordinate axis with respect to the world coordinate system.

As illustrated in Figure 4.1 the principal point P is the intersection of the optical axis and the image plane. When the principal point and the geometric center of the sensor are not coincident, the corrected image coordinates $(x_{\text{cor}}, y_{\text{cor}})$ are calculated using

$$X_{\text{cor}} = U - C_x \quad Y_{\text{cor}} = V - C_y \quad (4.2)$$

where (U, V) are the image coordinates of a point on the image plane. The result is simply a coordinate shift. In an ideal camera model, it is assumed that P and O_s are coincident. Thus C_x and C_y are equal to zero.

The radial distortions are modeled by k_1 and k_2 . The distorted coordinates (X_d, Y_d) are obtained using

$$X_d = kX_{\text{cor}} \quad Y_d = kY_{\text{cor}} \quad (4.3)$$

where,

$$k = 1 + k_1 r^2 + k_2 r^4$$

$$r^2 = X_{\text{cor}}^2 + Y_{\text{cor}}^2 \quad (4.4)$$

In this study, it is assumed that the camera model is ideal or pinhole. Therefore k_1 and k_2 are equal zero.

The ideal camera model can be summarized as follows:

- the principal point and the coordinate center of the sensor are coincident.
- all coefficients of lens distortion are neglected.
- the optical axis is perpendicular to the sensor.
- the focal distance is taken as the supplied catalogue value.

4.3 Coordinate Systems and Transformations

Figure 4.2 describes the cartesian coordinate systems which are used in the formulations. Let O -XYZ be the world coordinate system with origin O fixed on the ground. The X -axis and the Y -axis, lying on the ground, are perpendicular to each other, and the Z -axis is vertical to the ground and points upwards. Also let O_c - $X_cY_cZ_c$ be the camera coordinate system with origin O_c being the lens center of the camera. The Y_c -axis is the optical axis of the camera and points away from the camera. The X_c - Z_c plane is parallel to the image plane. The image coordinate system defined in terms of Q -UV in which origin Q is the image plane

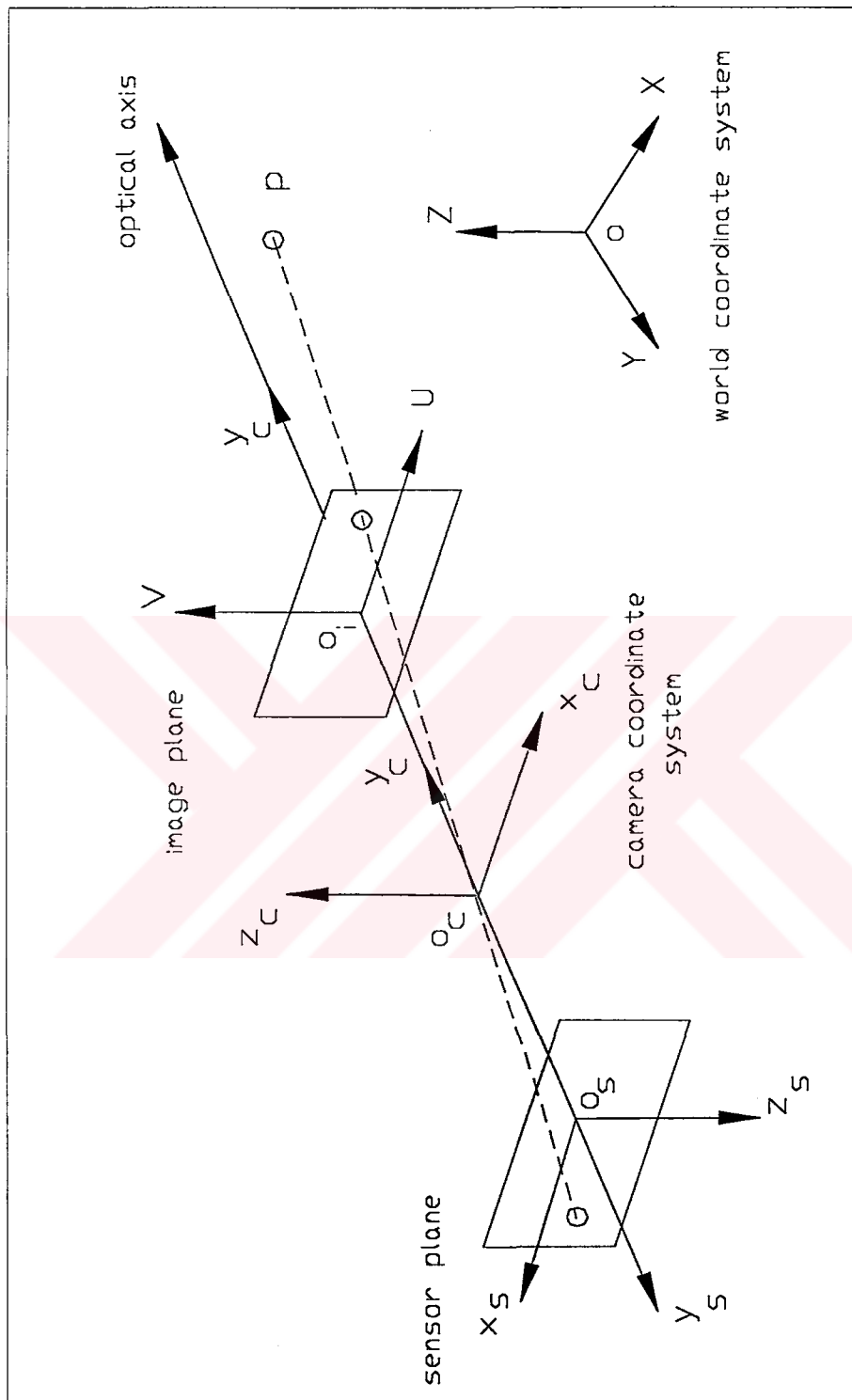


Figure 4.2 Coordinate System

center, and the U-axis and V-axis are parallel to the X_c -axis and Y_c -axis, respectively.

Computer vision problems often involve interpreting the information on a two dimensional (2D) image of a three dimensional (3D) world in order to determine the location and orientation of a camera from using objects portrayed in the image.

$$P_{\text{WORLD}}(X, Y, Z) \longleftrightarrow P_{\text{IMAGE}}(U, V)$$

A perspective transformation (also called an image transformation) projects 3-D point onto a 2-D plane. We need to develop perspective projection equations by including all transformations between various coordinate systems. Relation between the camera coordinate system and the world coordinate system is given in terms of translation and rotation. The image plane is not parallel to X-Z plane of the world coordinate system as shown in Figure 4.3.

A point (X, Y, Z) in the world coordinate system must be expressed relative to the position (X_0, Y_0, Z_0) of the camera lens. Translation of a point in the world coordinate system to the location of the camera coordinate center is accomplished by using transformation matrix and the point (X, Y, Z) has new coordinates.

$$\begin{bmatrix} X - X_0 \\ Y - Y_0 \\ Z - Z_0 \end{bmatrix}$$

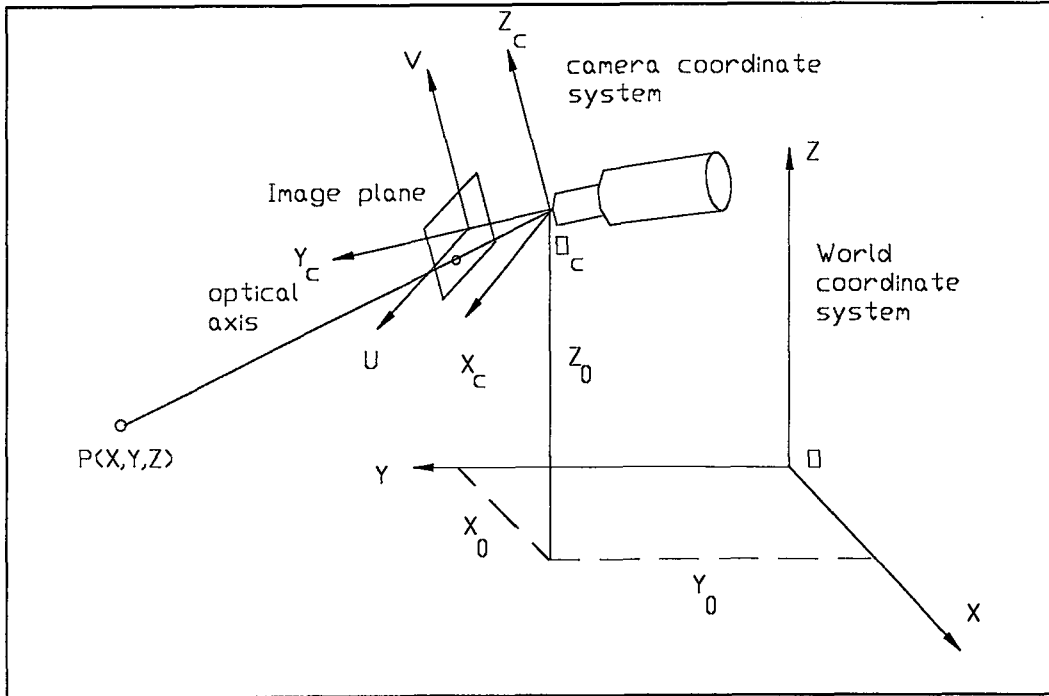


Figure 4.3 Camera viewing in 3-D scene and the imaging model

Direction of X , Y and Z axes of the camera reference frame differ from those of the world reference frame. We represent the rotation by which the world reference frame is brought into correspondence with the camera reference frame as a sequence of three rotations. The rotation around the X -axis which is called tilt angle ϕ , is the first rotation. The rotation around Y -axis which is called swing angle ξ , is the second rotation. The rotation around the Z -axis which is called pan angle θ , is the third rotation. These rotations are shown in Figure 4.4.

With reference to Figure 4.4, rotation about the Z coordinate axis by an angle θ is achieved by using the following transformation;

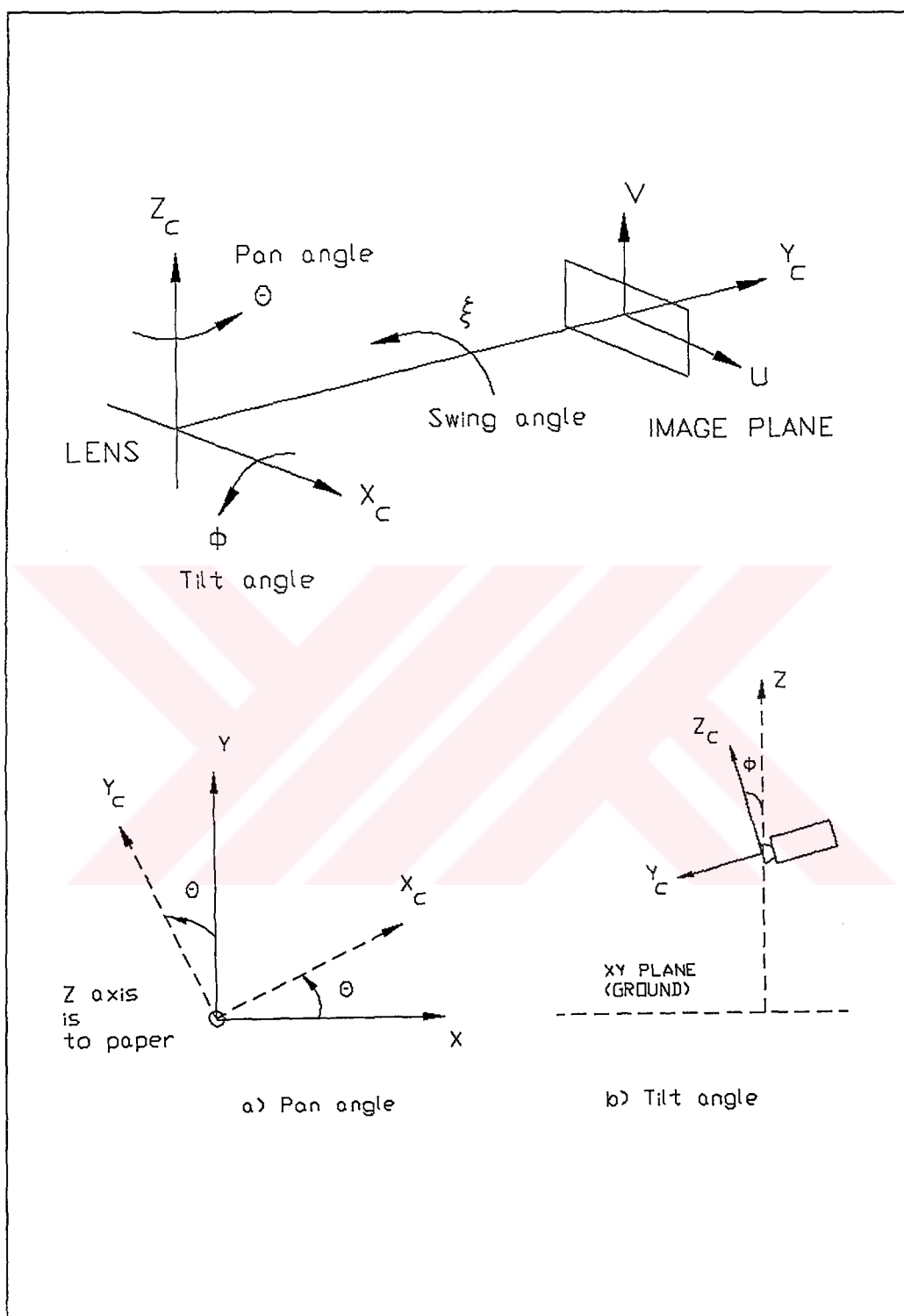


Figure 4.4 Definitions of Rotations

$$R(\theta) = \begin{bmatrix} \cos \theta & \sin \theta & 0 \\ -\sin \theta & \cos \theta & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

The rotation angle θ is measured counterclockwise when looking at the origin from a point on the Z axis. This transformation affects only the values of X and Y coordinates.

Rotation about the X axis by an angle ϕ is performed by using the following transformation

$$R(\phi) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos \phi & \sin \phi \\ 0 & -\sin \phi & \cos \phi \end{bmatrix}$$

Finally, rotation about the Y axis by an angle ξ is achieved by using the following transformation

$$R(\xi) = \begin{bmatrix} \cos \xi & 0 & -\sin \xi \\ 0 & 1 & 0 \\ \sin \xi & 0 & \cos \xi \end{bmatrix}$$

Fig.4.4 shows that the pan angle is measured between Y_c and Y axes. When Y_c and Y axes are aligned, the pan angle is equal to zero. Tilt is the angle between Z_c and Z axes. When Z_c and Z axes are aligned, the tilt angle is 0° . A counterclockwise rotation of the camera implies positive angles.

These three rotation matrices can be concatenated into a single matrix,

$$R(\phi, \xi, \theta) = R(\phi)R(\xi)R(\theta)$$

where

$$R = \begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{bmatrix}$$

and substituting explicit forms;

$$R = \begin{bmatrix} \cos\xi \cos\theta & \cos\xi \sin\theta & -\sin\xi \\ \sin\phi \sin\xi \cos\theta - \cos\phi \sin\theta & \sin\phi \sin\xi \sin\theta + \cos\phi \cos\theta & \sin\phi \cos\xi \\ \cos\phi \sin\xi \cos\theta + \sin\phi \sin\theta & \cos\phi \sin\xi \sin\theta - \sin\phi \cos\theta & \cos\phi \cos\xi \end{bmatrix}$$

The point (XYZ) in the world coordinate system is then represented by the point (X_C, Y_C, Z_C) in the camera coordinate system where,

$$\begin{bmatrix} X_C \\ Y_C \\ Z_C \end{bmatrix} = \begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{bmatrix} \begin{bmatrix} X - X_0 \\ Y - Y_0 \\ Z - Z_0 \end{bmatrix} \quad (4.5)$$

Having a representation for the 3-D point in the camera reference frame, then its perspective projection is taken. It is assumed that a camera is pinhole and an image plane is at a distance f in front of the camera lens. Let (U,V) be coordinates of the projection point of P in the image coordinate system; then the transformation between (U,V) and (X_c, Y_c, Z_c) is:

$$U = f \frac{X_c}{Y_c} \quad V = f \frac{Z_c}{Y_c} \quad (4.6)$$

or in matrix form,

$$\begin{bmatrix} U \\ V \end{bmatrix} = \frac{f}{Y_c} \begin{bmatrix} X_c \\ Z_c \end{bmatrix} \quad (4.7)$$

The optical axis of the camera is the Y_c axis in this equation. Rewriting these equations in terms of rotational dependencies,

$$\begin{aligned} \frac{U}{f} &= \frac{r_{11}(X - X_0) + r_{21}(Y - Y_0) + r_{31}(Z - Z_0)}{r_{21}(X - X_0) + r_{22}(Y - Y_0) + r_{23}(Z - Z_0)} \\ \frac{V}{f} &= \frac{r_{31}(X - X_0) + r_{32}(Y - Y_0) + r_{33}(Z - Z_0)}{r_{21}(X - X_0) + r_{22}(Y - Y_0) + r_{23}(Z - Z_0)} \end{aligned} \quad (4.8)$$

This pair of equation is known as the general or fundamental perspective projection equations². Then the problem is to determine six camera parameters $(X_0, Y_0, Z_0, \phi, \xi, \theta)$ where (X_0, Y_0, Z_0) is the camera lens position in the world coordinate system and (ϕ, ξ, θ) are the tilt, swing and pan angles of the camera. The camera constant f is known and does not vary in this study. It is assumed that the camera constant f is equal to focal length of the camera in some of the literature [14,17,26,30,33]. In the following section, firstly Newton-Raphson Method is presented. Secondly, Liu and Huang's method is discussed.

4.4 Newton-Raphson Method

Equations (4.8) show that the relationship between the measured 2-D camera coordinates and the 3-D world coordinates is a nonlinear function of X_0, Y_0, Z_0, ϕ, ξ and θ . Given enough pairs of corresponding 2D and 3D points and initial approximate solution, the unknown parameters X_0, Y_0, Z_0, ϕ, ξ and θ can be solved by using the Newton Raphson Method.

This method is based on determining values of the unknown parameters that minimize the sum of the squares of the residuals. For this approaches, successful solutions are often highly dependent on good initial guesses for the unknown parameters. In those cases, when an initial approximate solution to the Newton Raphson problem is known or given, the exact solution to the Newton Raphson problem can be obtained by iteratively

² For more details see Appendix A

solving the linearized problem in which the linearization is taken around the current approximate solution.

Suppose β_1, \dots, β_M are the unknown parameters governing each of the nonlinear transformation g_1, \dots, g_K and that $\alpha_1, \dots, \alpha_K$ are the observed or measured values of g_1, \dots, g_K

$$\alpha = \begin{bmatrix} \alpha_1 \\ \alpha_2 \\ \vdots \\ \alpha_K \end{bmatrix} \quad \text{and} \quad g = \begin{bmatrix} g_1(\beta_1, \dots, \beta_M) \\ g_2(\beta_1, \dots, \beta_M) \\ \vdots \\ g_K(\beta_1, \dots, \beta_M) \end{bmatrix}$$

which minimizes the following equation.

$$\|\alpha - g\| = \sum_{k=1}^M (\alpha_k - g_k(\beta_1, \dots, \beta_M))^2 \quad (4.9)$$

To solve the case when g_1, \dots, g_K are nonlinear functions and there is a given initial approximate solution $\beta^0 = (\beta_1^0, \dots, \beta_M^0)$, we begin by linearizing the nonlinear transformations around β^0 and solve for the adjustment $\Delta\beta = (\Delta\beta_1, \dots, \Delta\beta_M)$, which when added to β constitute a better approximate solution.

