A Feature Recognizing Vision System to Help Guide an Automatic Vehicle

By

Aili Zheng

100

A Thesis

Submitted to the Faculty of Graduate Studies

in Partial Fulfillment of the Requirements

for the Degree of

Master of Science

in

Industrial Engineering

ISBN 0-612-13595-0

Name _____

Dissertation Abstracts Int ational is arranged by broad, general subject categories. Please select the one subject which most nearly describes the content of your dissertation. Enter the corresponding four-digit code in the spaces provided.

industrial — SUBJECT TERM

0546 — SUBJECT CODE

U·M·I

Subject Categories

THE HUMANITIES AND SOCIAL SCIENCES

COMMUNICATIONS AND THE ARTS
Architecture 0729
Art History 0377
Cinema 0900
Dance 0378
Fine Arts 0357
Information Science 0723
Journalism 0391
Library Science 0399
Mass Communications 0708
Music 0413
Speech Communication 0459
Theater 0465

EDUCATION
General 0515
Administration 0514
Adult and Continuing 0516
Agricultural 0517
Art 0273
Bilingual and Multicultural 0282
Business 0688
Community College 0275
Curriculum and Instruction 0727
Early Childhood 0518
Elementary 0524
Finance 0277
Guidance and Counseling 0519
Health 0680
Higher 0745
History of 0520
Home Economics 0278
Industrial 0521
Language and Literature 0279
Mathematics 0280
Music 0522
Philosophy of 0998
Physical 0523
Psychology 0525
Reading 0535
Religious 0527
Sciences 0714
Secondary 0533
Social Sciences 0534
Sociology of 0340
Special 0529
Teacher Training 0530
Technology 0710
Tests and Measurements 0288
Vocational 0747

LANGUAGE, LITERATURE AND LINGUISTICS
Language
- General 0679
- Ancient 0289
- Linguistics 0290
- Modern 0291

Literature
- General 0401
- Classical 0294
- Comparative 0295
- Medieval 0297
- Modern 0298
- African 0316
- American 0591
- Asian 0305
- Canadian (English) 0352
- Canadian (French) 0355
- English 0593
- Germanic 0311
- Latin American 0312
- Middle Eastern 0315
- Romance 0313
- Slavic and East European 0314

PHILOSOPHY, RELIGION AND THEOLOGY
Philosophy 0422
Religion
- General 0318
- Biblical Studies 0321
- Clergy 0319
- History of 0320
- Philosophy of 0322

Theology 0469

SOCIAL SCIENCES
American Studies 0323
Anthropology
- Archaeology 0324
- Cultural 0326
- Physical 0327

Business Administration
- General 0310
- Accounting 0272
- Banking 0770
- Management 0454
- Marketing 0338

Canadian Studies 0385
Economics
- General 0501
- Agricultural 0503
- Commerce-Business 0505
- Finance 0508
- History 0509
- Labor 0510
- Theory 0511

Folklore 0358
Geography 0366
Gerontology 0351
History
- General 0578
- Ancient 0579
- Medieval 0581
- Modern 0582
- Black 0328
- African 0331
- Asia, Australia and Oceania 0332
- Canadian 0334
- European 0335
- Latin American 0336
- Middle Eastern 0333
- United States 0337

History of Science 0585
Law 0398
Political Science
- General 0615
- International Law and Relations 0616
- Public Administration 0617

Recreation 0814
Social Work 0452
Sociology
- General 0626
- Criminology and Penology 0627
- Demography 0938
- Ethnic and Racial Studies 0631
- Individual and Family Studies 0628
- Industrial and Labor Relations 0629
- Public and Social Welfare 0630
- Social Structure and Development 0700
- Theory and Methods 0344

Transportation 0709
Urban and Regional Planning 0999
Women's Studies 0453

THE SCIENCES AND ENGINEERING

BIOLOGICAL SCIENCES
Agriculture
- General 0473
- Agronomy 0285
- Animal Culture and Nutrition 0475
- Animal Pathology 0476
- Food Science and Technology 0359
- Forestry and Wildlife 0478
- Plant Culture 0479
- Plant Pathology 0480
- Plant Physiology 0817
- Range Management 0777
- Wood Technology 0746

Biology
- General 0306
- Anatomy 0287
- Biostatistics 0308
- Botany 0309
- Cell 0379
- Ecology 0329
- Entomology 0353
- Genetics 0369
- Limnology 0793
- Microbiology 0410
- Molecular 0307
- Neuroscience 0317
- Oceanography 0416
- Physiology 0433
- Radiation 0821
- Veterinary Science 0778
- Zoology 0472