At the r th iteration let $\beta^r = (\beta_1^r, \dots, \beta_M^r)$ be the current approximate solution. The linearization proceeds by representing each unknown variable

by a first-order Taylor series expansion and substituting the linearized expression into the following criterion,

$$G^t \Delta\beta = (\alpha - g^t) \quad (4.10)$$

where t is the t th iteration, the vector $\Delta\beta$ contains changes in the parameter values.

$$g^t = \begin{bmatrix} g_1(\beta_1^t, \dots, \beta_M^t) \\ g_2(\beta_1^t, \dots, \beta_M^t) \\ \vdots \\ g_K(\beta_1^t, \dots, \beta_M^t) \end{bmatrix}$$

G^t is the Jacobian matrix given by

$$G^t = \begin{bmatrix} \frac{\partial g_1}{\partial \beta_1} & \frac{\partial g_1}{\partial \beta_2} & \dots & \frac{\partial g_1}{\partial \beta_M} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial g_K}{\partial \beta_1} & \frac{\partial g_K}{\partial \beta_2} & \dots & \frac{\partial g_K}{\partial \beta_M} \end{bmatrix}_{K \times M}$$

$K=M$ and each partial derivative of G^t is evaluated at $(\beta_1^t, \dots, \beta_M^t)$. The equation (4.10) is solved for $\Delta\beta$ which are then to the current β to form the next β as in:

$$\begin{bmatrix} \beta_1^{t+1} \\ \vdots \\ \beta_K^{t+1} \end{bmatrix} = \begin{bmatrix} \beta_1^t \\ \vdots \\ \beta_K^t \end{bmatrix} + \begin{bmatrix} \Delta\beta_1 \\ \vdots \\ \Delta\beta_K \end{bmatrix} \quad (4.11)$$

This procedure is repeated until the solution converges, that is until a threshold value falls below an acceptable stopping criterion.

$$\sum_K \varepsilon_K = |\beta_K^{t+1} - \beta_K^t| \quad (4.12)$$

In order to navigate through its environments a mobile robot needs to determine its position relative to certain objects. In this project, the purpose of the mobile robot is to find a door around its environment and move through the door with a narrow clearance. To ensure this purpose the world coordinate system is attached to the door (the center of door-floor edge line) as it is shown in Fig 4.5. Thus two base edge points of a door are used. If the door size is known, the world coordinate (X_1, Y_1, Z_1) of point P_1 is equal to $(-\frac{w}{2}, 0, 0)$ and the world coordinate (X_2, Y_2, Z_2) of point P_2 is equal to $(\frac{w}{2}, 0, 0)$ and w is the width of the door. This information is used for perspective projection equations and the Newton Raphson solution as an input.

The camera parameters for a CCD camera are explained in the previous section. The robot position can be described by three parameters as it is shown in Fig 4.6. These parameters are:

- The lateral position along the X axis between the camera and the navigation line (the x-shift) is denoted by X_0 .

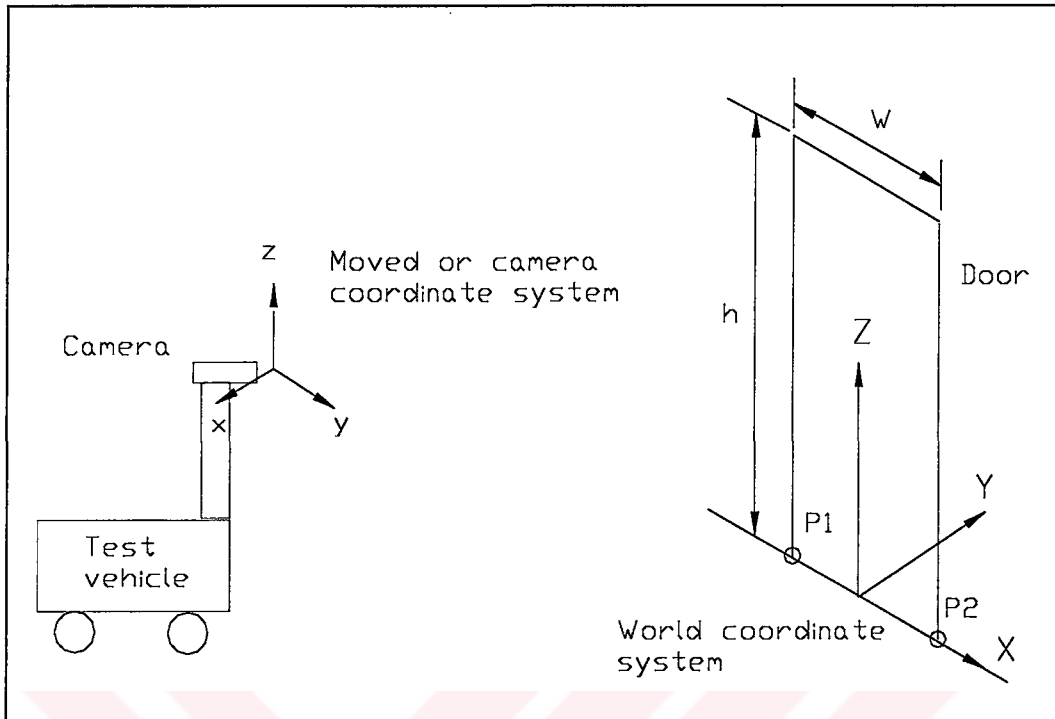


Figure 4.5 System Geometry

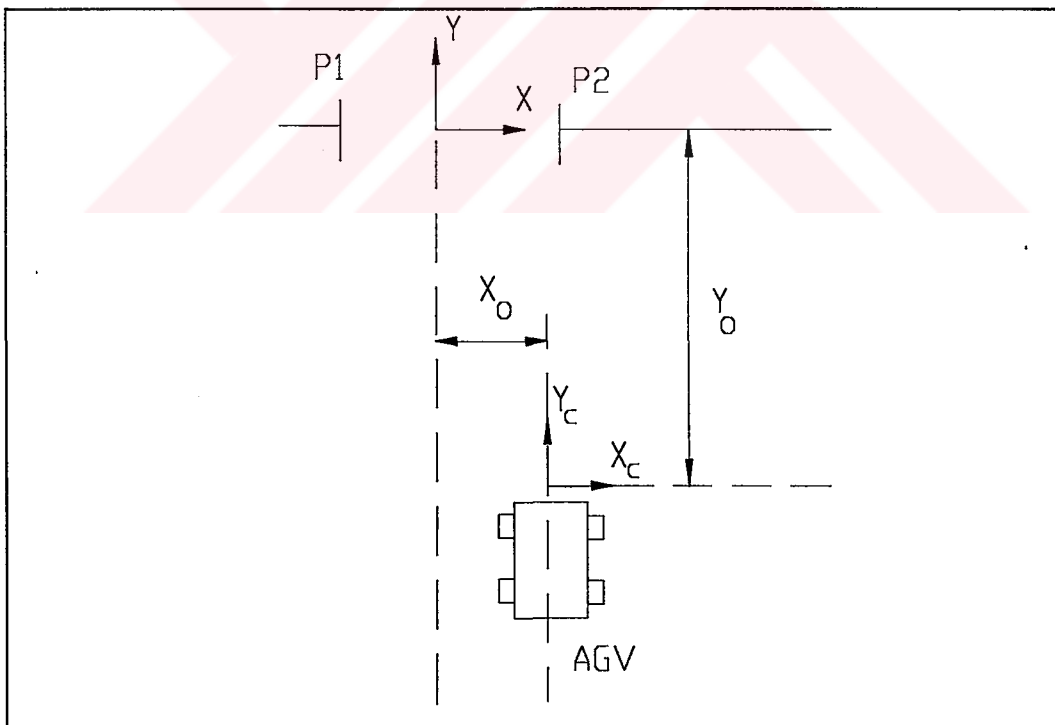


Figure 4.6 Mobile Robots Location at in door environments

- The lateral position along the Y axis between the sensor and the door is denoted by Y_0 .
- The pan angle is denoted by θ .

Let the unknown translation vector be given by (X_0, Y_0, Z_0) . The point (X_n, Y_n, Z_n) of the door in the world coordinate system becomes the point (X_{nc}, Y_{nc}, Z_{nc}) of the camera coordinate system, where

$$\begin{bmatrix} X_{nc} \\ Y_{nc} \\ Z_{nc} \end{bmatrix} = R(\phi, \xi, \theta) \begin{bmatrix} X - X_0 \\ Y - Y_0 \\ Z - Z_0 \end{bmatrix} \quad n=1, \dots, N \quad N=2 \quad (4.13)$$

n : the number of points

The observed perspective projection equation is then given by the following equation

$$\begin{bmatrix} U_n \\ V_n \end{bmatrix} = \frac{f}{Y_{nc}} \begin{bmatrix} X_{nc} \\ Z_{nc} \end{bmatrix} \quad (4.14)$$

In this study, the camera is fixed in front of a moving platform and the camera height Z_0 is known. In addition, the swing angle ξ is equal to zero. It is assumed that the camera model is pinhole. Then the problem is determination of four camera parameters $\beta = (X_0, Y_0, \theta, \phi)$.

In the notation of the Newton Raphson method,

$$g_{2n-1}(\beta) = U_n(X_0, Y_0, \theta, \phi) = f \frac{X_{nc}(X_0, Y_0, \theta, \phi)}{Y_{nc}(X_0, Y_0, \theta, \phi)}$$

$$g_{2n}(\beta) = V_n(X_0, Y_0, \theta, \phi) = f \frac{Z_{nc}(X_0, Y_0, \theta, \phi)}{Y_{nc}(X_0, Y_0, \theta, \phi)} \quad (4.15)$$

where n is the number of points. There are two sets of 2-D to 3-D point correspondences as follows:

$$P_1:(X_1, Y_1, Z_1) \longleftrightarrow (U_1, V_1)$$

$$P_2:(X_2, Y_2, Z_2) \longleftrightarrow (U_2, V_2)$$

(X_1, Y_1, Z_1) and (X_2, Y_2, Z_2) are the coordinates of points P_1 and P_2 in world coordinate system, respectively, and (U_1, V_1) and (U_2, V_2) are the coordinates of the projection points P_1 and P_2 in the image coordinate system. Equation 4.10 is defined in this form:

$$G' \begin{bmatrix} \Delta X_0 \\ \Delta Y_0 \\ \Delta \theta \\ \Delta \phi \end{bmatrix} = \begin{bmatrix} \alpha_1 - g_1' \\ \alpha_2 - g_2' \\ \alpha_3 - g_3' \\ \alpha_4 - g_4' \end{bmatrix} = \begin{bmatrix} \Delta U_1 \\ \Delta V_1 \\ \Delta U_2 \\ \Delta V_2 \end{bmatrix}$$

where at iteration t , g_1', g_2', g_3' and g_4' are the function values of perspective projection equations.

for point P_1 ;

$$g_1^t(X_0, Y_0, \theta, \phi) = U_1^t(X_0, Y_0, \theta, \phi)$$

$$g_2^t(X_0, Y_0, \theta, \phi) = V_1^t(X_0, Y_0, \theta, \phi)$$

for point P_2 ;

$$g_3^t(X_0, Y_0, \theta, \phi) = U_2^t(X_0, Y_0, \theta, \phi)$$

$$g_4^t(X_0, Y_0, \theta, \phi) = V_2^t(X_0, Y_0, \theta, \phi)$$

and $\alpha_1, \alpha_2, \alpha_3$ and α_4 are the coordinates of points in the image plane U_1, V_1, U_2 and V_2 , respectively.

By using equations (4.15) the jacobian matrix is obtained in this notation.

$$G^t = \begin{bmatrix} \frac{\partial U_1^t}{\partial X_0} & \frac{\partial U_1^t}{\partial Y_0} & \frac{\partial U_1^t}{\partial \theta} & \frac{\partial U_1^t}{\partial \phi} \\ \frac{\partial V_1^t}{\partial X_0} & \frac{\partial V_1^t}{\partial Y_0} & \frac{\partial V_1^t}{\partial \theta} & \frac{\partial V_1^t}{\partial \phi} \\ \frac{\partial U_2^t}{\partial X_0} & \frac{\partial U_2^t}{\partial Y_0} & \frac{\partial U_2^t}{\partial \theta} & \frac{\partial U_2^t}{\partial \phi} \\ \frac{\partial V_2^t}{\partial X_0} & \frac{\partial V_2^t}{\partial Y_0} & \frac{\partial V_2^t}{\partial \theta} & \frac{\partial V_2^t}{\partial \phi} \end{bmatrix}$$

and equation (4.10) can be obtained in the following form;

$$\begin{bmatrix} \frac{\partial U_1^t}{\partial X_0} & \frac{\partial U_1^t}{\partial Y_0} & \frac{\partial U_1^t}{\partial \theta} & \frac{\partial U_1^t}{\partial \phi} \\ \frac{\partial V_1^t}{\partial X_0} & \frac{\partial V_1^t}{\partial Y_0} & \frac{\partial V_1^t}{\partial \theta} & \frac{\partial V_1^t}{\partial \phi} \\ \frac{\partial U_2^t}{\partial X_0} & \frac{\partial U_2^t}{\partial Y_0} & \frac{\partial U_2^t}{\partial \theta} & \frac{\partial U_2^t}{\partial \phi} \\ \frac{\partial V_2^t}{\partial X_0} & \frac{\partial V_2^t}{\partial Y_0} & \frac{\partial V_2^t}{\partial \theta} & \frac{\partial V_2^t}{\partial \phi} \end{bmatrix} \begin{bmatrix} \Delta X_0 \\ \Delta Y_0 \\ \Delta \theta \\ \Delta \phi \end{bmatrix} = \begin{bmatrix} U_1 - U_1^t \\ V_1 - V_1^t \\ U_2 - U_2^t \\ V_2 - V_2^t \end{bmatrix} \quad (4.16)$$

This equation is solved for $\Delta\beta=(\Delta X_0,\Delta Y_0,\Delta\theta,\Delta\phi)$. And then the new approximate solution is given by;

$$\begin{bmatrix} X_0^{t+1} \\ Y_0^{t+1} \\ \theta^{t+1} \\ \phi^{t+1} \end{bmatrix} = \begin{bmatrix} X_0^t \\ Y_0^t \\ \theta^t \\ \phi^t \end{bmatrix} + \begin{bmatrix} \Delta X_0 \\ \Delta Y_0 \\ \Delta \theta \\ \Delta \phi \end{bmatrix}$$

This procedure is repeated until the solution converges. LU decomposition techniques [32] is used for solving Eq.(4.16) in this study. Fig 4.7 shows the implementation flowchart of this process.

4.5 Liu and Huangs' Method

In this section Liu and Huang's method for determining camera location from straight line correspondences is presented. Liu and Huang proposed a method for the determination of camera location from 2-D to

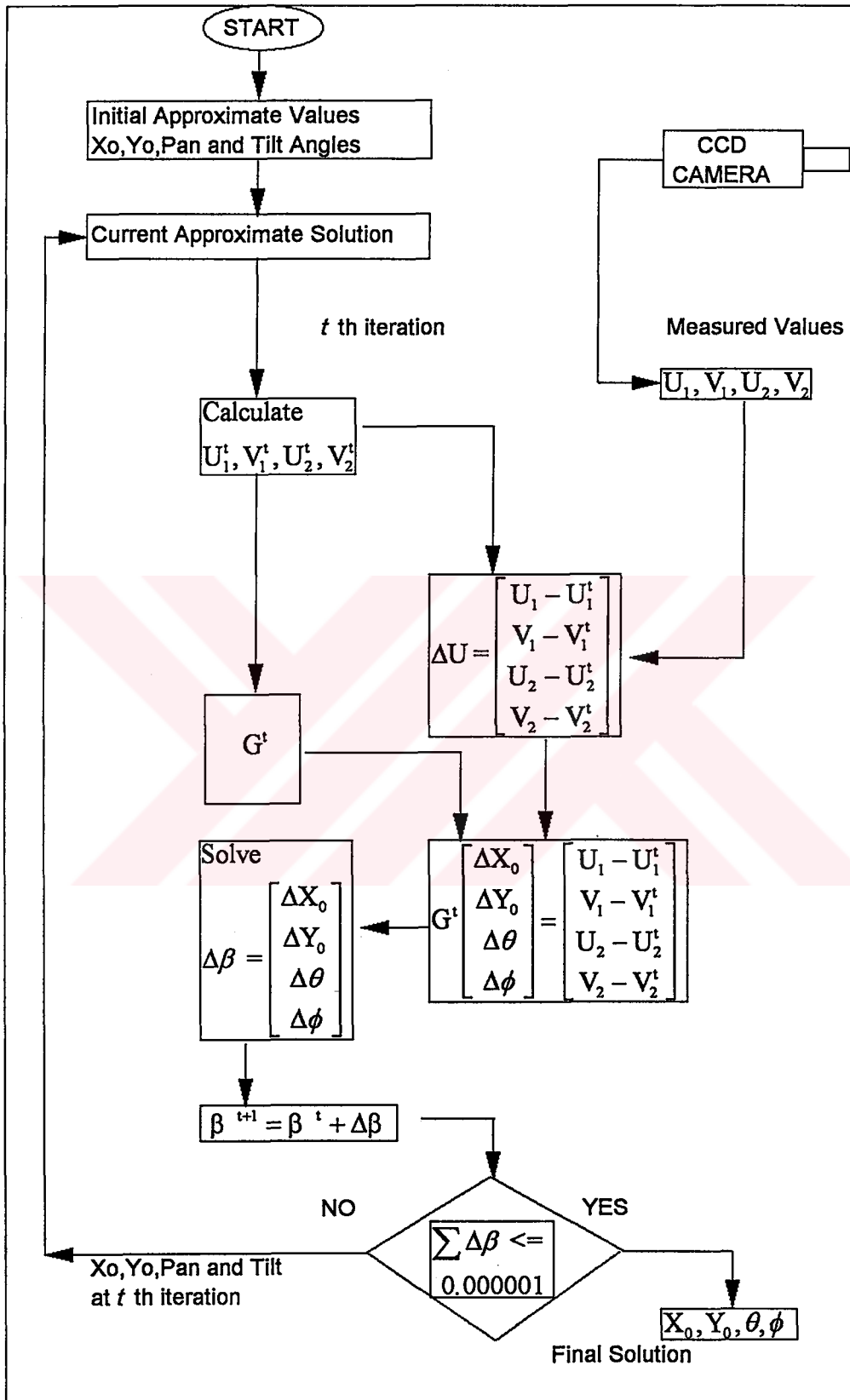


Figure 4.7 Flowchart of the Newton Raphson Method Solution.

3-D straight line or point correspondences[30]. Lines can be created from given points. The coordinate systems and transformation equations are explained in section 4.3. As indicated earlier, the camera coordinate system is related to the world coordinate system and located on the ground by a rotation followed by a translation. The rotation matrix and the translation vector can be solved separately.