Biophysics
- General 0786
- Medical 0760

EARTH SCIENCES
Biogeochemistry 0425
Geochemistry 0996
Geodesy 0370
Geology 0372
Geophysics 0373
Hydrology 0388
Mineralogy 0411
Paleobotany 0345
Paleoecology 0426
Paleontology 0418
Paleozoology 0985
Palynology 0427
Physical Geography 0368
Physical Oceanography 0415

HEALTH AND ENVIRONMENTAL SCIENCES
Environmental Sciences 0768
Health Sciences
- General 0566
- Audiology 0300
- Chemotherapy 0992
- Dentistry 0567
- Education 0350
- Hospital Management 0769
- Human Development 0758
- Immunology 0982
- Medicine and Surgery 0564
- Mental Health 0347
- Nursing 0569
- Nutrition 0570
- Obstetrics and Gynecology 0380
- Occupational Health and Therapy 0354
- Ophthalmology 0381
- Pathology 0571
- Pharmacology 0419
- Pharmacy 0572
- Physical Therapy 0382
- Public Health 0573
- Radiology 0574
- Recreation 0575
- Speech Pathology 0460
- Toxicology 0383

Home Economics 0386

PHYSICAL SCIENCES
Pure Sciences
Chemistry
- General 0485
- Agricultural 0749
- Analytical 0486
- Biochemistry 0487
- Inorganic 0488
- Nuclear 0738
- Organic 0490
- Pharmaceutical 0491
- Physical 0494
- Polymer 0495
- Radiation 0754

Mathematics 0405
Physics
- General 0605
- Acoustics 0986
- Astronomy and Astrophysics 0606
- Atmospheric Science 0608
- Atomic 0748
- Electronics and Electricity 0607
- Elementary Particles and High Energy 0798
- Fluid and Plasma 0759
- Molecular 0609
- Nuclear 0610
- Optics 0752
- Radiation 0756
- Solid State 0611

Statistics 0463

Applied Sciences
Applied Mechanics 0346
Computer Science 0984
Engineering
- General 0537
- Aerospace 0538
- Agricultural 0539
- Automotive 0540
- Biomedical 0541
- Chemical 0542
- Civil 0543
- Electronics and Electrical 0544
- Heat and Thermodynamics 0348
- Hydraulic 0545
- Industrial 0546
- Marine 0547
- Materials Science 0794
- Mechanical 0548
- Metallurgy 0743
- Mining 0551
- Nuclear 0552
- Packaging 0549
- Petroleum 0765
- Sanitary and Municipal 0554
- System Science 0790

Geotechnology 0428
Operations Research 0796
Plastics Technology 0795
Textile Technology 0994

PSYCHOLOGY
General 0621
Behavioral 0384
Clinical 0622
Developmental 0620
Experimental 0623
Industrial 0624
Personality 0625
Physiological 0989
Psychobiology 0349
Psychometrics 0632
Social 0451

A FEATURE RECOGNIZING VISION SYSTEM TO HELP GUIDE AN AUTOMATIC VEHICLE

BY

AILI ZHENG

A Thesis submitted to the Faculty of Graduate Studies of the University of Manitoba in partial fulfillment of the requirements of the degree of

MASTER OF SCIENCE

Summary

As manufacturing industries move to more flexible manufacturing systems to handle large production amounts in short runs, the demand for automatic systems increases dramatically. To meet this challenge, automatic vehicles, as one part of material handling systems, have to provide the linkage between production units, and must possess a higher degree of flexibility than conventional automatic guided vehicle systems. This flexibility can only be obtained by the capability of autonomous navigation.

Most existing automatic vehicle systems rely on odometers to estimate their position. Since the errors are accumulated due to wheel slippage, a vehicle may eventually lose its way. This position loss is a problem especially in situations where the vehicle must navigate to avoid obstacles that may appear on the guide-path or to interact with other manufacturing units (robots, machine tools etc.). Therefore, the position determination of an automatic vehicle in real time working space is becoming imperative in the operation of an automatic vehicle.

To increase the flexibility of an automatic guided vehicle and determine the position of an automatic vehicle in a manufacturing environment, a machine vision system is proposed. This vision system uses a single camera, passive vision technique, and a map of the related environment. The technique to determine the position of the vehicle is to have the vehicle relate to fixed marks in the room. This is accomplished by a TV camera and a recognition algorithm that has been installed in a computer.

The vision system focuses on the implementation of the target feature recognition algorithm that can uniquely identify custom-made or realistic targets' features without knowing their sizes. The recognition algorithm contains two major phases: an information extraction phase, and a recognition implementation phase. The information extraction process converts the real object brightness intensity information into the useful information

for feature identification. The recognition implementation process compares the obtained edge features with the desired target feature to be identified in order to decide if the desired target in the real scene exists.

The vision system has been tested under a variety of illuminations, using images taken from different camera viewpoints. The test results have shown that the proposed vision system is promising in term of its performance when searching for a specific target feature. The total processing time is 2.4 seconds without any improvement to speed up the system software.

In the following thesis, the background and problem are described in Chapter 1, related literature reviews are given in Chapter 2, the system overview and requirements are given in Chapter 3, the recognition procedure of the system is described in Chapters 4 and 5, the test results are presented in Chapter 6, and at last some further work about the system software is discussed before concluding the thesis.

Acknowledgments

First of all, I would like to acknowledge my advisor Dr. D. Strong for his excellent guidance, support and patience through the research and thesis writing. Words do not describe my deep gratitude.

I would also like to thank Prof. O. Hawaleshka, Dr. W. H. Lehn and Hassen Zghal for editing the thesis and providing valuable suggestions.

Finally but not at least, I would like to thank my husband, Mr. Lei Guan, for his constant encouragement, support and love.

Table of Contents

Summary i
Acknowledgments iii
Table of Contents iv
Chapter 1. Introduction 1
Chapter 2. Literature Review
on Vehicle Guidance Techniques 6
2.1 Primary Vehicle Guidance Techniques 7
2.1.1 Fixed Path Guidance 7
2.1.2 Software Programmable Path Techniques 8
2.2 Passive Imaging for Vehicle Guidance 10
2.2.1 Standard Marks 10
2.2.2 Passive Stereo Imaging 14
2.2.3 Outdoor Road Following Guidance 17
2.3 Active Imaging for Vehicle Guidance 18
2.4 Sensor-Based Guidance Navigation 20
2.5 Neural Network for Visual Processing 22
Chapter 3. A Feature Recognizing Vision System
to Help Guide an Automatic Vehicle 24
3.1 Vision System and Vehicle Control 24
3.2 Objective and Requirement of the Vision System 25
3.2.1 The Objective of the Vision System 26
3.2.2 Requirements of the Vision System 26
3.3 The Feature Recognizing Vision System 27
3.3.1 System Overview 28

3.3.2 System Elements 29
3.4 Recognition Algorithm 31
3.4.1 Gray Scale Manipulation 32
3.4.2 Edge Detection 33
3.4.3 Log Difference of Edge Intensity Measurement 36
3.4.4 Feature Identification 37
Chapter 4. Information Extraction Processing 39
4.1 Gray Scale Manipulation 39
4.2 Edge Detection 49
4.2.1 Differentiation and Thresholding 50
4.2.2 One Pixel Edge determination 52
4.3 Log Difference of Edge Intensity Measurement 54
Chapter 5. Recognition Implementation 55
5.1 Feature Identification 55
5.2 Software Implementation 62
5.2.1 Flow Chart of the Recognition Algorithm 64
5.2.2 Programming Implementation 66
Chapter 6. Experimental Results 69
6.1 Test Environment and Method 69
6.2 Image analysis result before feature identification 70
6.3 Calibration of Image Acquisition System 72
6.4 Test Results 73
6.4.1 Recognizing Gray Targets 73
6.4.2 Recognizing an Electrical Outlet 76
6.4.3 Recognizing a Switch Plate 77
6.4.4 Recognizing other alternatives 78
6.5 Discussion of the Test Results 80

6.5.1 The Processing Time 80
6.5.2 Recognition Result 81
Chapter 7. Discussion and Conclusion 84
7.1 Discussions 84
7.1.1 Hardware Limitations 84
7.1.2 Software Improvements 84
7.2 Conclusion 88

Chapter 1

Introduction

Automatic guided vehicles, or programmed carriers which transport goods from point to point, have been introduced and explored for many years in automatic traffic control and factory automation. They are a part of automatic material handling technology and have become an integral part of today's automated factory. Using an automatic guided vehicle, has the following advantages:

1) The vehicle takes over tasks that are considered to be hazardous, boring and repetitive for human beings.
2) The vehicle saves labor and provides a more efficient connection between parts stock and production centers.
3) The movement of parts can be done accurately, safely with minimum material damage.
4) The vehicle provides the capabilities to increase the system productivity and flexibility.

Most commonly used automatic guided vehicle systems are guided by a fixed path. In these systems, the vehicle follows the fixed path by sensing radio waves transmitted from the "steering cables" buried underneath the floor. The major disadvantages of these systems are its significant installation costs and the difficulty to alter and repair the path.

As manufacturing industries move to more flexible manufacturing systems which can have great production but in short runs, the demand for automatic systems increases dramatically. To meet this challenge, automatic vehicles as one part of the material handling system, must provide an effective linkage between production units, and must possess higher degree of flexibility than conventional automatic guided vehicle systems. The flexibility can only be obtained through a capability of autonomous navigation.

To increase the flexibility of an automatic guided vehicle, several alternative methods such as visual guidance, sensor guidance and neural network, have been put forward. All methods incorporate the automatic guided vehicle system with more intelligence which is either on the vehicle or in the environment where it works.