A 3-D line l in the world coordinate system is represented in the parametric form:

$$\vec{p} = \vec{n}t + \vec{p}_0 \quad (4.17)$$

where $\vec{n} = (a, b, c)^T$ is the direction of the line or the cosine vector of a line l and $\vec{p}_0 = (x_0, y_0, z_0)^T$ is a point on the line³. The image of this 3D line in the image coordinate system Q-UV can be expressed as [17,26,28,29,30] :

$$L: AU + BV + Cf = 0$$

The line L is on the image plane having surface normal $N = [A \ B \ C]$ and also this vector is the normal of the projection plane. Thus in the image coordinate system, the vector $N = [A \ B \ C]$ and line L are perpendicular and from geometrical relations this expression is obtained easily [28,29,30]:

$$\vec{N} \cdot \vec{n}_c = 0 \quad (4.18)$$

³ For more details see Appendix B

In the camera coordinate system ($O_c - X_c Y_c Z_c$), the direction of 3D line l is expressed in

$$\vec{n}_c = R \vec{n}$$

where R is a rotation matrix [29,30]. Since a 3D line is always perpendicular to the normal of its image plane, we have:

$$\vec{N} \cdot \vec{n}_c = 0$$

or this expression is written as;

$$\vec{N} \cdot \vec{n}_c = \vec{N} R \vec{n} = 0 \quad (4.19)$$

and in matrix form

$$\begin{bmatrix} A & B & C \end{bmatrix} \begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{bmatrix} \begin{bmatrix} a \\ b \\ c \end{bmatrix} = 0 \quad (4.20)$$

A line correspondence derived from point correspondences to estimate the rotation matrix by using Eq.(4.19). Then the translation vector

can be easily computed. A line can be constructed from two points. As indicated in Section 4.3 there are two sets of 2D to 3D door corner point correspondences. Taking these two image points (U_1, V_1) and (U_2, V_2) and the related space points (X_1, Y_1, Z_1) and (X_2, Y_2, Z_2) respectively, a line L on the image plane can be generated by connecting these two points :

$$AU + BV + Cf = 0 \quad \text{and} \quad A^2 + B^2 + C^2 = 1$$

where

$$A = \frac{(V_1 - V_2)}{k} f$$

$$B = \frac{(U_2 - U_1)}{k} f$$

$$C = \frac{(U_1 V_2 - V_1 U_2)}{k}$$

$$k = \sqrt{(V_1 - V_2)^2 f^2 + (U_2 - U_1)^2 f^2 + (U_1 V_2 - U_2 V_1)^2}$$

and the 3D line by connecting points P_1 and P_2 which have the direction \vec{n} :

$$a = \frac{(X_1 - X_2)}{m}$$

$$b = \frac{(Y_1 - Y_2)}{m}$$

$$c = \frac{(Z_1 - Z_2)}{m}$$

$$m = \sqrt{(X_1 - X_2)^2 + (Y_1 - Y_2)^2 + (Z_1 - Z_2)^2}$$

To simplify the formulas, it is assumed that both the tilt and the swing angles are zero, since the mobile robot only rotates relative to the Y axis of the world coordinate system. Thus the rotation matrix has only pan angle and is represented as it follows:

$$R = \begin{bmatrix} \cos\theta & \sin\theta & 0 \\ -\sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

Using equation (4.20), we have :

$$\begin{bmatrix} A & B & C \end{bmatrix} \begin{bmatrix} \cos\theta & \sin\theta & 0 \\ -\sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} a \\ b \\ c \end{bmatrix} = 0$$

which can be derived to be the following form:

$$e_1 \times \sin\theta + e_2 \times \cos\theta = e_3 \quad (4.21)$$

where

$$\begin{aligned} e_1 &= A \times b - B \times a \\ e_2 &= A \times a + B \times b \\ e_3 &= -C \times c \end{aligned}$$

and

$$\cos^2\theta + \sin^2\theta = 1 \quad (4.22)$$

From Equations (4.21) and (4.22), the pan angle is obtained as:

$$\begin{aligned}\cos\theta &= \frac{e_3e_2 + \sqrt{e_3^2e_2^2 - (e_1^2 + e_2^2)(e_3^2 - e_1^2)}}{e_1^2 + e_2^2} \\ \sin\theta &= \frac{e_3 - e_2 \cos\theta}{e_1} \\ \text{and} \\ \theta &= \sin^{-1} \frac{e_3 - e_2 \cos\theta}{e_1}\end{aligned}\tag{4.23}$$

To determine the translation vector $T = (X_0, Y_0, Z_0)^T$, point correspondences directly. Eq. (4.5) can be defined this form:

$$\begin{bmatrix} X_{nc} \\ Y_{nc} \\ Z_{nc} \end{bmatrix} = R \begin{bmatrix} X_n \\ Y_n \\ Z_n \end{bmatrix} - R \begin{bmatrix} X_0 \\ Y_0 \\ Z_0 \end{bmatrix}\tag{4.24}$$

where n is the number of points and

$$P_{gn} = RP_n \stackrel{\Delta}{=} \begin{bmatrix} X_{gn} \\ Y_{gn} \\ Z_{gn} \end{bmatrix} \quad T' = RT \stackrel{\Delta}{=} \begin{bmatrix} X'_0 \\ Y'_0 \\ Z'_0 \end{bmatrix}\tag{4.25}$$

From general perspective projection equations (4.6), the coordinates of the image point can be written as:

$$\begin{aligned}
U_n &= f \frac{X_{nc}}{Y_{nc}} = f \frac{X_{gn} - X'_0}{Y_{gn} - Y'_0} \\
V_n &= f \frac{Z_{nc}}{Y_{nc}} = f \frac{Z_{gn} - Z'_0}{Y_{gn} - Y'_0}
\end{aligned}
\tag{4.26}$$

which can be expressed in a matrix form as follows:

$$\begin{bmatrix} 1 & \frac{-U_n}{f} & 0 \\ 0 & \frac{-V_n}{f} & 1 \end{bmatrix} \begin{bmatrix} X'_0 \\ Y'_0 \\ Z'_0 \end{bmatrix} = \begin{bmatrix} X_{gn} - Y_{gn} \frac{U_n}{f} \\ Z_{gn} - Y_{gn} \frac{V_n}{f} \end{bmatrix}
\tag{4.27}$$

In this equation (X'_0, Y'_0, Z'_0) are unknown variables. We can get two equations from each set of 2D to 3D point correspondences and therefore two points are needed to solve T' .

From Liu and Huang's works [30], we imagine that there is an intermediate coordinate system $\overline{O}_g - \overline{X}_g \overline{Y}_g \overline{Z}_g$ which is formed by rotating the world coordinate system by the rotation matrix R (but without any translation) to determine the translation vector of the camera. It is evident that the relationship between the intermediate coordinate system $\overline{O}_g - \overline{X}_g \overline{Y}_g \overline{Z}_g$ and the camera coordinate system $O_c - X_c Y_c Z_c$ is a pure translation. As it is indicated earlier, let P_1 and P_2 be two corner points of a door at indoor environments; (X_1, Y_1, Z_1) and (X_2, Y_2, Z_2) are their coordinates in the world coordinate system $O-XYZ$, respectively, and

$(\bar{X}_{g1} \bar{Y}_{g1} \bar{Z}_{g1})$ and $(\bar{X}_{g2} \bar{Y}_{g2} \bar{Z}_{g2})$ are their coordinates in the intermediate coordinate system $(\bar{O}_g - \bar{X}_g \bar{Y}_g \bar{Z}_g)$. The coordinates of their projection points on the image plane coordinate system Q-UV are (U_1, V_1) and (U_2, V_2) , respectively. Then for each point, following expressions are obtained;

$$\begin{bmatrix} \bar{X}_{g1} \\ \bar{Y}_{g1} \\ \bar{Z}_{g1} \end{bmatrix} = R \begin{bmatrix} X_1 \\ Y_1 \\ Z_1 \end{bmatrix} = \begin{bmatrix} \cos \theta & \sin \theta & 0 \\ -\sin \theta & \cos \theta & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} X_1 \\ Y_1 \\ Z_1 \end{bmatrix}$$

$$= \begin{bmatrix} X_1 \cos \theta + Y_1 \sin \theta \\ -X_1 \sin \theta + Y_1 \cos \theta \\ Z_1 \end{bmatrix}$$

and for point P_2

$$\begin{bmatrix} \bar{X}_{g2} \\ \bar{Y}_{g2} \\ \bar{Z}_{g2} \end{bmatrix} = R \begin{bmatrix} X_2 \\ Y_2 \\ Z_2 \end{bmatrix} = \begin{bmatrix} \cos \theta & \sin \theta & 0 \\ -\sin \theta & \cos \theta & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} X_2 \\ Y_2 \\ Z_2 \end{bmatrix}$$

$$= \begin{bmatrix} X_2 \cos \theta + Y_2 \sin \theta \\ -X_2 \sin \theta + Y_2 \cos \theta \\ Z_2 \end{bmatrix}$$

Substituting the above equations into Equations (4.27);

$$\begin{bmatrix} 1 & -\frac{U_1}{f} & 0 \\ 0 & -\frac{V_1}{f} & 1 \end{bmatrix} \begin{bmatrix} X'_o \\ Y'_o \\ Z'_o \end{bmatrix} = \begin{bmatrix} \bar{X}_{g1} - \bar{Y}_{g1} \frac{U_1}{f} \\ \bar{Z}_{g1} - \bar{Y}_{g1} \frac{V_1}{f} \end{bmatrix}$$

$$\begin{bmatrix} 1 & -\frac{U_2}{f} & 0 \\ 0 & -\frac{V_2}{f} & 1 \end{bmatrix} \begin{bmatrix} X'_0 \\ Y'_0 \\ Z'_0 \end{bmatrix} = \begin{bmatrix} \bar{X}_{g2} - \bar{Y}_{g2} \frac{U_2}{f} \\ \bar{Z}_{g2} - \bar{Y}_{g2} \frac{V_2}{f} \end{bmatrix} \quad (4.28)$$

Equations (4.26) can be solved to be:

$$X'_0 = \frac{\bar{X}_{02}U_1 - \bar{X}_{01}U_2 + (\bar{Y}_{01} - \bar{Y}_{02})\frac{U_1U_2}{f}}{U_1 - U_2}$$

$$Y'_0 = \frac{f(\bar{X}_{02} - \bar{X}_{01}) + (\bar{Y}_{01}U_1 - \bar{Y}_{02}U_2)}{U_1 - U_2}$$

$$Z'_0 = \frac{(\bar{X}_{02} - \bar{X}_{01})V_1 + (U_1 - U_2)\bar{Z}_{01} + (\bar{Y}_{01} - \bar{Y}_{02})\frac{U_2V_1}{f}}{U_1 - U_2}$$

When T' is found, the translation vector T can be computed from Eq.(4.25) as follows:

$$T' = RT \Rightarrow T = R^{-1} T'$$

$$T = R^{-1} T' = \begin{bmatrix} \cos \theta & -\sin \theta & 0 \\ \sin \theta & \cos \theta & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} X'_0 \\ Y'_0 \\ Z'_0 \end{bmatrix}$$

So finally, have:

$$T = \begin{bmatrix} X_0 \\ Y_0 \\ Z_0 \end{bmatrix} = \begin{bmatrix} X'_0 \cos \theta - Y'_0 \sin \theta \\ X'_0 \sin \theta + Y'_0 \cos \theta \\ Z'_0 \end{bmatrix} \quad (4.29)$$

where y-axis is the optical axis and Z_0 is the height of the camera. The change of the Z_0 is zero because the mobile robot only moves on the ground. Therefore there is no need to calculate the value of Z_0 .

4.6 Computer Implementation

Computer codes of the mentioned methods are developed in C++ language. For the Newton Raphson Method (NRM), each partial of Jacobian Matrix is evaluated at unknown robot position parameters by using REDUCE 3.4.

As indicated in Section 4.4 to obtain solution matrix at t th iteration, LU decomposition techniques are used and implemented in this system[32]. Our initial approximate solution for NRM can come from experiments. Initial approximate solution need to be within 10% of scale for the translation parameters and within 15° or equal to zero for the rotational parameters for a successful. Fig 4.7 in Section 4.4 shows the implementation flowchart of this process. Generally, a solution is obtained

after nine iterations. Iterations are assumed to converged for $\sum \Delta\beta \leq 0.000001$.

A mobile robot needs to estimate its position relative to certain object to navigate through its environment. In this project, as an object a door is used in mobile robot's environment to estimate its position and the purpose of the mobile robot is to find a door around its environment and move through the door with a narrow clearance. To calculate the mobile robot position, as an input a 3D information of a door in the world coordinate system is needed. To ensure this purpose the world coordinate system is attached to the door (the center of door floor edge line). Thus two base edge points P1, P2 of a door are used. If the door size is used, the world coordinate (X_1, Y_1, Z_1) of point P1 is equal to $(-\frac{w}{2}, 0, 0)$ and the world coordinate (X_2, Y_2, Z_2) of point P2 is equal to $(\frac{w}{2}, 0, 0)$ and w is the width of the door and equal to 1070 mm.

After capturing an image by using a single CCD camera and several image processing and analysis algorithms, the coordinates of all lines in the digitized image are determined. Two base edge points of the door are used as an 2D information for two methods mentioned before. Figure 4.8 shows the system strategy and Figure 4.9 shows the several image processing output and coordinates of lines in digitized image. In Figure 4.9, the door vertical lines are 7 and 10. For line 7 the end point coordinates is (54,210) in pixels and for line 10 the end point coordinates (128,210) in pixels.

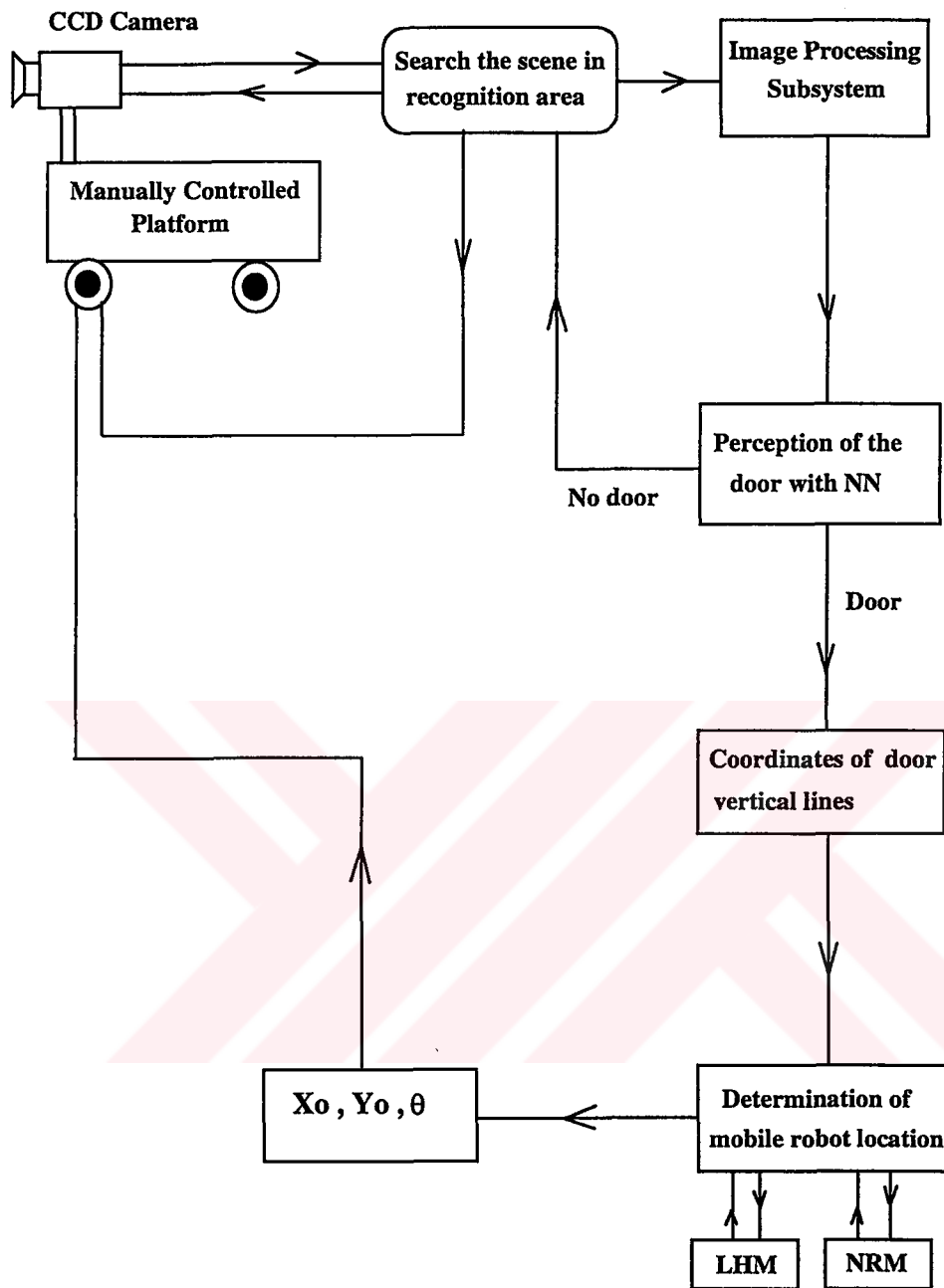
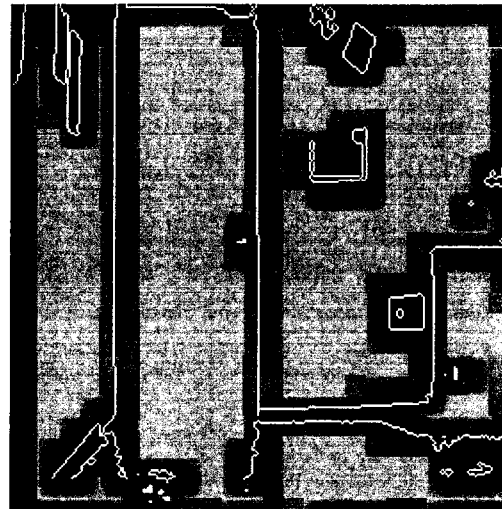


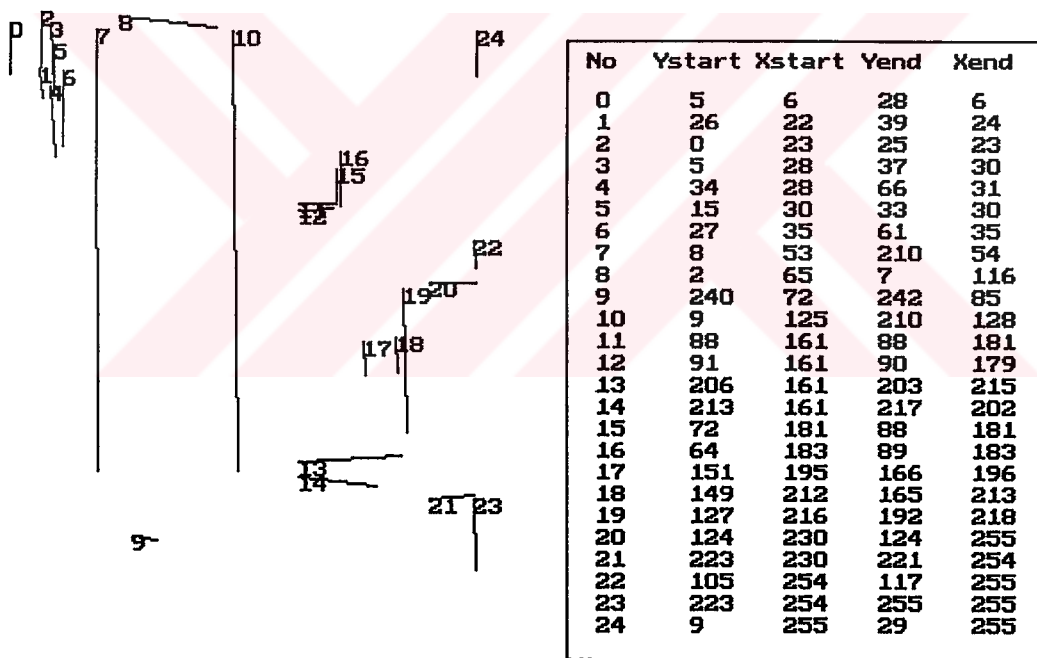
Figure 4.8 System Strategy



A binary image of a door



Edge Detected Image



Line detected image and coordinates of lines in the image

Figure 4.9 Several Image Processing Outputs and Coordinates of Lines in the Image Plane.

These two points of the door are used for both Liu and Huang's Method and the Newton Raphson Method Algorithms to estimate the mobile robot location with respect to a door. After finding coordinates of two points in digitized image, these coordinates are represented in sensor plane(or in image plane) and obtained (U_1, V_1) and (U_2, V_2) of point P1 and P2 respectively.

Door width : 1070 mm

Sensor size : 6.3 mm x 4.7 mm

Image Size : 256 x 256 pixels

Effective Picture (Sensor) Elements : 752 x 582

f : 11.50 mm (from camera properties)

In the following, the position of the mobile robot for this image is given by using two methods.

True Values	NRM Algorithm	LHM Algorithm
$X_o = 50$ cm	$X_o = 53.50$ cm	$X_o = 53.50$ (cm)
$Y_o = 700$ cm	$Y_o = 673.76$ cm	$Y_o = 678.63$ (cm)
$\theta = 0$ (degree)	$\theta = 0.00$ (degree)	$\theta = 0.00$ (degree)

Main advantages of these two algorithms is that only two points in the image as an input are sufficient to calculate the X_o , Y_o and θ parameters if it is assume that a camera is ideal.

CHAPTER 5

EXPERIMENTS AND RESULTS

5.1 General

This chapter describes experiments performed for determination of the mobile robot location. Firstly experimental setup and experimental limitations and system assumptions will be described. Later, results will be discussed and observed errors will be analysed. All experiments are performed in Mechatronic Laboratory of the Mechanical Engineering Department at METU. Results are calculated with two different methods: the Newton Raphson Method and Liu and Huang's Method. Test results are plotted using MS Excel 5.0. Tables and graphics show the numerical values of mobile robot location parameters as well as errors. These results will be given for both algorithms.

5.2 Experimental Setup

In this project, a moving platform with four casters is used to simulate a mobile robot. Figure 5.1 and Figure 5.2 show a general view of the test vehicle.

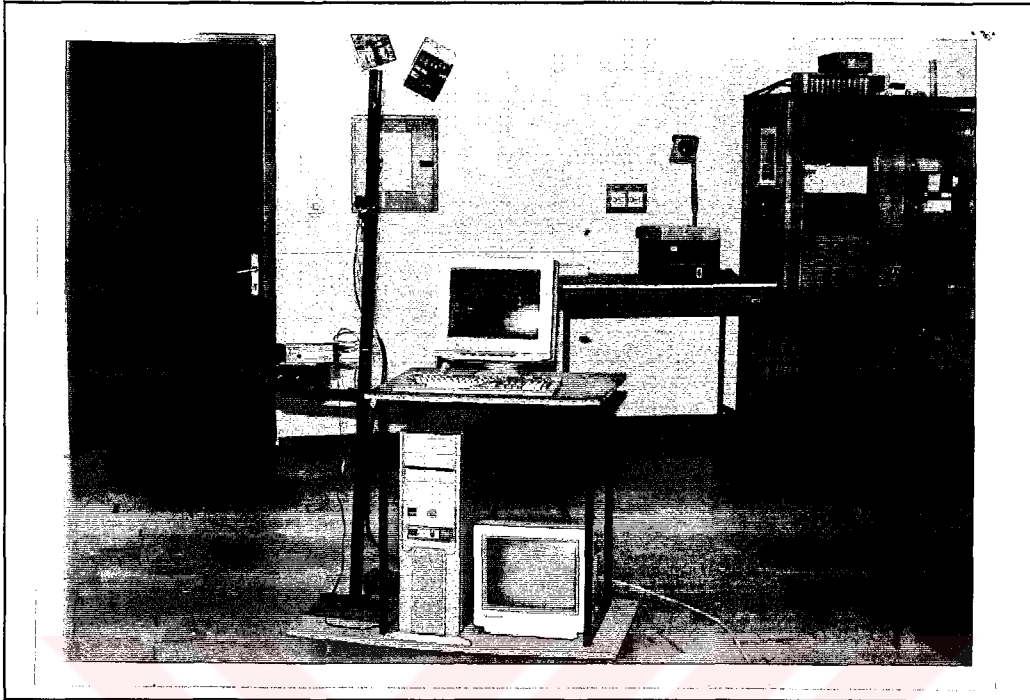


Figure 5.1 General view of Test Vehicle (1)

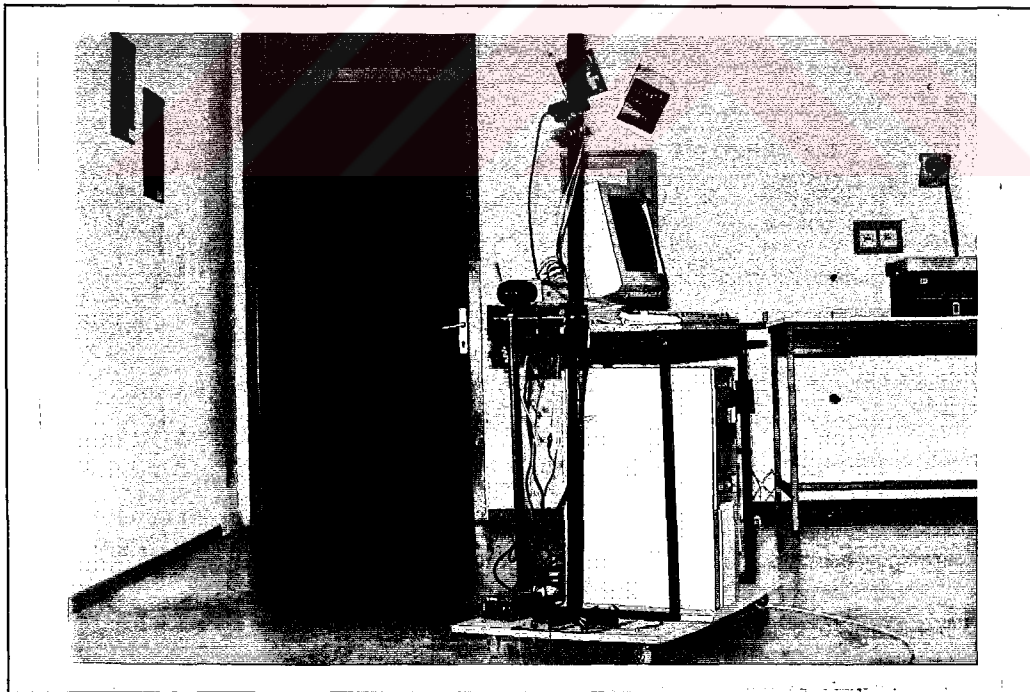


Figure 5.2 General view of Test Vehicle (2)

The experimental equipment includes:

- Intel 80486 DX-33 processor based computer,
- The DT2851 High Resolution Frame Grabber,
- Sony SSC-M350CE/M370CE CCD black and white video camera
Sensor area 6.3 x 4.7 mm (1/4 x 3/6 inches)
Picture (sensor) elements 752 x 582 (horizontal/vertical). The number of CCD picture elements (about 437000) is sufficient to ensure high resolution,
- Two monitors.

5.3 Assumptions and Experimental Limitations

Following assumptions and experimental limitations are considered at the beginnig of this project.

- Sharp color difference between the door and walls.
- No obstacles around the navigation area.
- The camera is ideal. All components of lens distortion coefficients are equal to zero.

- No restriction on illumination.
- The working distance is between 7.5m and 4m in these experiments (This range is determined after some preliminary experiments).
- Image of the door is not masked by other objects in the scene.
- The mobile robot is assumed to move parallel or perpendicular to the wall.
- A camera is fixed on the test vehicle with known and fixed camera height.

5.4 Experimental Parameters and Types of Experiments

As indicated in the previous chapter, mobile robot position can be described by using three parameters. These are X_o , Y_o and θ . X_o is a current position along the x axis between the camera and navigation line. Y_o is a distance along the y axis between the door and the camera. θ is the rotation of a mobile robot with respect to a reference point. For all experiments these three parameters will be considered.

In the navigation guidance phase, the test vehicle is moved on the ground to simulate the mobile robot navigation. The navigation

experiments are simulated at indoor environments, the range of working distance is between 7.5m to 4m.

In order to see performance of the algorithms three types of experiments are developed. For Type1, two sets of experiments are done by decreasing Y_o from 7m to 4m while keeping X_o and θ equal to zero. For Type2, two sets of experiments are done by decreasing Y_o from 7m to 4m while keeping X_o equal to 50cm and θ equal to zero. Finally for Type3, one set of experiments is done. In this experiment the X_o and θ are kept constant. X_o is equal to -10cm and θ is equal to zero. These types of experiments are shown in Table 5.1 and Figure 5.3, Figure 5.4. and Figure 5.5.

Experiment Type	X_o	Y_o	θ
1	0	\neq	0
2	> 0	\neq	0
3	< 0	\neq	0

Experiments	Experiment Type
Set 1	1
Set 2	1
Set 3	2
Set 4	2
Set 5	3

Table 5.1 Experiment Types and Sets

Image Size : 256 x 256 pixels
 Sensor Size : 6.3 mm x 4.7 mm
 Camera Height : 900 mm
 Door Size : 1070 mm x 2100 mm

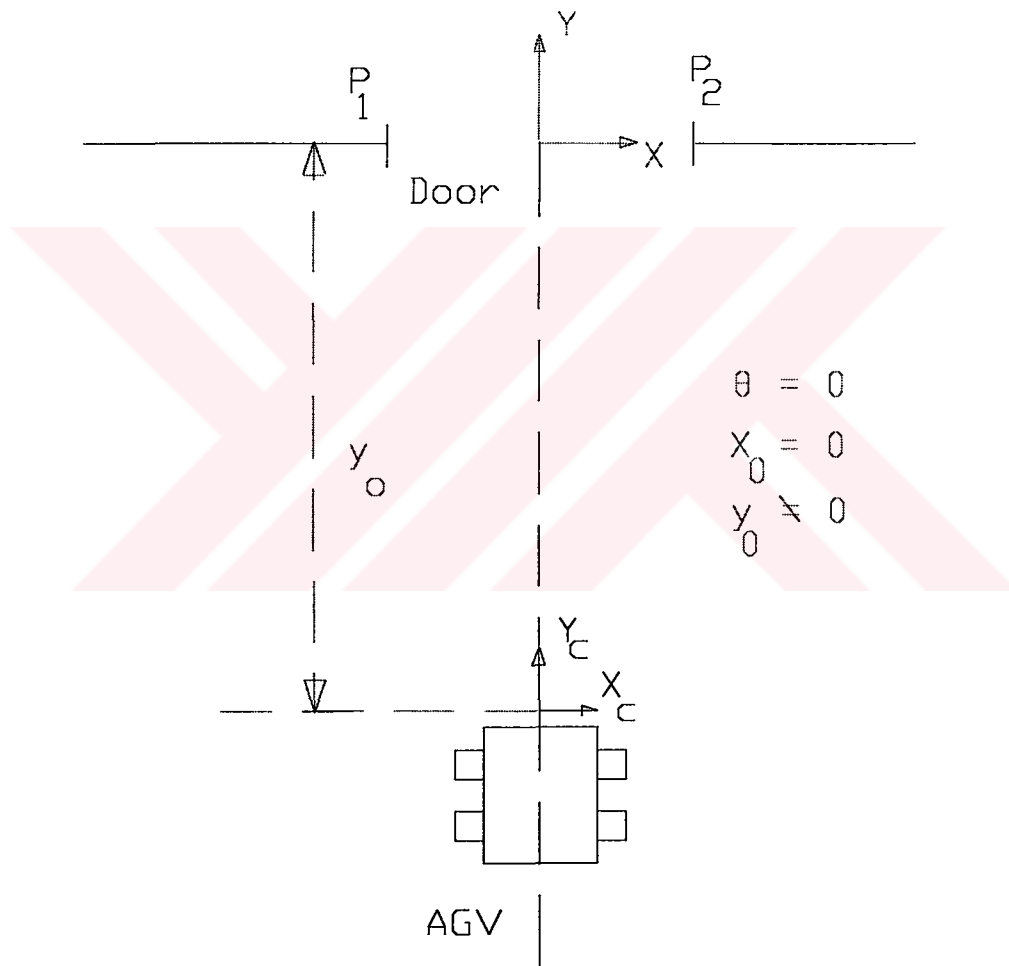


Figure 5.3 Experiment Type 1.

Image Size : 256 x 256 pixels
 Sensor Size : 6.3 mm x 4.7 mm
 Camera Height : 900 mm
 Door Size : 1070 mm x 2100 mm

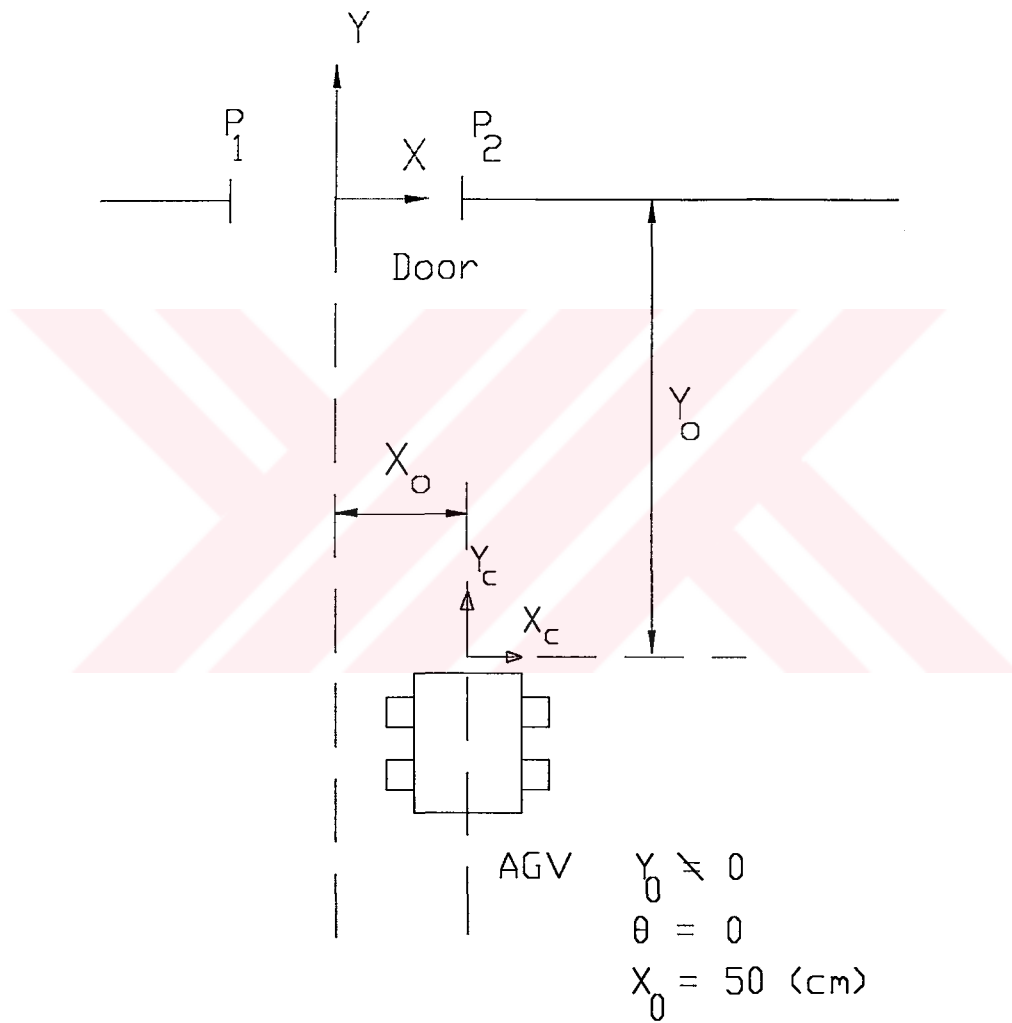


Fig 5.4 Experiment Type 2.

Image Size : 256 x 256 pixels
 Sensor Size : 6.3 mm x 4.7 mm
 Camera Height : 900 mm
 Door Size : 1070 mm x 2100 mm

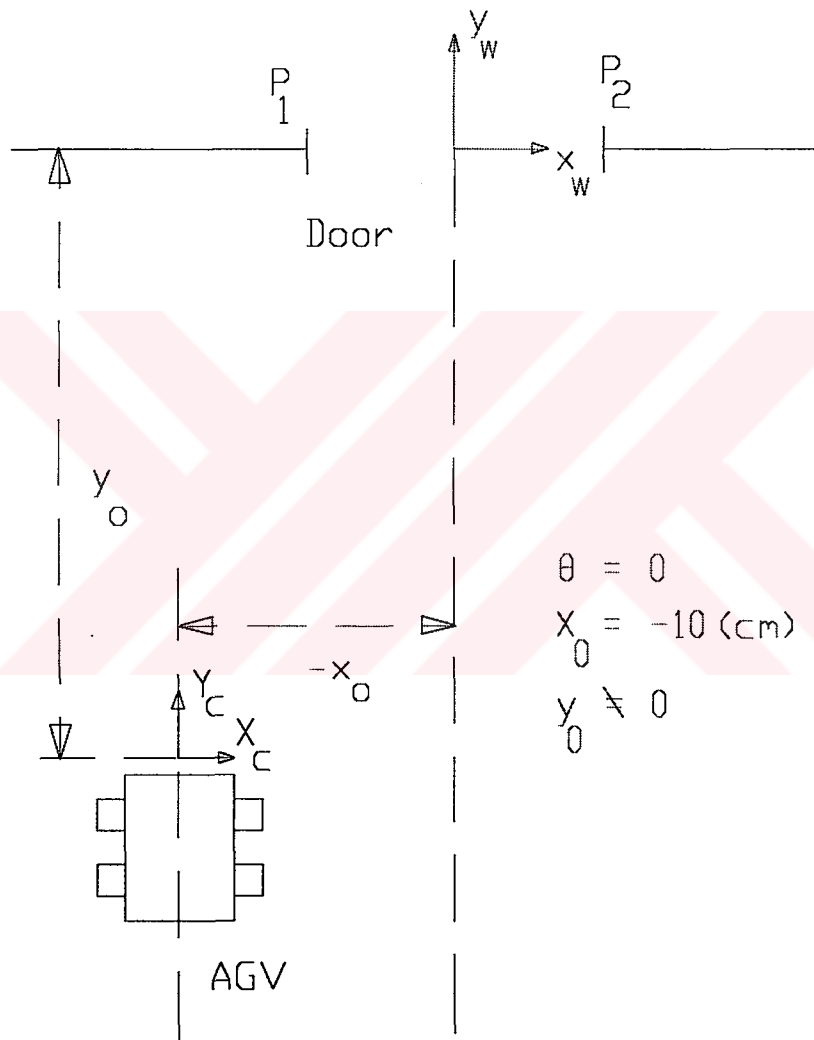


Figure 5.5 Experiment Type 3.