One method is to provide the vehicle with one or more sensors to sense the environment. This method uses sensors such as ultrasonic sensors, laser beam scanners and infrared sensors, sometimes combined with external beacons that are specially structured. The latter may not be able to provide the beacon's position when a laser beam senses an environment that may be interfered by workers and machines. The disadvantage of this method is the lack of flexibility and viewing extent. A triangulation method[6] using *infrared sensors*, has been developed to compute a robot's position relative to fixed beacons in a factory, but this method requires expensive sensors and high cost to build and set up.

Visual guidance techniques using machine vision are much more efficient and more flexible than fixed path guidance techniques. In that, vision systems cost less to build, maintain and set up, and perform better than conventional methods. In addition, vision systems require very limited computational resources which can be provided by a standard microprocessor.

Visual guidance techniques include passive imaging, stereo imaging and active imaging. *Passive imaging* methods image a scene naturally without on-board lighting devices, and derive useful information for final decision making. The advantages of this technique are its inexpensiveness and ease of set-up.

Stereo imaging methods calculate the range information of an object through a simple triangulation, but they have a corresponding problem in matching images. The correspondence mismatching is a common problem which leads to incorrect object identification and incorrect object disparity measurement[9-10]. The correlation of objects

is also expensive in terms of processing time. The calibration problem[18] needs to be solved in order to satisfy some physical constraints.

Active imaging techniques[7-8] do not require any prior knowledge about a robot's environment, and are fast and efficient. They do, however, need devices to project signals such as a laser beam, ultrasound or grid of light onto the viewed objects.

Neural network applications in automatic pattern recognition and vehicle guidance have been explored since the 1980's. *Neural networks* simulate the brain to learn environments, but there still exist some unsolved problems such as unpredictable characteristic of 3D target and scene clutter.

As stated before, the position and flexibility of automatic vehicles are the major concerns in guiding an automatic vehicle. Most existing automatic vehicle systems rely on odometers to estimate their position. Since errors are accumulated due to the slippage of wheels, a vehicle may eventually lose its way. This position loss is a problem especially in situations where the vehicle must navigate to avoid obstacles that may appear on the guide-path or to interact with other manufacturing units (robots, machine tools etc.). Therefore, the position determination of an automatic vehicle in real time working space is becoming imperative in the operation of an automatic vehicle.

This thesis deals with the creation of a vision system to guide an automatic vehicle. Considering that most vision methods reported for automatic vehicle navigation are for outdoor navigation, while those in manufacturing environments are less reported, the vision system to be proposed is to be used in manufacturing environments. Through our research, an efficiently practical solution to recognize guide marks in manufacturing environments, is expected to be achieved. The vision system should have the following characteristics:

a. The system must be competitive in cost, reliability, processing speed and performance.

b. It must be easily implemented and robust enough to operate under a variety of manufacturing environments.

c. It must possess a "user interface" feature which would allow a human operator to modify system parameters so that the system can be operated under a variety of environment settings.

Our vision system is based on a single camera with combined passive vision technique, and a map of the related environment. It provides information to the control system of the automatic vehicle, and uses an on-board computer containing its software for the guidance of the vehicle. The computer stores a map which states the absolute position of guide marks and the path along which the vehicle is required to travel. As the vehicle position is approximately determined by the vehicle wheels, the statement of vehicle wheel knowledge needs to be updated from time to time because of wheel slippage. The technique used in this research is to determine the wheel position relative to fixed marks in the room. This is accomplished by a TV camera and a recognition algorithm installed in the computer.

The thesis focuses on the implementation of a target feature recognition algorithm that can uniquely identify custom-made or realistic targets without knowing their sizes, and the proposing of a practical solution that has potentiality to be used in a manufacturing environment.

The recognition algorithm contains two major phases: an information extraction phase, and a recognition implementation phase. The information extraction process converts the real object brightness intensity information into useful information for feature identification. The recognition implementation process compares the obtained edge features with the desired target feature to be identified to decide if the desired target in the real scene exists.

The thesis is organized as follows: Chapter 2 reviews guidance techniques for automatic vehicles; Chapter 3 overviews the vision system including the objectives and

requirements of the vision system and system elements. Chapter 4 describes the first level of recognition procedure or information extraction. Chapter 5 gives how the recognition process and the software are implemented. Chapter 6 presents experimental results along with related discussion. Chapter 7 discusses the potential usage and further improvement of the vision system. Chapter 8 summarizes the system features and further work to be recommended.

Chapter 2

Literature Review on Vehicle Guidance Techniques

A number of techniques related to vehicle guidance for navigation have been reported. In this chapter, we shall briefly review several techniques for automatic vehicle guidance. These techniques include primary techniques, passive imaging, active imaging, sensor-based and neural network applications in visual processing. All these techniques focus on the issue of how to guide a vehicle while it is in motion. Some of them derive the information of the vehicle position, while others create a map of the vehicle's local environment. However, in practice, most of these methods have limitations including setup time, computation time, inability to handle complex scenes, route alteration difficulties, size of reference object, accuracy and range extent. The system to be described in Chapter 3 has fewer limitations compared with the approaches that have been previously proposed and reported here.

Although visual guidance techniques have not yet become commercial, research has been ongoing for several years. Most vision techniques for vehicle guidance uses passive or active imaging approaches. Whatever the technique, a vision system always includes three steps: first, acquiring an image of a scene; second, processing data in the image to find its important features; and finally, making a decision on the basis of the obtained information.

Sensor-based vehicle guidance generally use ultrasonic, sonar or infrared sensors to obtain data of robot environments for collision avoidance and navigation. Neural network application in computer vision has been explored for years, but applications in vehicle

guidance have been rarely reported. In this chapter, a brief discussion is given on issues related to our interest.

2.1 Primary Vehicle Guidance Techniques

The primary vehicle guidance technique consists of fixed path guidance i.e. of conventional techniques and software programmable path techniques using internal or external references.

2.1.1 Fixed Path Guidance

The typical examples of fixed path guidance are "wire-guided" and "strip-line" systems. The *"wire-guided"* systems use a network of buried cables, which are arranged in the form of complex closed loops where each loop carries a different frequency A.C. signal. In such networks, "steering cables" for a vehicle to follow, are associated with "communication cables". The vehicle is guided by sensing radio waves transmitted from the "steering cables". The "*inductive* or *electromagnetic*" guidance[1-3] is the most commonly used wire-guided automatic vehicle system. Two "steering coils" underside of the vehicle, sense the location of "steering cables" and produce two signals. These two signals are then amplified and mutually compared, to produce an error signal which is then used via a servo system to steer the vehicle.

Wire-guided systems are fairly reliable but have a major drawback. The wires or cables must be embedded in the floor, which causes significant installation costs and makes the path difficult to alter and repair. In addition, the system network must be kept fairly simple, as junctions are not easy to manage.

The *strip-line* guidance[4] technique is similar to buried wire guidance, except in the sensing method. The line could be on the floor or ceiling, and consists of a metal tape, reflection tape or a painted line which has been painted using visible or invisible dye mixed with glass microbeads. The vehicle traces the line using a photo-sensor which senses the reflecting light from the strip.

The advantage of the strip-line guidance over "wire-guided" systems is that, paths can be fixed quickly and are easily alterable. It does, however, have the following disadvantages. First, when the lines are painted on the floor, they must be repainted from time to time as the lines get erased through wear and tear. If the lines are reflection tape, they must be kept clean. Finally, the lines could be obscured by small bits of debris thus disabling the guidance process.

2.1.2 Software Programmable Path Techniques

Instead of using fixed paths, the use of software to program paths for a vehicle is one of the alternatives for vehicle guidance. The software programmable path techniques use references, which are either internal or external, to locate vehicles in their surroundings.

2.1.2.1 Using Internal Position Reference

These approaches include *dead-reckoning*, and *inertial navigation*. The *Dead-reckoning* method[38] uses optical shaft encoders to periodically measure the precise rotation of the vehicle's driven wheel. The vehicle can then calculate its position with respect to its surroundings if the start position is known. The *Inertial navigation* method uses a gyroscope whose axis is parallel to the direction of the vehicle motion. If the vehicle deviates from its path, the resulting acceleration which is perpendicular to the direction of motion will be detected by the gyroscope. This acceleration is then integrated twice to yield the position deviation from the desired path, which can then be corrected by means of a servo mechanism. However, when the vehicle moves at a constant velocity, the position deviation can not be detected because there will be no acceleration.

These methods, have following drawbacks:

The *Dead-reckoning* method suffers from the accumulation of errors generated by the vehicle's wheel slippage, which eventually leads the vehicle to lose its way. The

Inertial navigation method bears the expensive cost for the gyroscope and frequent maintenance.

2.1.2.2 Using External Reference

A few systems use a concept similar to that stated in previous section, to locate a vehicle, except that they use external references and on-board optical devices. They are called *corner-cube laser-scanning, bar-code mark,* and *spot-mark* systems[44]. All these systems use different on-board optical sensor systems including laser beam scanner, sonic or ultrasonic transmitter and video camera, to relate the vehicles and any one of fixed references such as retroflectors, bar-code patterns and glass balls. These references could be set on the ground, the top or ceiling of the route, and the optical sensors are set either on the top or bottom of a vehicle.