5.5 Presentation of the Experimental Results

In these experiments, Mobile robot location as it is calculated by using camera viewing parameters is compared to the measured position of the mobile robot.

Table 5.2 and Table 5.3 show the experimental results according to Liu and Huang's Method (LHM) and Newton Raphson Method algorithms (NRM) respectively, and Table 5.4 shows the absolute % errors in Y_o under Experiment Type 1 and Set 1. Similarly, for Set 2 under Experiment Type 1, Table 5.5 and Table 5.6 indicate the experimental results according to LHM and NRM respectively, Table 5.7 shows the absolute % errors in Y_o . Table 5.2, Table 5.3, Table 5.5 and Table 5.6 indicate the true position values against the computed values of three mobile robot position parameters. The computed results are obtained from the two algorithms mentioned before. The error is defined as the difference between the measured data and the computed results. Under Experiment Type 1 for both Set1 and Set 2, the distance error is in the range of -4 cm to -18 cm for LHM, -10 cm and -25 cm for NRM if Y_o is between 400cm to 500 cm, and no larger than -38 cm if it is between 500 cm and 700 cm for both LHM and NRM algorithms, it is shown in Figure 5.6 and 5.7. Table 5.4 and Table 5.7 show the absolute % errors in Y_o . For LHM, the maximum absolute % Y_o error is less than 5%. The performance of NRM algorithm is very similar to that of the LHM algorithm. The maximum absolute % Y_o error of NRM is smaller than 6% as it is clear from in Figure 5.8 and 5.9.

Table 5.2 Experimental Results according to Liu and Huang's Method for Experiment Type 1 and Set 1.

True Value			Computed Value			Error		
Yo (cm)	Xo (cm)	θ (°)	Yo (cm)	Xo (cm)	θ (°)	Yo (cm)	Xo (cm)	θ (°)
700	0	0	670.2	29	6.01	-29.8	29	6.01
650	0	0	627.89	6.85	-1.88	-22.11	6.85	-1.88
600	0	0	583.29	20.93	5.25	-16.71	20.9	5.25
550	0	0	523.14	3.76	-0.78	-26.86	3.76	-0.78
500	0	0	482.66	18.51	-1.45	-17.34	18.51	-1.45
450	0	0	444.41	5.20	0.00	-5.59	5.2	0.00
400	0	0	395.42	7.16	0.20	-4.58	7.16	0.20

Table 5.3 Experimental Results according to Newton Raphson Method for Experiment Type 1 and Set1.

True Value			Computed Value			Error		
Yo (cm)	Xo (cm)	θ (°)	Yo (cm)	Xo (cm)	θ (°)	Yo (cm)	Xo (cm)	θ (°)
700	0	0	662.24	44.41	5.58	-37.76	44.41	5.58
650	0	0	622.51	5.90	-1.64	-27.49	5.90	-1.64
600	0	0	573.83	30.24	4.23	-26.17	30.24	4.23
550	0	0	516.85	4.80	-1.14	-33.15	4.80	-1.14
500	0	0	475.40	23.56	1.93	-24.60	23.56	1.93
450	0	0	436.99	5.20	0.00	-13.01	5.20	0.00
400	0	0	386.64	7.86	0.30	-13.36	7.86	0.30

Table 5.4 Absolute % Errors in Yo for Experiment Type 1 and Set 1

Yo (cm)	Absolute % Yo Error	
	Liu & Huang's	Newton Raphson
700	4.25	5.40
650	3.41	4.23
600	2.79	4.36
550	4.88	6.03
500	3.47	4.92
450	1.24	2.89
400	1.83	3.34

Table 5.5 Experimental Results according to Liu and Huang's Method for Type 1 and Set 2.

True Value			Computed Value			Error		
Yo (cm)	Xo (cm)	θ (°)	Yo (cm)	Xo (cm)	θ (°)	Yo (cm)	Xo (cm)	θ (°)
700	0	0	678.50	7.35	1.02	-21.5	7.35	1.02
650	0	0	620.08	-8.83	1.86	-29.92	-8.83	1.86
600	0	0	577.26	3.35	-0.86	-22.74	3.35	-0.86
550	0	0	528.76	4.06	-2.38	-21.24	4.06	-2.38
500	0	0	482.82	7.20	0.72	-17.18	7.20	0.72
450	0	0	436.70	1.06	-0.65	-15.20	1.06	-0.65
400	0	0	385.75	4.94	-1.13	-14.85	4.94	-1.13

Table 5.6 Experimental Results according to Newton Raphson Method for Experiment Type 1 and Set 2.

True Value			Computed Value			Error		
Yo (cm)	Xo (cm)	θ (°)	Yo (cm)	Xo (cm)	θ (°)	Yo (cm)	Xo (cm)	θ (°)
700	0	0	673.36	29.11	2.10	-26.64	29.11	2.10
650	0	0	615.71	23.99	3.51	-34.29	23.99	3.51
600	0	0	572.20	9.88	-1.53	-27.80	9.88	-1.53
550	0	0	523.62	2.28	-1.28	-26.38	2.28	-1.28
500	0	0	476.68	7.20	1.07	-24.97	7.20	1.07
450	0	0	430.28	4.35	-0.88	-21.64	4.35	-0.88
400	0	0	378.51	2.51	0.66	-21.49	2.51	0.66

Table 5.7 Absolute % Errors in Yo for Experiment Type 1 and Set 2.

Yo (cm)	Absolute %Yo Error	
	Liu & Huang's	Newton Raphson
700	3.07	3.8
650	4.60	5.27
600	3.79	4.63
550	3.86	4.79
500	3.43	4.99
450	3.38	4.80
400	3.71	5.37

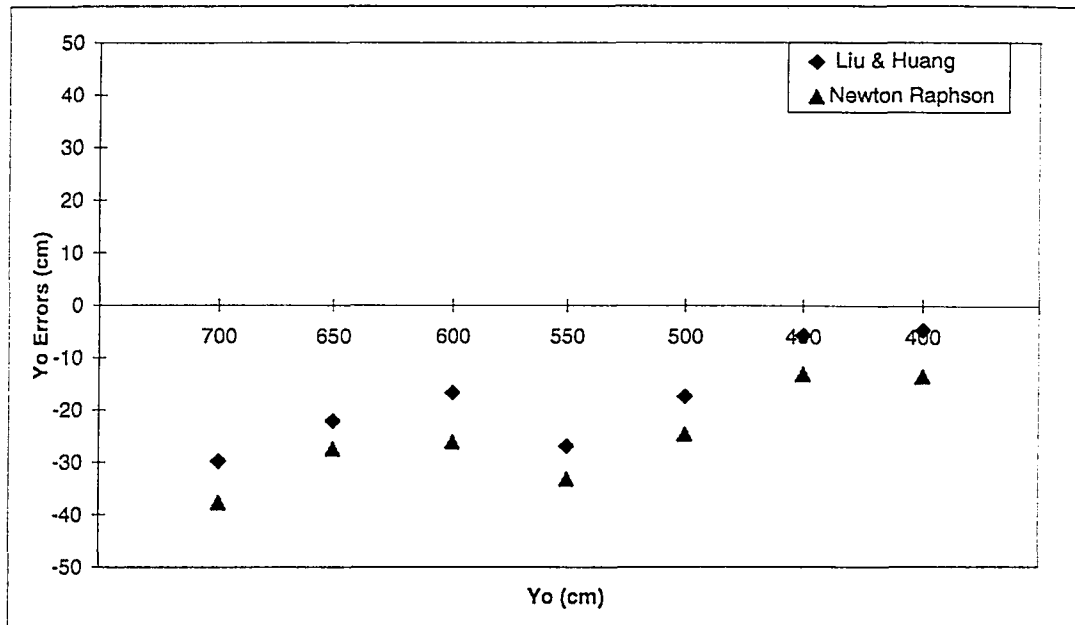


Figure 5.6 Yo Errors for Experiment Type 1 and Set 1

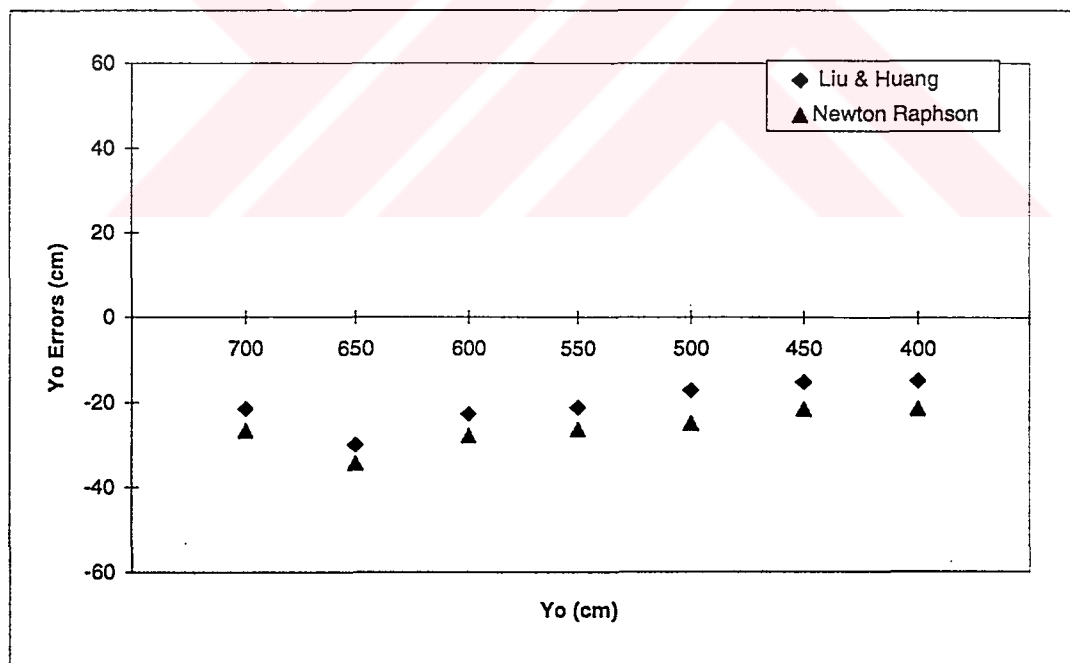


Figure 5.7 Yo Errors for Experiment Type 1 and Set 2

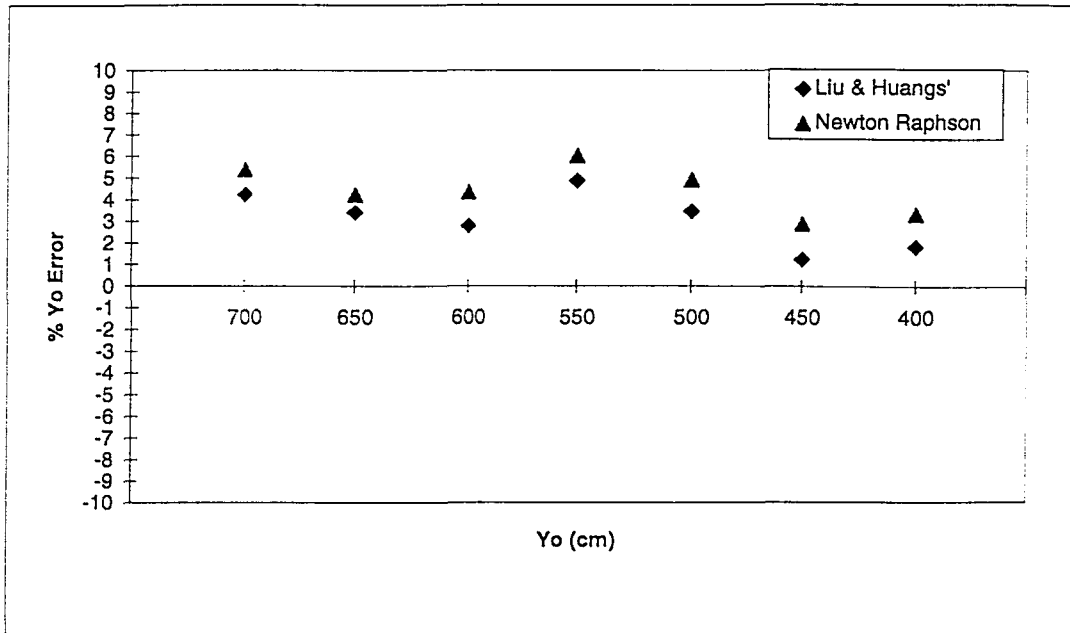


Figure 5.8 Absolute % Errors in Y_o for Experiment Type 1 and Set 1

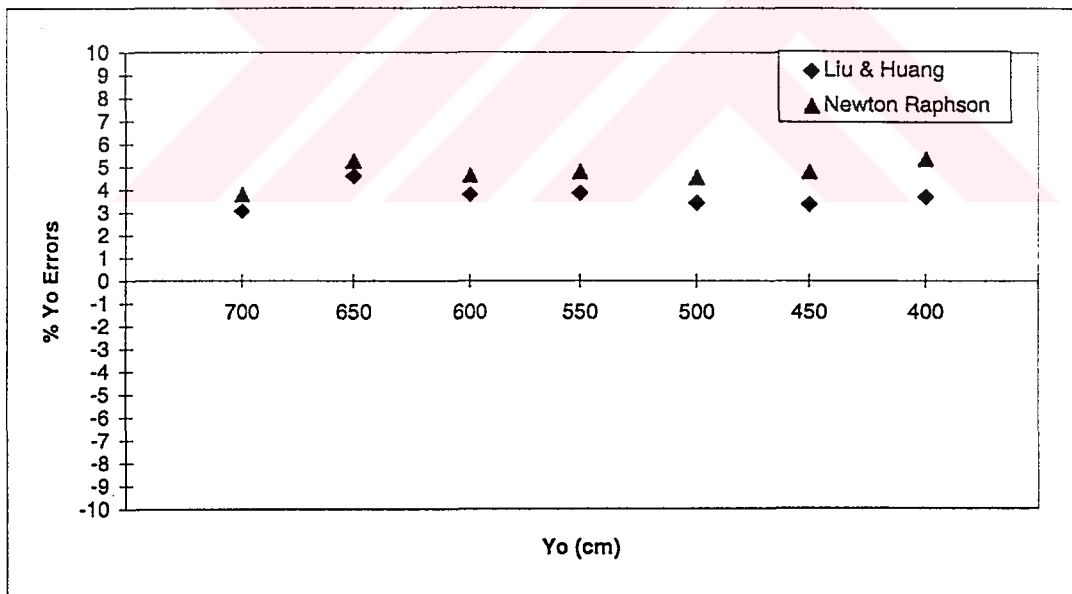


Figure 5.9 Absolute % Errors in Y_o for Experiment Type1 and Set 2

As indicated Experiment Type 1, Table 5.8 and Table 5.9 show the experimental results of mobile robot location parameters according to LHM and NRM respectively, and Table 5.10 shows the absolute % errors in Y_o under Experiment Type 2 and Set 3. For Set 4 under Experiment Type 2, Table 5.11 and Table 5.12 indicate the experimental results according to LHM and NRM respectively, Table 5.13 shows the absolute % errors in Y_o .

Under Experiment Type 2 both Set 3 and Set 4, the distance Y_o error is in the range of -17 cm to -23 cm if Y_o is from 700 cm to 600 cm and -3 cm to -18 cm if the distance Y_o is between 600 cm and 450 cm for LHM algorithm. About NRM, the Y_o error is smaller than -37 cm if the distance Y_o is in the range of 700 cm to 600 cm and it is no larger than 26 cm if Y_o is between 600 cm and 450 cm, it is shown in Table 5.8, Table 5.9, Table 5.11, Table 5.12 and Figure 5.10, Figure 5.11.

Again it should be noticed that the maximum absolute % Y_o error is less than 4% for LHM and 5.5% for NRM algorithms. These are shown in Table 5.10, Table 5.14 and Figure 5.12, Figure 5.13 for both algorithms and both Set 3 and Set 4 under Type2.

Finally, Type 3 is considered as a last experiment. In this experiment type for Set 5, Table 5.14 and Table 5.15 show the true values and computed values of three mobile robot parameters for LHM and NRM algorithms respectively. Also Table 5.16 indicate absolute % errors in Y_o .

Table 5.8 Experimental Results according to Liu and Huang's Method for Experiment Type 2 and Set 3.

True Value			Computed Value			Error		
Yo (cm)	Xo (cm)	θ (°)	Yo (cm)	Xo (cm)	θ (°)	Yo (cm)	Xo (cm)	θ (°)
700	50	0	676.68	71.18	4.06	-23.32	21.18	4.06
650	50	0	626.75	62.65	1.88	-23.25	12.65	1.88
600	50	0	582.47	67.61	2.62	-17.53	17.61	2.62
550	50	0	546.25	53.88	-0.82	-3.75	3.88	-0.82
500	50	0	482.19	55.51	1.45	-17.81	5.51	1.45
450	50	0	444.12	51.95	0.67	-5.88	1.95	0.67

Table 5.9 Experimental Results according to Newton Raphson Method for Type 2 and Set 3.

True Value			Computed Value			Error		
Yo (cm)	Xo (cm)	θ (°)	Yo (cm)	Xo (cm)	θ (°)	Yo (cm)	Xo (cm)	θ (°)
700	50	0	663.80	87.59	4.17	-36.20	37.59	4.17
650	50	0	619.16	76.93	1.8	-30.84	26.93	1.8
600	50	0	575.38	76.56	1.56	-24.62	26.56	1.56
550	50	0	543.27	42.8	-1.37	-6.73	-7.2	-1.37
500	50	0	475.00	61.32	1.07	-25.00	11.32	1.07
450	50	0	436.48	57.56	0.90	-21.64	7.56	0.9

Table 5.10 Absolute % Errors in Yo for Experiment Type 2 and Set 3

Yo (cm)	% Yo Error	
	Liu & Huang's	Newton Raphson
700	3.34	5.17
650	3.58	4.74
600	2.92	4.10
550	0.68	1.22
500	3.56	5.00
450	1.31	4.80

Table 5.11 Experimental Results according to Liu and Huang's Method for Experiment Type 2 and Set 4.

True Value			Computed Value			Error		
Yo (cm)	Xo (cm)	θ (°)	Yo (cm)	Xo (cm)	θ (°)	Yo (cm)	Xo (cm)	θ (°)
700	50	0	678.63	53.5	0	-21.37	3.50	0
650	50	0	625.85	62.43	3.76	-24.15	12.43	3.76
600	50	0	578.13	57.76	-1.73	-21.87	7.76	-1.73
550	50	0	539.57	60.00	0.81	-10.43	10.00	0.81
500	50	0	492.01	51.94	0.74	-7.99	1.94	0.74

Table 5.12 Experimental Results according to Newton Raphson Method for Experiment Type 2 and Set 4.