A Sonic beacon system[42] uses a sonic or ultrasonic transmitter and referencing reflectors stickered on the vehicle route, to relate a vehicle and the reference points. The ultrasonic transmitter is scanned in a rotation mode, and the reading of its rotation angle is taken according to the reflected sonic wave received from a reflector. An on-board computer processes the received data and indicates the position of the vehicle. Some systems use a laser beam scanner to help vehicles to get same information as a sonic beacon system does. As a beacon, a laser scanning system which has a precision digital encoder and a data transmitter to send its angular data to a vehicle, is located at a predetermined position in a room. The sensors whose spatial relationships are given on the surface of the vehicle, sense the rotation of scanning laser beams. With the obtained data from the sensors, the vehicle's positional data can be derived.

These methods required extensive customization of the working environment for the installation of the beacons and any changes in the route network.

Several vision systems using a single camera have been reported in the literature. They use standard landmarks or painted lines to relate a vehicle to landmarks, with

minimal *a priori* knowledge of an environment. Standard landmarks included the diamond pattern[11], the planar quadrangle[12], the calibration sphere[13], and the painted strip[15-17]. All such schemes use the prior knowledge of a landmark's feature points to extract their patterns to relate the vehicle to the landmarks.

2.2 Passive Imaging for Vehicle Guidance

This section describes in detail research on passive imaging technique related to automatic vehicle guidance. Some guidance methods based on this technique were reported in [20], and they include mark tracking, landmark following, stereo vision for navigation and inverse projection. Mark tracking is used for deriving the position data in relation to the vehicle from the designed marks, placed in the environment. Landmark following refers to the use of landmarks as position references, for example: road intersections, road curbs and road boundaries, to derive the necessary information for a land vehicle. Stereo vision technique uses depth data in the scene to derive position data in a 3D environment. Inverse projection methods try to recover the three-dimensional information or surface orientation from a two-dimensional image. See Appendix B.

In passive imaging, a scene is stationary. The passive method images a scene naturally without on-board lighting devices, and then analyzes the image to collect useful signatures which are used for final decision making. The most significant advantages of this technique are its low cost, ease of set-up and wide range of view.

2.2.1 Standard Marks

The advantages of using standard marks are that they require less computational work with less prior knowledge about the robot's environments, and that it provides accurate statements relating the vehicle to the standard mark if the prior knowledge of the standard mark is available.

Triple Point Guide Mark

Kwon et al[19] proposed a method using triple circle guide marks or their equivalent of a single pattern with three feature points, to derive the position and orientation for robots with respect to a world frame system. A robot's location information is derived from a coordinate transformation based on the knowledge of these guide marks' coordinates in a world and a robot frame systems.

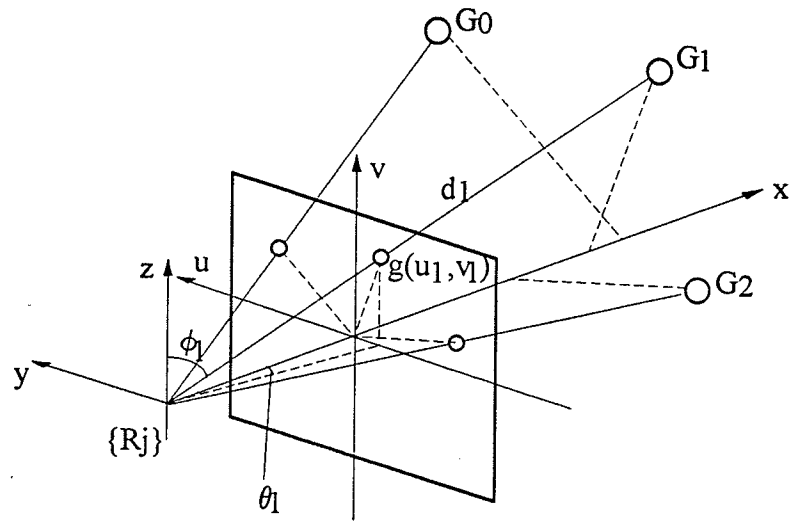


Fig. 2.1 Mapping of triple guide marks on the image plane

Fig. 2.1 illustrates a triple guide mark made of three small circles, projected on the camera image plane. For each pair of a guide mark and a robot, the mark's coordinates in the robot frame can be represented by a polar angle ϕ, a horizontal angle θ, and the distance d from the camera optical center (origin of robot frame). ϕ and θ can be calculated after image processing, but d has to be determined. By applying similar representations to three circles, nine equations with twelve unknowns are created. With prior knowledge of these circles' positions in the world frame, the robot's position is found.

This proposed system has been tested in a lab. where a reference was fastened to a post. It took 4 min for a robot to travel 1.3m using this technique.

Wang's approach

Wang[20] proposed a method using an artificial target having two vertical bars set on a negative background, to accurately locate targets in the camera field of view. The pattern's size is scaled as shown in Fig. 2.2, with a typical size of 5 x 7 inches. The pattern is set up such that its vertical bars are always perpendicular to the floor.

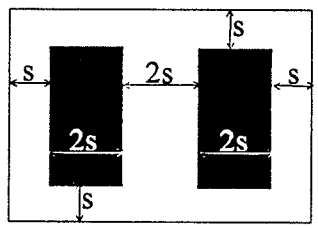


Fig. 2.2 Target proposed by Wang

By collecting the signature along a horizontal sweep of the camera sensor, which contains black, white and black consecutive segments with equal length, and combining this information with that found using other techniques, the true target can be identified. From the height and the width of the pattern appearing in the image plane, the distance between the target and a camera, and the pattern rotated angle with the respect to the image plane can be derived. The system has been tested under normal lighting condition in a laboratory and in a manufacturing environment, with one target and multiple targets in a scene. The computing time for completed processing was 250 ms, and the relavent experimental results were presented.

Kabuka & Arenas's approach

Kabuka & Arena[21] described a method which used a pattern to accurately determine the local position and orientation of a robot after it located itself near a particular operation station. The pattern comprised two portions: 1) two identification bar

codes which provide a unique code so that the viewed pattern can be easily distinguished from other patterns in the environment and also be an aid for scanning the pattern in a minimal amount of time; and 2) the displacement pattern which was used to calculate the position and orientation of the camera with respect to the entire pattern. The entire pattern's size was not mentioned.

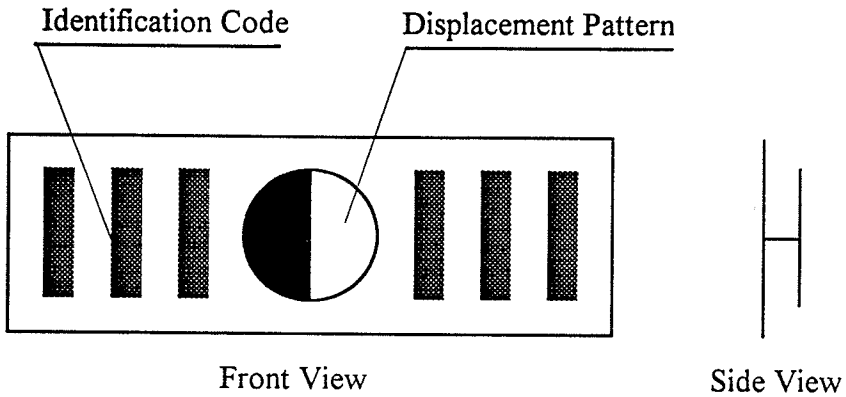


Fig. 2.3 Pattern proposed by Kabuka & Areana

Because the projection of a circle in the image plane always results in an ellipse, the parameters of the ellipse can be used to determine the robot viewing position relative to the pattern. The parameters of the ellipse in the image plane were calculated using a position invariant moment technique. The experimental results were presented and showed that the accuracy of the method was relatively poor.

This approach assumed that the viewed ellipse was centered in the image plane; i.e. the optical axis of the camera must pass through the pattern center. This assumption restricts the application of the vision system significantly.

Anemo Ceiling Landmark

The Anemo ceiling landmark is a air-diffusing device which is usually fixed on the ceiling. Fukuda et al[22] developed a simple algorithm to find a robot's position and

orientation in absolute coordinates by determining these marks' locations with respect to the robot. The Anemo shapes look like those in Fig. 2.4.

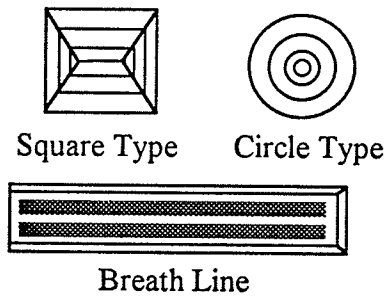


Fig. 2.4 Illustration of Anemo

In processing the image data, image segmentation was performed first and then the Anemo's feature was identified. Then the distance and angle to the landmark center from the robot were calculated based on the knowledge of the landmark location in absolute coordinates and the height of the landmark. This process completed its recognition and distance measurement in 4 seconds. The experimental results were presented on the basis of the comparison to the results of dead-reckoning[38].

2.2.2 Passive Stereo Imaging

Stereo imaging is an important technique for building a three dimensional description of a scene by triangulation of corresponding points in images observed from different viewpoints. Most of the research on passive stereo imaging has been devoted to binocular and trinocular stereo imaging.

Binocular Stereo Imaging

Binocular stereo imaging uses two cameras which are observing the same scene from two slightly different viewpoints. As soon as two image points are matched and identified as a correspondence of the same physical point, the three-dimensional coordinates of this physical point are computed. Most of stereo imaging systems are developed to build maps with 3D representation and to detect the existence of obstacles.