True Value			Computed Value			Error		
Yo (cm)	Xo (cm)	θ (°)	Yo (cm)	Xo (cm)	θ (°)	Yo (cm)	Xo (cm)	θ (°)
700	50	0	673.76	53.5	0.00	-26.24	3.50	0.00
650	50	0	614.45	86.95	3.24	-35.55	36.95	3.24
600	50	0	574.13	48.19	-1.39	-25.87	-1.81	-1.39
550	50	0	531.22	69.27	1.21	-18.78	19.27	1.21
500	50	0	483.77	58.85	1.00	-16.23	8.85	1.00

Table 5.13 Absolute % Errors in Yo for Experiment Type 2 and Set 4

Yo (cm)	% Yo Error	
	Liu & Huang's	Newton Raphson
700	3.05	3.75
650	3.77	5.47
600	3.36	4.31
550	1.89	3.41
500	1.6	3.25

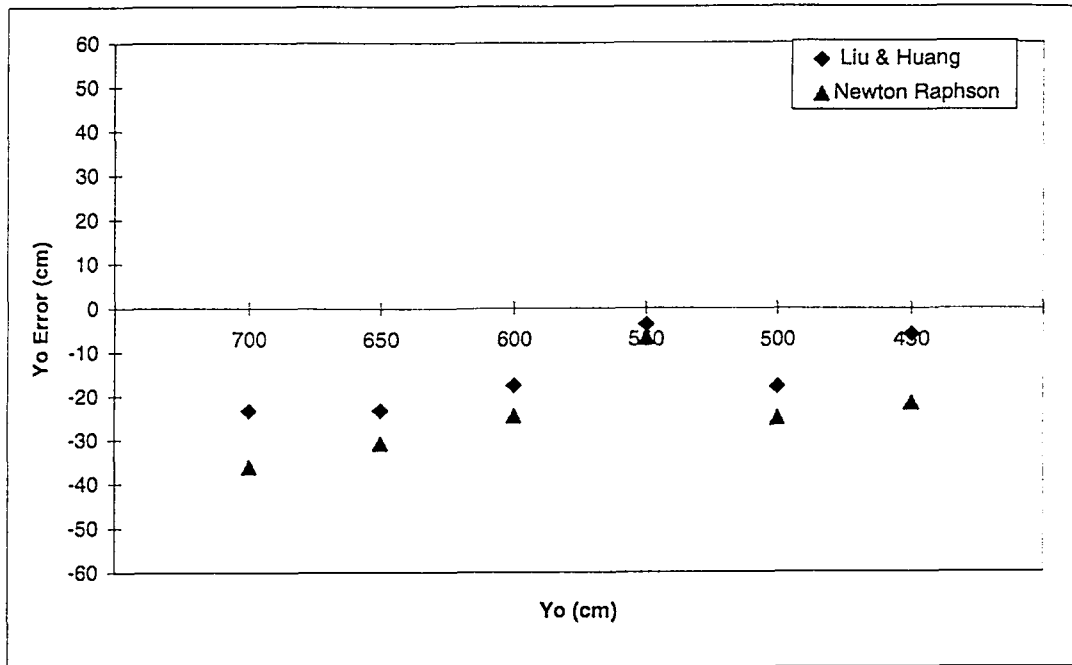


Figure 5.10 Yo Errors for Experiment Type 2 and Set 3.

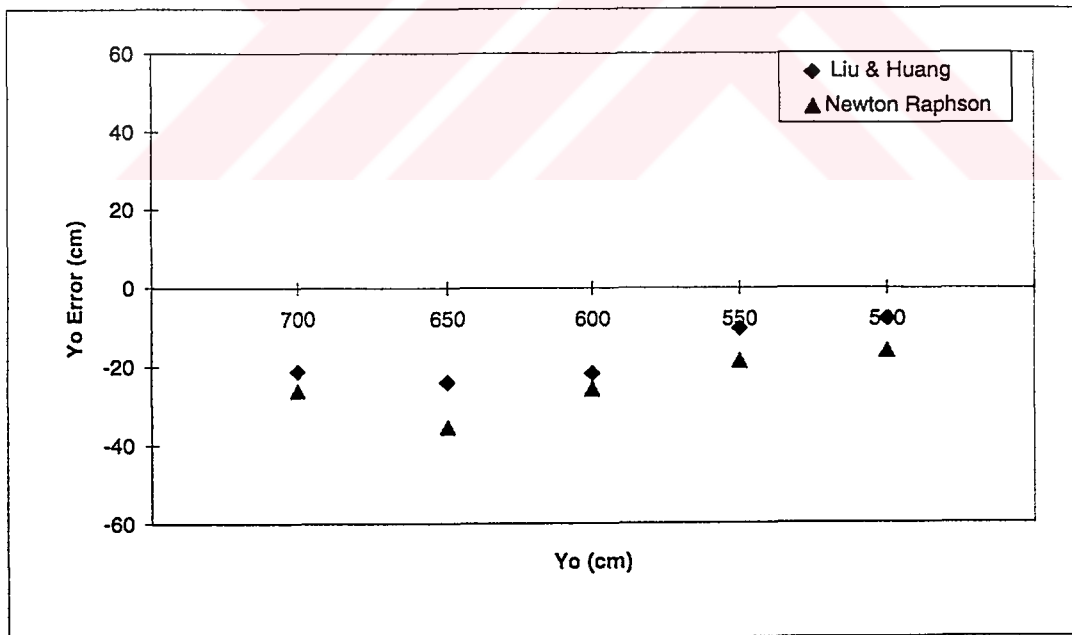


Figure 5.11 Yo Errors for Experiment Type 2 and Set 4.

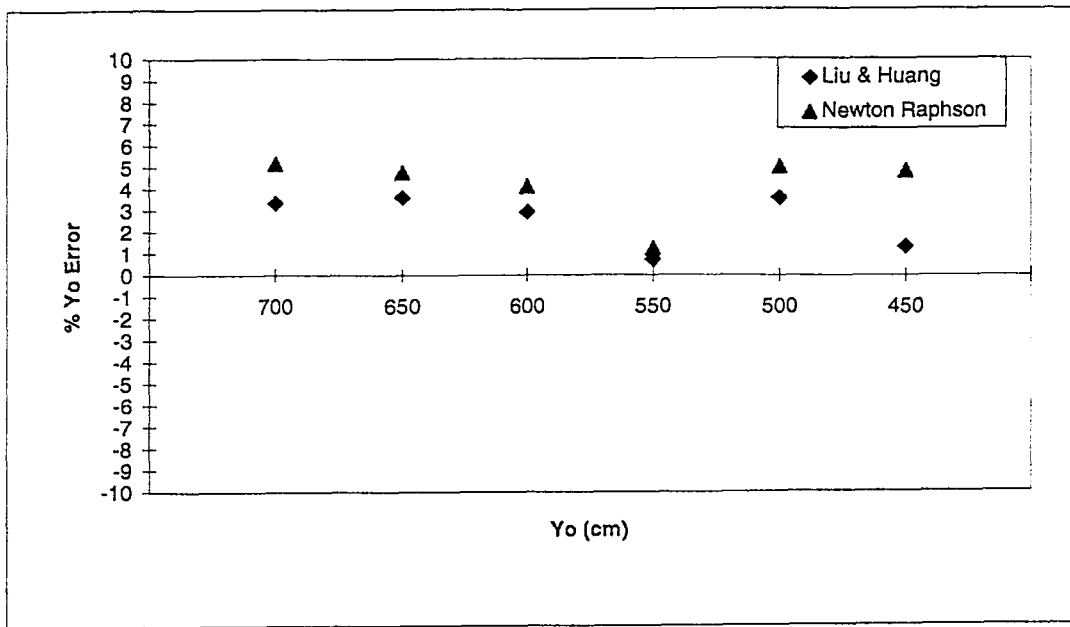


Figure 5.12 Absolute % Errors in Y_o for Experiment Type 2 and Set 3.

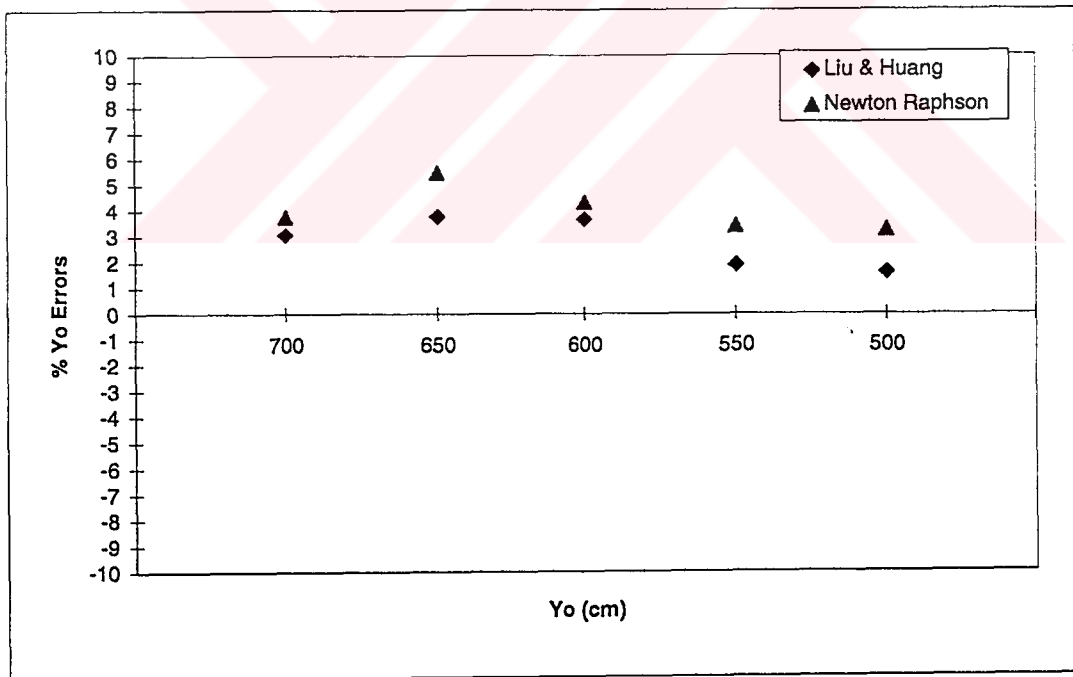


Figure 5.13 Absolute % Errors in Y_o for Experiment Type 2 and Set 4.

Table 5.14 Experimental Results according to Liu and Huangs' Method for Experiment Type 3 and Set 5.

True Value			Computed Value			Error		
Yo (cm)	Xo (cm)	θ (°)	Yo (cm)	Xo (cm)	θ (°)	Yo (cm)	Xo (cm)	θ (°)
750	-10	0	727.67	-16.6	-1.09	-22.33	-6.60	-1.09
700	-10	0	697.03	-18.73	-1.18	-2.97	-8.73	-1.18
650	-10	0	627.41	-22.57	-1.88	-22.59	-12.57	-1.88
600	-10	0	577.47	-7.55	1.59	-22.53	2.45	1.59
550	-10	0	528.54	-12.54	-0.79	-21.46	-2.54	-0.79
500	-10	0	478.21	-11.19	-0.72	-21.79	-1.19	-0.72

Table 5.15 Experimental Results according to Newton Raphson Method for Experiment Type 3 and Set 5.

True Value			Computed Value			Error		
Yo (cm)	Xo (cm)	θ (°)	Yo (cm)	Xo (cm)	θ (°)	Yo (cm)	Xo (cm)	θ (°)
750	-10	0	720.87	-40.98	-2.20	-29.13	-30.98	-2.20
700	-10	0	688.27	-54.67	-4.03	-11.73	-44.67	-4.03
650	-10	0	620.74	-35.23	-1.64	-29.26	-25.23	-1.64
600	-10	0	571.68	-1.91	1.39	-28.32	8.09	1.39
550	-10	0	521.60	-21.37	-1.16	-28.4	-11.37	-1.16
500	-10	0	476.69	-17.58	-0.95	-23.31	-7.58	-0.95

Table 5.16 Absolute % Errors in Yo for Experiment Type 3 and Set 5

Yo (cm)	% Yo Error	
	Liu & Huangs'	Newton Raphson
750	2.98	3.88
700	0.42	1.68
650	3.47	4.50
600	3.75	4.72
550	3.9	5.16
500	4.35	4.66

According to Table 5.14 and Figure 5.14, the maximum Y_o error is no larger than -23 cm for Type 3 and Set 5. About Table 5.15 and Figure 5.14, the Y_o error is between -11 cm and -30 cm for NRM under Type 3 and Set5.

The maximum absolute % Y_o error under Type 3 and Set 5 is smaller than 4.5% for LHM and 5.5 % for NRM, these are indicated in Table 5.16 and Fig 5.15.

The X_o deteriorates as the orientation (θ) error increases. For the orientation and X_o parameters, the performance of the LHM and the NRM algorithms are similar but LHM is better than that of the NRM algorithm under high direction errors and between 7.5 m and 6 m of the working distance. These results are depicted in Figure 5.16, 5.17, 5.18, 5.19, and 5.20. These Figures show the X_o and Orientation Errors and when the absolute orientation errors increase, the X_o errors climb sharply.

All these errors are believed to develop because of the following reasons:

- It was assumed that the camera is ideal at the beginning of the thesis. All coefficients of the lens distortion parameters are assumed to be zero. The camera used in the experiments is a black and white CCD camera and it is not ideal in the sense of the assumptions.

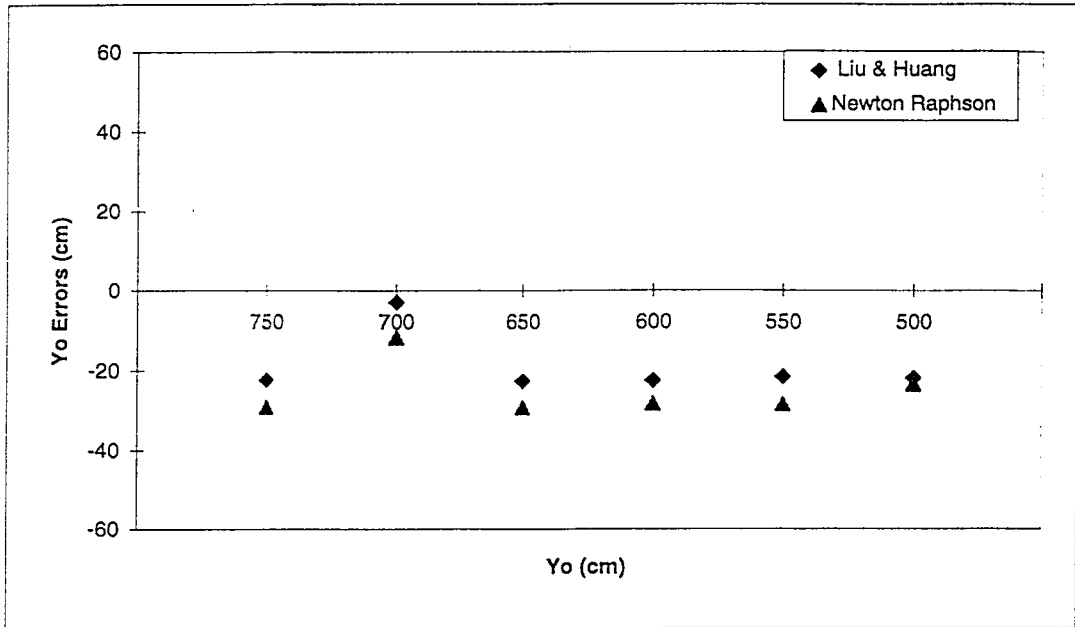


Figure 5.14 Yo Errors for Experiment Type 3 and Set 5.

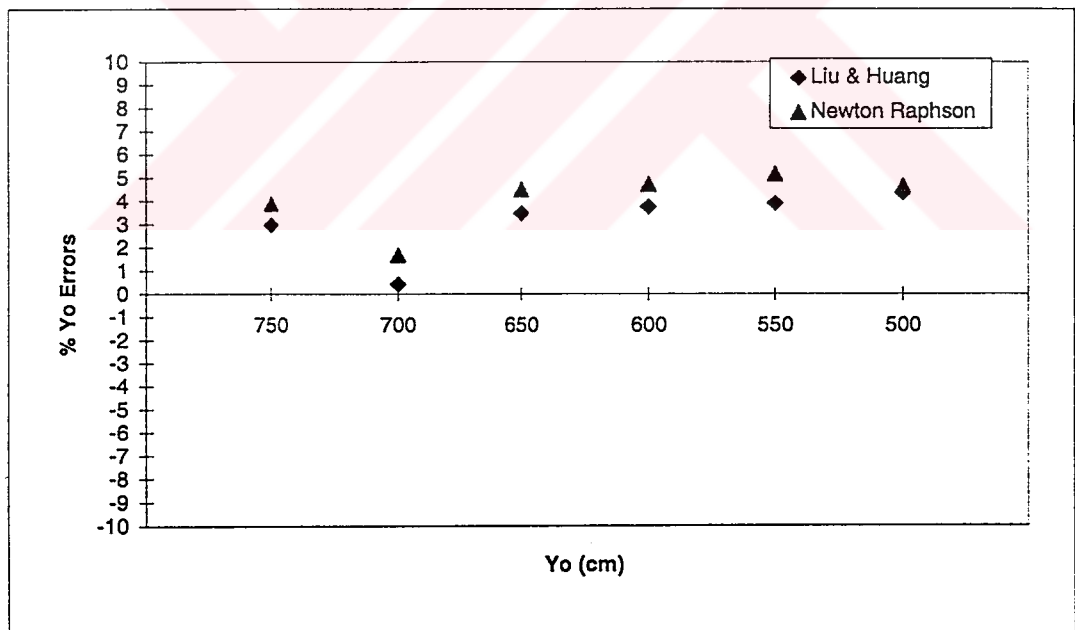
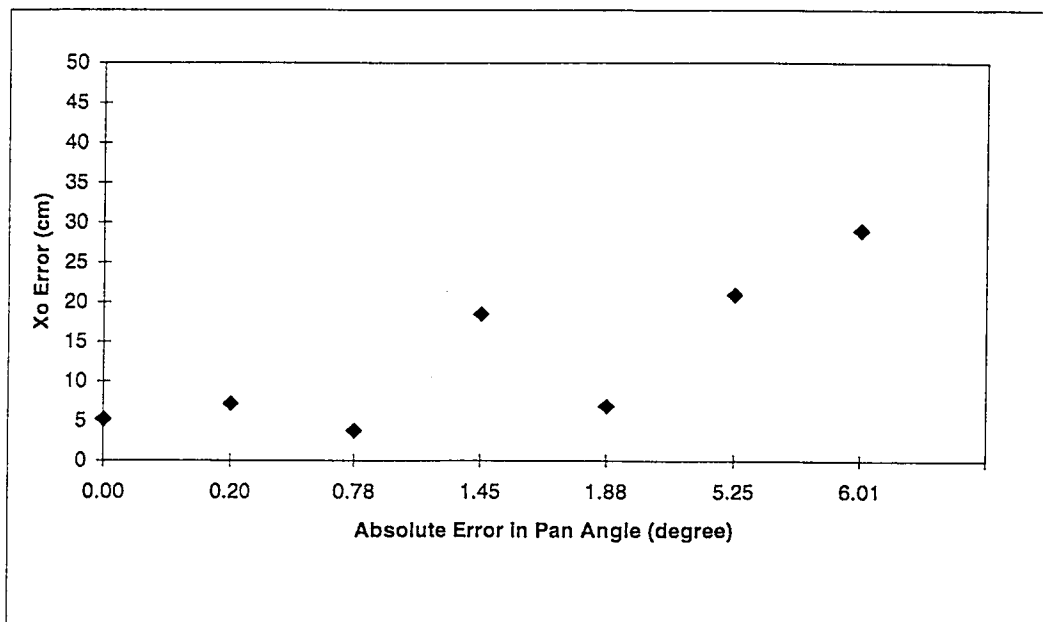
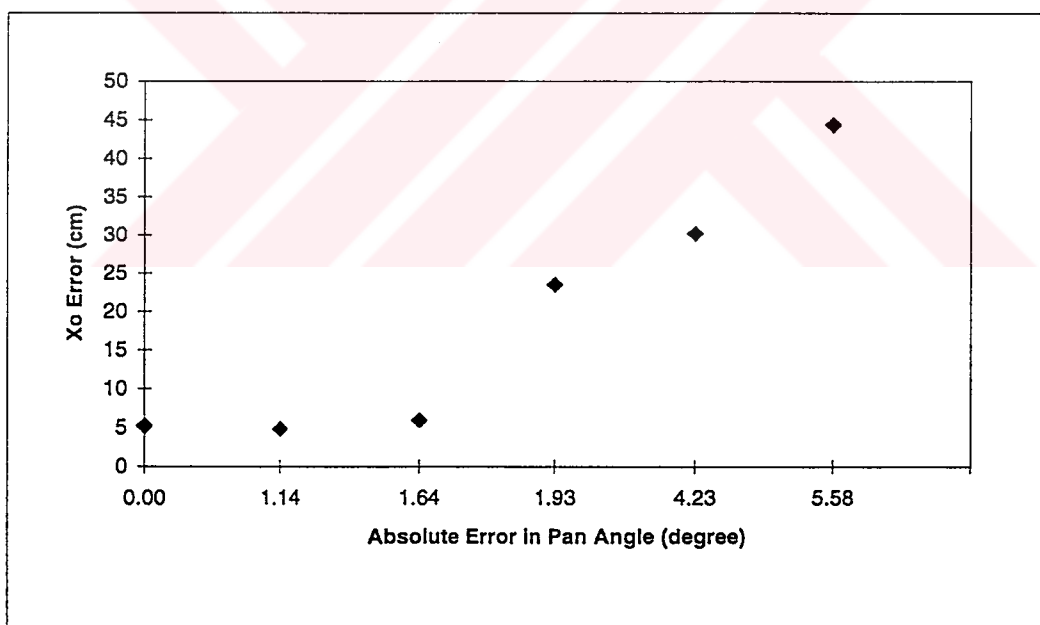


Figure 5.15 Absolute % Errors in Yo for Experiment Type 3 and Set 5.

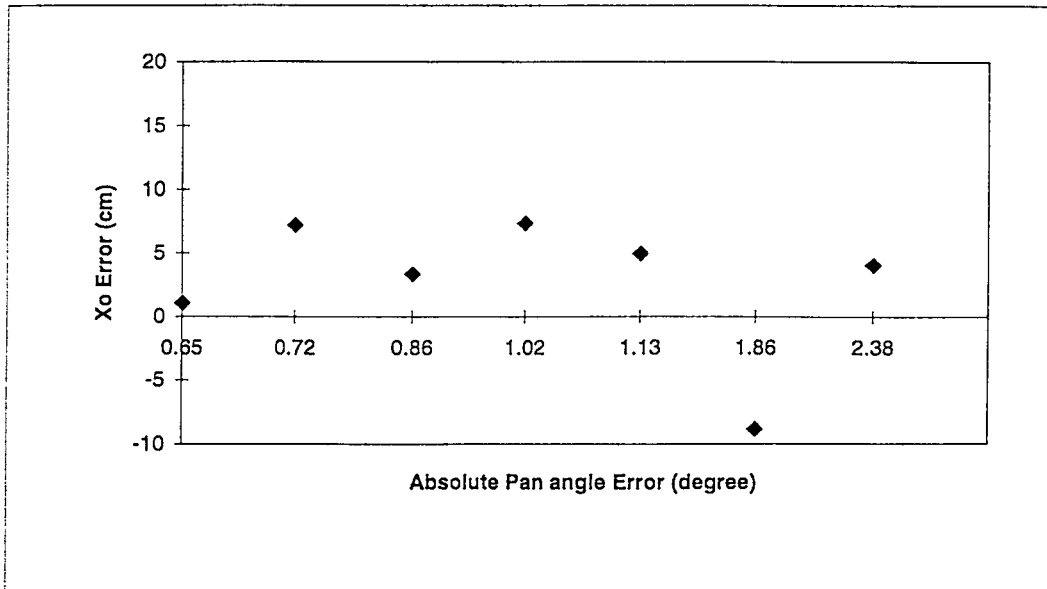


(a)

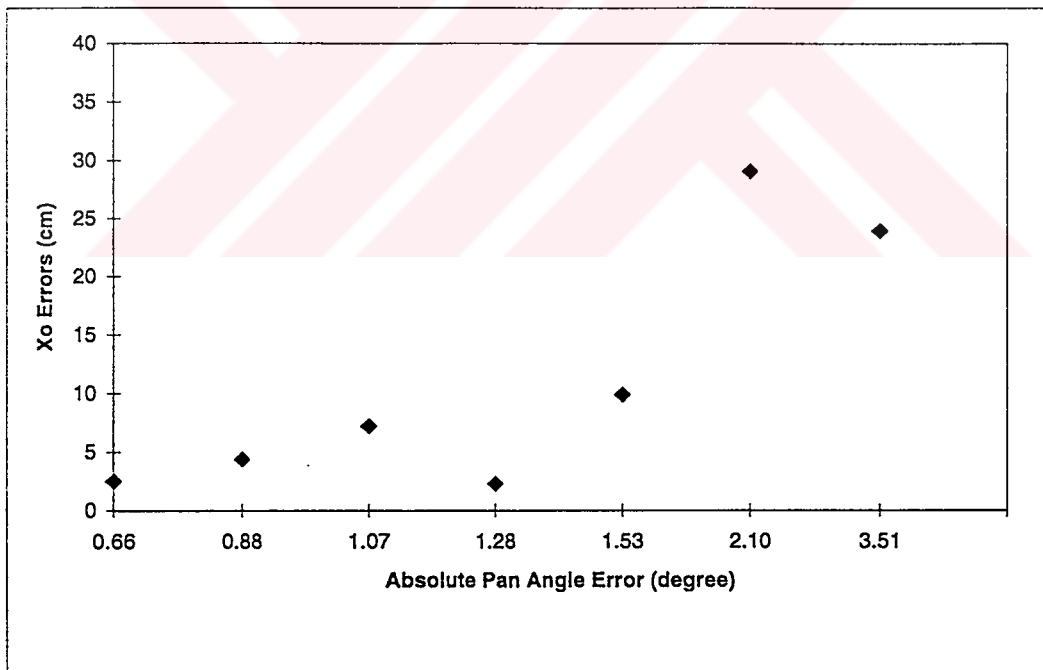


(b)

Fig 5.16 X₀ and Orientation Errors according to (a) LHM; (b) NRM for Experiment Type 1 and Set 1.

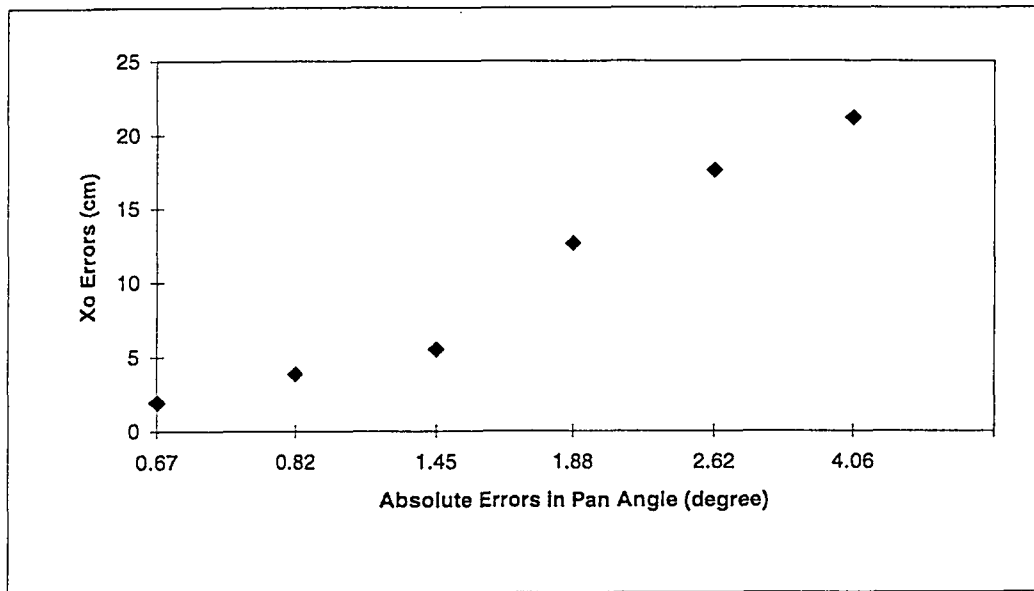


(a)

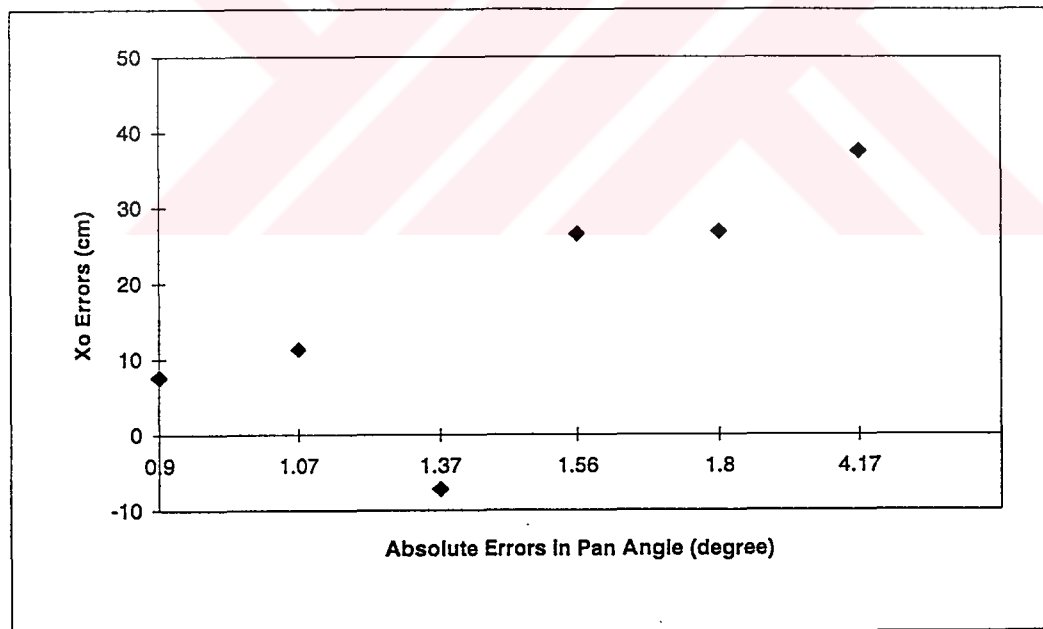


(b)

Figure 5.17 X0 and Orientation Errors according to (a) LHM; (b) NRM for Experiment Type 1 and Set 2.

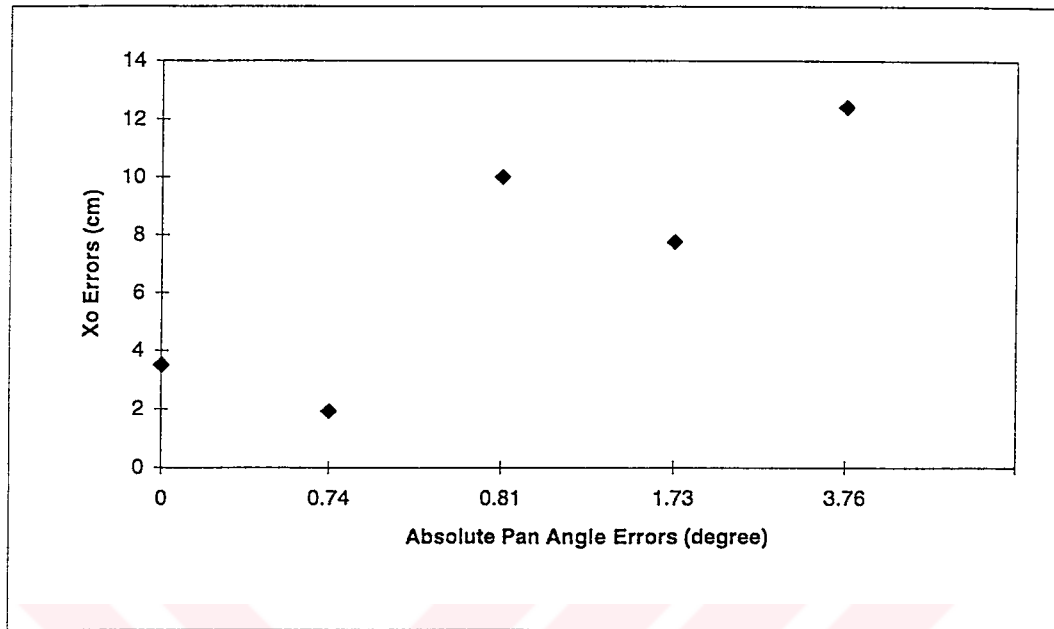


(a)

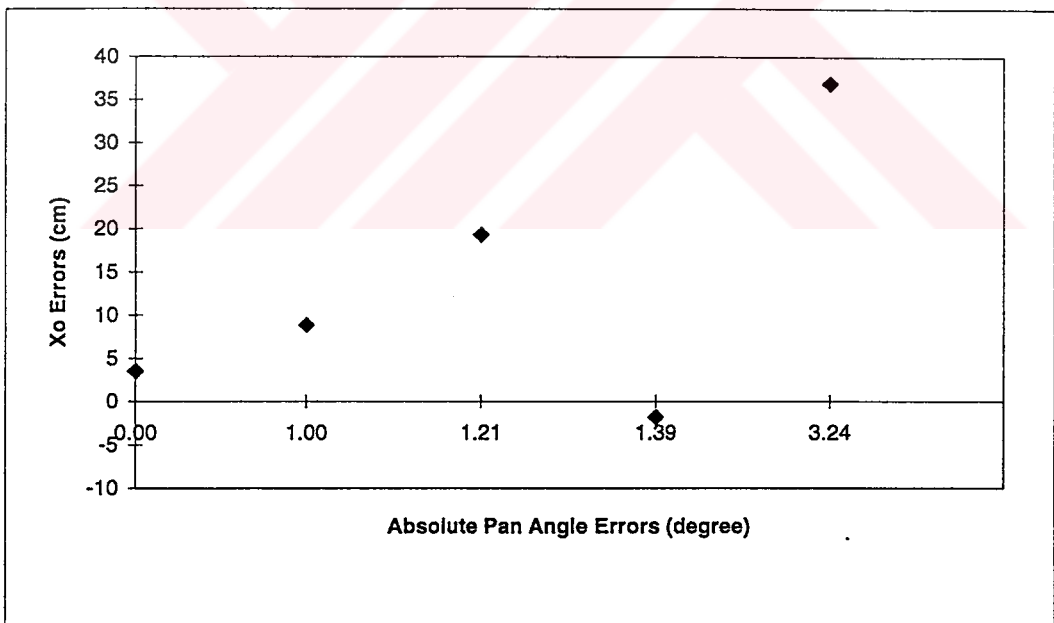


(b)

Figure 5.18 X0 and Orientation Errors according to (a) LHM; (b) NRM for Experiment Type 2 and Set 3.

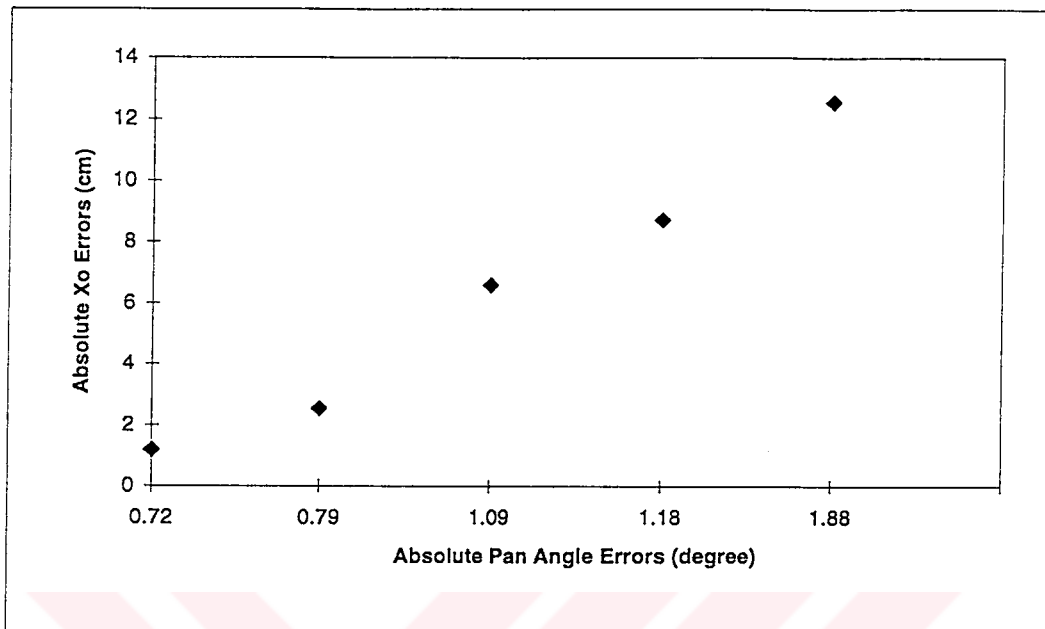


(a)

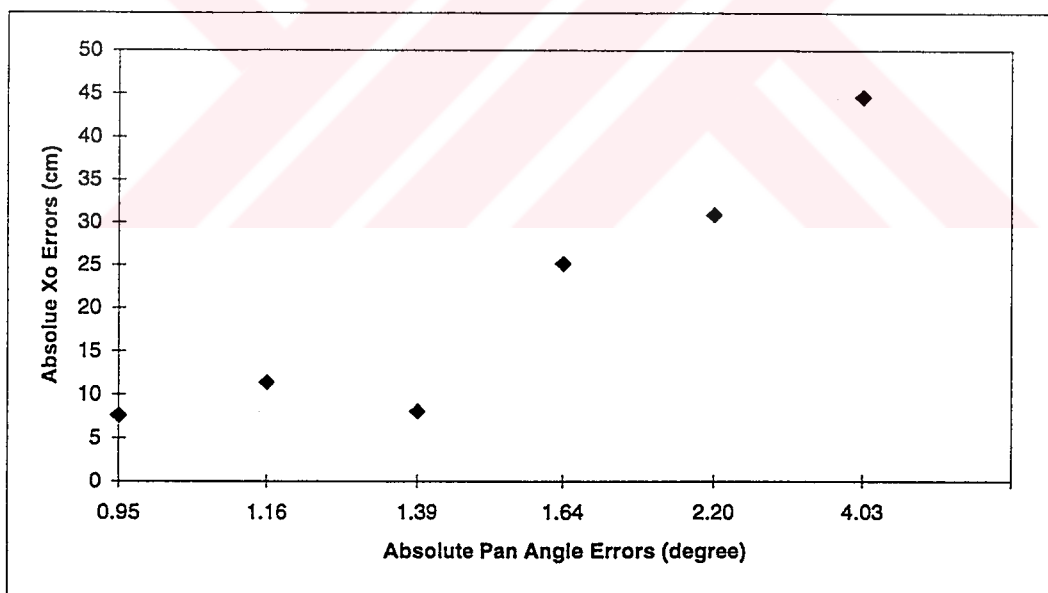


(b)

Figure 5.19 X0 and Orientations Errors according to (a) LHM; (b) NRM for Experiment Type 2 and Set 4.



(a)



(b)

Figure 5.20 Xo and Orientation Errors according to (a) LHM; (b) NRM for Experiment Type 3 and Set 5.

- Dimensional construction of the test vehicle is not highly accurate. The height of the camera is fixed. The tilt and swing angles of the camera head are assumed zero.
- Noise in the image is inevitable during the experiments because of surroundings effects.
- To calculate the mobile robot position, we use door as an object at indoor environments.
- In this study, the illumination is not controllable. If possible, the vision system should be able to adapt to changes of illumination. One solution is to use light meters and adjusting the illumination system.
- Performance of the distance estimation depends on performances of the level image processing and analyses algorithms of this project.

CHAPTER 6

DISCUSSIONS AND CONCLUSIONS

Determination of a mobile robot location by using a camera is an inevitable step in robot vision, mobile robot navigation, automated visual inspection and many other similar tasks if the camera is the only sensor. The objective of this work is to calculate the camera viewing parameters from 2D image of a door shape by using a single CCD camera. In order to determine location of the mobile robot, the camera viewing parameters must be determined. In this respect, two algorithms for determination of camera location are presented and implemented. These are Newton Raphson Method (NRM) and Liu and Huang's Method (LHM).

In the navigation guidance phase, the test vehicle is moved on the floor to simulate the mobile robot navigation. The navigation experiments are performed at indoors environments and the range of working distance is taken between 7.5m to 4m. Experiments are performed to determine performance of the system for three different conditions. Errors of the mobile robot location parameters are calculated by comparing the true position values (measured) against the computed values. Finally, these results are discussed and sources of errors are explained. Main advantage

of these two methods is that only two points in the image as an input are sufficient to calculate the related parameters of the mobile robot locations.

During the experiments the following conclusions are drawn:

- Absolute % Error in Y_o (translation on Y direction of the camera or distance from door to a mobile robot) is less than 5% for Liu and Huang's Algorithm.
- Generally, maximum Y_o error is less than -30 cm between 7.5 m and 6 m. From 6m to 4 m, it is smaller than -25 cm for LHM.
- Absolute % Error in Y_o is less than 6% for Newton Raphson Method (NRM) solution.
- Generally, maximum Y_o error is less than -35 cm for NRM.
- The accuracy of X_o (translation on the X direction of the camera) deteriorates as the orientation θ error increases.
- Generally, in the range of 7.5m and 6m, the errors of X_o and orientation parameters are larger than the errors for the range of 6m and 4m.
- Main cause of errors is the believed to be uncontrolled illumination.

In this work, the camera is assumed to be ideal at the beginning of the thesis. Methods for estimating the mobile robot location are developed and implemented based on this assumption. As a future work, all internal camera parameters can be considered. The methods mentioned before may be developed with both external and internal camera parameters. The internal camera parameters is specified by the camera constant f , the distance between the image plane and the camera lens; (C_x, C_y) , positions of the optical center on the sensor plane; (k_1, k_2) first and second items of radial lens distortion coefficient.

It is observed that the main deficiency of the system is illumination. This problem can be eliminated by using a new camera with adjustable and automatic settings.

The moving platform and camera head (tripod) can be constructed more accurate. This head should be able to control height and tilt angle of the camera in real time.

Software development of the present study was develop under DOS environment. Some base memory problems are observed during the study. For next study, UNIX environment may be used.

REFERENCES

- [1] R.D. Schraft, "Mechatronics and Robotics for Service Applications", IEEE Robotics & Automation Magazine, pp. 31-37 (December, 1994).
- [2] G. Haris, "Robotics Decontamination Keeps Operators Clear of Danger", Industrial Robot, Vol.20, No.3, pp.30-33 (1993).
- [3] C.R. Weisbin and D.Lavery, "Nasa Rover and Telerobotics Technology Program", IEEE Robotics & Automation Magazine, pp.14-21 (December, 1994).
- [4] J.Evans, B. Krishnamurty and B. Barrows, "Handling Real-World Motion Planning: A Hospital Transport Robot", IEEE Control Systems, pp. 15-19 (February, 1992).
- [5] T. Orwing, "Cybermotion's Roving Robots", Industrial Robot, Vol. 20, No. 3, pp. 26-29 (1993).
- [6] G. Giralt, "Mobile Robots", NATO ASI Series-Robotics and Artificial Intelligence ,Vol. F11, pp.365-393 (1984).
- [7] K.C. Drake, E.S. Mcvey and R.M. Inigo, "Sensing Error for a Mobile Robot Using Line Navigation", IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. Pami-7, No. 4, pp. 485-490 (Jully, 1985).
- [8] C. Kendi, "Design and Production of a Mobile Robot Prototype as a Technology Demonstrator", M.S. Thesis in Mechanical Engineering, METU (September, 1994).

- [9] C. Kendi and A. Erden, "Development of a Prototype Mobile Robots As Technology Demonstrator", 6th International Machine Design and Production Conferences, METU, 1994.
- [10] A. Alansal, " Development of An Image Processing System for A Mobile Robots", MS Thesis at METU, Fall 1996-1997.
- [11] E. Çokal, " Development of An Image Analyses System for A Mobile Robots", MS Thesis at METU, Fall 1996-1997.
- [12] M. Kaiser, V. Klingspor, J.R. Millan and M. Accame, "Using Machine Learning Techniques in Real-World Mobile Robots", Intelligent Robotic Systems, pp. 37- 45 (April, 1995).
- [13] H. Schneiderman and M. Nashman, "A Discriminating Feature Tracker for Vision-Based Autonomous Driving", IEEE Transactions on Robotics and Automation, Vol. 10, No. 6, pp 769-775 (December, 1994).
- [14] R.M. Haralick, "Determining Camera Parameters from the Perspective Projection of a Rectangle", Pattern Recognition, Vol. 22, No. 3, pp. 225-230 (1989).
- [15] R.Y. Tsai, "A Versatile Camera Calibration Technique for High-Accuracy 3D Machine Vision Metrology Using Off-the-Shelf TV Cameras and Lenses", IEEE Journal of Robotics and Automation, Vol. Ra-3, No. 4, (August, 1987).
- [16] R.M. Haralick, "Matching Wire Frame Objects from Their Two Dimensional Perspective Projections", Pattern Recognition, Vol. 17, No. 6, pp. 607-619 (1984)
- [17] R.M. Haralick, "Solving Camera Parameters from the Perspective Projection of a Parameterized Curve", Pattern Recognition, Vol. 17, No. 6, pp. 637-645 (1984).

- [18] B.E. Platin, Z.Gan and G. Olgac, "3-D Object Configuration Sensor Utilizing Single-Camera Images", The American Society of Mechanical Engineers, 90-WA/DSC-22 (November,1990).
- [19] U.Ercan Acar," 3-D Position Reconstruction of Bodies Using Monocular Images", MS Thesis at METU, Fall 1995-1996.
- [20] Z. Chen, D. Tseng and J. Lin, "A Simple Vision Algorithm for 3-D Position Determination Using a Single Calibration Object", Pattern Recognition, Vol. 22, No. 2, pp. 173-187 (1989).
- [21] Y. Wu, S. Iyengar, R. Jain and S. Bose, "A New Generalized Computational Framework for Finding Object Orientation Using Perspective Trihedral Angle Constraint", IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 16, No. 10, pp. 469-480 (October, 1994).
- [22] J.W. Courtney, M.J. Magee and J.K. Aggarwal, "Robot Guidance Using Computer Vision", Pattern Recognition, Vol. 17, No. 6, pp. 585-592 (1984).
- [23] N. Ayache and O.D. Faugeras, "Maintaining Representations of the Environment of a Mobile Robot", IEEE Transactions on Robotics and Automation, Vol. 5, No. 6, pp. 804-819 (December, 1989)
- [24] X. Wan and G. Xu, "Camera Parameters Estimation and Evaluation in Active Vision System", Pattern Recognition, Vol. 29, No. 3, pp. 439-447 (1996).
- [25] X. Zhuang and Y. Huang, "Robuts 3D-3D Pose Estimation", IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 16, No. 8, pp. 267-275 (August, 1994).
- [26] L.H. Shapiro and R.M.Haralic, "Computer and Robot Vision", Addison-Wesley Publishing Company, Massachusetts (1993).

- [27] Arslan E. and A. Erden, "Developed of A Vision System for Generating Data Files of 2D Physical Objects for Drafting Software Packages", II. Türk Yapay Zeka ve Sinir Agları Sempozyumu, Bogaziçi University, 24-25 Haziran 1993.
- [28] T. N. Tan, G.D. Sullivan and K.D. Baker, "Closed-Form Algorithms for Object Pose and Scale Recovery in Constrained Scenes", Pattern Recognition, Vol. 29, No. 3, pp. 449-461 (1996).
- [29] P.S. Lee, Y.E. Shen and L. Wang, "Model-Based Location of Automated Guided Vehicles in the Navigation Sessions by 3D Computer Vision", Journal of Robotic Systems, Vol. 11, No. 3, pp. 181-195 (1994).
- [30] Y. Liu and T.S. Huang, "Determination of Camera Location from 2-D to 3-D Line and Point Correspondences", IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 12, No. 1, pp. 28-37 (January, 1990).
- [31] K.C. Drake, E.S. McVey and R.M. Inigo, "Experimental Position and Ranging Results for a Mobile Robot", IEEE Journals of Robotics and Automation, Vol. RA-3, No. 1, pp. 31-41 (February, 1987).
- [32] S.C. Chapra and R.P. Canale, "Numerical Methods in Engineers", McGraw-Hill, Singapore (1990).
- [33] D. Vernon, "Machine Vision", Prentice Hall, London (1991).
- [34] M.H. Han and S. Rhee, "Camera Calibration for Three-Dimensional Measurements", Pattern Recognition, Vol. 25, No. 2, pp. 155-164 (1992).
- [35] B. Klaus and P. Horn, "Robot Vision", The MIT Press, London (1986).

APPENDIX A

GENERAL PERSPECTIVE PROJECTION EQUATIONS

$$\frac{U_n}{f} = \frac{X_{nc}}{Y_{nc}} \quad \text{and} \quad \begin{bmatrix} X_{nc} \\ Y_{nc} \\ Z_{nc} \end{bmatrix} = R(\phi, \xi, \theta) \begin{bmatrix} X_n - X_o \\ Y_n - Y_o \\ Z_n - Z_o \end{bmatrix}$$
$$\frac{V_n}{f} = \frac{Z_{nc}}{Y_{nc}}$$

If swing angle ξ is zero, we can get:

$$\frac{U_n}{f} = \frac{(X_n - X_o) \cdot \cos\theta - (Y_n - Y_o) \cdot \sin\theta}{- (X_n - X_o) \cdot \sin\theta \cdot \cos\phi + (Y_n - Y_o) \cdot \cos\theta \cdot \cos\phi + (Z_n - Z_o) \cdot \sin\phi}$$

$$\frac{V_n}{f} = \frac{(X_n - X_o) \cdot \sin\theta \cdot \sin\phi - (Y_n - Y_o) \cdot \cos\theta \cdot \sin\phi + (Z_n - Z_o) \cdot \cos\phi}{- (X_n - X_o) \cdot \sin\theta \cdot \cos\phi + (Y_n - Y_o) \cdot \cos\theta \cdot \cos\phi + (Z_n - Z_o) \cdot \sin\phi}$$

This pair of equations is known as the general or fundamental perspective projections equations.

This pair of equations can be inverted so that a ratio of world coordinates can be expressed in terms of a ratio of rotated image coordinates. To do this we simply need to add this expression.

$$1 = \frac{-(X_n - X_0) \cos \phi \sin \theta + (Y_n - Y_0) \cos \theta \cos \phi + (Z_n - Z_0) \sin \phi}{-(X_n - X_0) \cos \phi \sin \theta + (Y_n - Y_0) \cos \theta \cos \phi + (Z_n - Z_0) \sin \phi}$$

In matrix form ;

$$\begin{bmatrix} \frac{U_n}{f} \\ \frac{f}{f} \\ \frac{V_n}{f} \end{bmatrix} = \frac{R \begin{bmatrix} X_n - X_0 \\ Y_n - Y_0 \\ Z_n - Z_0 \end{bmatrix}}{-(X_n - X_0) \cos \phi \sin \theta + (Y_n - Y_0) \cos \theta \cos \phi + (Z_n - Z_0) \sin \phi}$$

$$\lambda = -(X_n - X_0) \cos \phi \sin \theta + (Y_n - Y_0) \cos \theta \cos \phi + (Z_n - Z_0) \sin \phi$$

$$\frac{1}{f} R^{-1} \begin{bmatrix} U_n \\ f \\ V_n \end{bmatrix} = \frac{1}{\lambda} \begin{bmatrix} X - X_0 \\ Y - Y_0 \\ Z - Z_0 \end{bmatrix}$$

$$\frac{1}{\lambda} \begin{bmatrix} X - X_0 \\ Y - Y_0 \\ Z - Z_0 \end{bmatrix} = \frac{1}{f} \begin{bmatrix} \cos \theta U_n - \cos \phi \sin \theta f + \sin \phi \sin \theta V_n \\ \sin \theta U_n + \cos \phi \cos \theta f - \sin \phi \cos \theta V_n \\ \sin \phi f + \cos \phi V_n \end{bmatrix}$$

Now by taking the ratios of component 1 to component 2 and component 3 to component 2, the denominator λ cancels and we obtain two equations;

$$\frac{X_n - X_0}{Y_n - Y_0} = \frac{\cos\theta U_n - \cos\phi \sin\theta f + \sin\phi \sin\theta V_n}{\sin\theta U_n + \cos\phi \cos\theta f - \sin\phi \cos\theta V_n}$$

$$\frac{Z_n - Z_0}{Y_n - Y_0} = \frac{\sin\phi f + \cos\phi V_n}{\sin\theta U_n + \cos\phi \cos\theta f - \sin\phi \cos\theta V_n}$$

$(X_n Y_n Z_n)$ is the coordinate of a point P_n in the world coordinate system.

$(U_n V_n)$ is the coordinate of this point on the image plane.

APPENDIX B

LINES IN SPACES AND PLANES

A vector \mathbf{n} and a line l are parallel if \mathbf{n} is parallel to the vector \mathbf{PoP} joining any two distinct points P_0 and P on l as shown in Figure B.1.

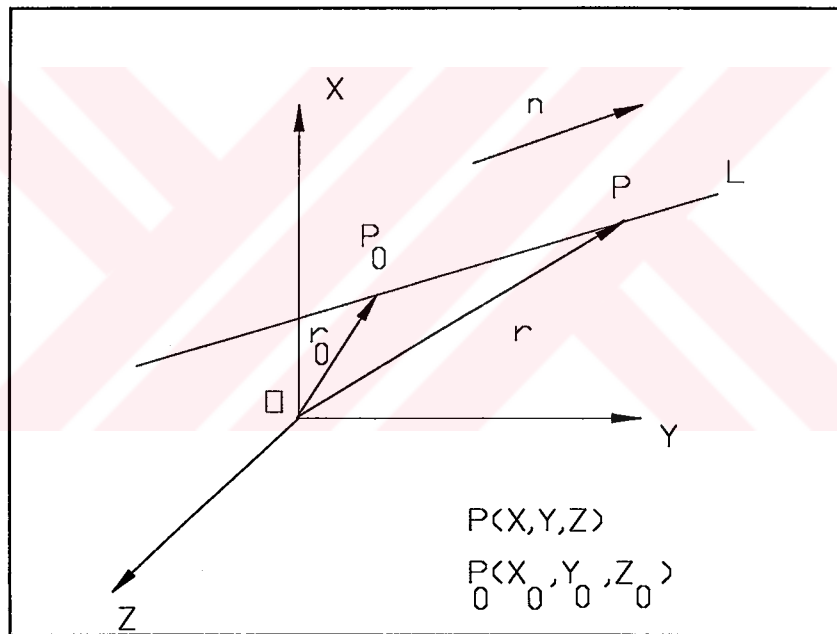


Figure B.1 Line in Space

It follows from geometry that a line in space is uniquely determined by a point P_0 on l and a vector \mathbf{n} parallel to the line. And \mathbf{PoP} can be written as follows:

$$\begin{aligned} P_0P &= t \vec{n} = \vec{r} - \vec{r}_0 \\ \vec{r} &= \vec{r}_0 + t \vec{n} \end{aligned} \quad (\text{B.1})$$

This is a vector equation of l . Since r_0 can be any vector that joins the origin to a point P_0 on l , and n is cosine vector or direction of a line. Expression B1 can be written matrix form:

$$\begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} a \\ b \\ c \end{bmatrix} t + \begin{bmatrix} x_0 \\ y_0 \\ z_0 \end{bmatrix} \quad (\text{B.2})$$

or equivalently;

$$\begin{aligned} x &= at + x_0 & y &= bt + y_0 & z &= ct + z_0 \\ t &= \frac{x - x_0}{a} = \frac{y - y_0}{b} = \frac{z - z_0}{c} \end{aligned} \quad (\text{B.3})$$

As shown Figure B2 , $N = Ai + Bj + Ck$ is a nonzero vector and perpendicular to $P_0P = (x - x_0)i + (y - y_0)j + (z - z_0)k$. If N is a perpendicular to P_0P , this equation is written:

$$\vec{N} \cdot \vec{P_0P} = A(x - x_0) + B(y - y_0) + C(z - z_0) = 0$$

and from B2; (B.4)

$$x - x_0 = at \quad y - y_0 = bt \quad z - z_0 = ct$$

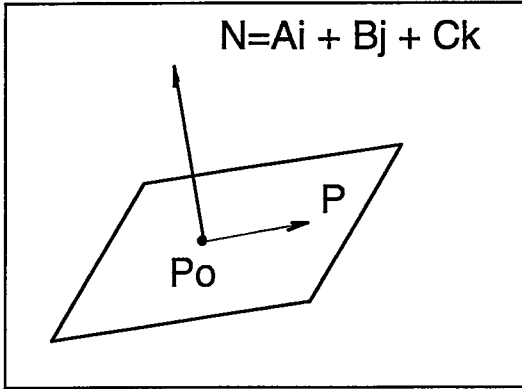


Figure B2. Line in plane and normal vector.

Expression B.3 can be written this form:

$$\begin{aligned} \vec{N} \cdot \vec{P_0P} &= Aat + Bbt + Cct = 0 \\ &= Aa + Bb + Cc = 0 \end{aligned} \quad (\text{B.4})$$

Then B.4 can be written :

$$\vec{N} \cdot \vec{n} = Aa + Bb + Cc = 0 \quad (\text{B.5})$$