Tsuji et al [23] proposed a visual path planning method which used binocular stereo imaging to find free spaces for robot navigation. Their approach first matched the line segments in both images, and tested whether or not line segments were considered to be on the floor to find the free spaces for a robot. Knowing the camera height from the floor and the distance between two cameras, image points can be labeled as floor, obstacle and non-floor by calculating the vertical projections of the image points onto the floor. The robot planned its path based on the information of these image points. No experimental results were presented.

Storjohann et al[24] developed a stereo vision system, to detect the presence of possible obstacles and construct a viewer centered map of free spaces for an automatic vehicle. They presented a novel approach that used inverse perspective mappings to facilitate the matching process in binocular stereo imaging. Fig. 2.5 shows the principle of their method. The inverse perspective mapping projects the image of a scene onto a horizontal plane different from the image plane while retaining the center of projections. This is equivalent to rotate the camera sensor around its central point (Io) to give a top view of the object being viewed.

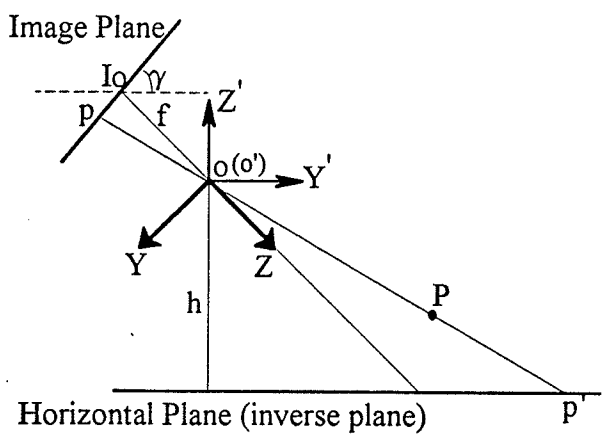


Fig. 2.5 Principle of inverse perspective mapping

Because their article was not very clear, the following description uses approximate terms to describe the matching of two images. Both images were first projected onto the horizontal plane--equivalent to the floor, and then a geometrical transformation, derived with a known camera model, was applied to one of projected images. and finally the transformed image was mapped into the other projected image. The areas that did not match each other were considered obstacles and therefore free spaces can be determined.

This method is a first functional version. It has several shortcomings. It is sensitive to small intensity differences and sometimes provides wrong position with some obstacles. The implementation of the system required 0.75s for one detection cycle.

Trinocular Stereo Imaging

Binocular stereo imaging has a major problem, that is, the correspondence of a point in both images, as mentioned above. Even with constraints to reduce the search for a unique solution, the process of eliminating false matches tends to be complex and slow, and many errors may remain. Trinocular stereo imaging is introduced for simplifying the search problem in binocular stereo imaging when one matches two images.

Ayache and Lustman[25] used three cameras to solve the correspondence problem associated with binocular stereo imaging, in order to improve the 3D reconstruction accuracy. Their method contained several stages: camera calibration, rectification of images, building 3-D segments, and image matching and validation. Camera calibration was used to determine the perspective matrix of the considered camera. Rectification of images was performed by a linear transformation of the image coordinates in projective space to have the coordinates of a point in three images satisfy geometric requirement for the correspondence of images. The process of building 3-D segments first builds 3-D lines from their 2-D images, and then computes 3-D line endpoints. The image matching and validation matches segments and conducts validation tests to get rid of wrong matches.

Zhang and Faugeras[26] described a trinocular stereo imaging system to incrementally build a world model with a mobile robot in an unknown indoor environment. Their emphasis was on the representation of the uncertainty of 3D line segments from stereo images and the integration of line segments measured at different instants. A local map of the environment was obtained by fusing segments from multiple views. By fusing a long sequence of stereo frames taken from different positions, one can obtain a global map of the environment for robot navigation. However, their method was not sufficient to describe complex scenes and the resulting world map lacked symbolic representation to identify walls, tables and doors in a scene.

2.2.3 Outdoor Road Following Guidance

Vision applications in this area refers to using the information extracted from the image of roads, to guide a land vehicle. Sometimes, this guidance technique is also coupled with other sensors, including sonar and laser sensors, to enhance the vehicle's ability to perceive its surroundings .

As a continuation of the work described in reference [39] (see Appendix B), Waxman et al[27] presented a visual navigation system for an autonomous land vehicle. The vision modules of the system operated in two processes: bootstrap and feedforward. The *bootstrap* requires analysis of entire images when the vehicle began its road-following. It uses linear feature extraction to identify linear features correlated to road boundaries, marks and shoulders in the image, which are broken into linear segments. This bootstrap process requires 90s CPU time on a VAX 11/785. The *feedforward* predicts the locations of dominant linear features near the bottom of the next image, to restrict future processing to a small portion of the visual field. Since the *feedforward* focuses its attention on a small portion of the field instead of the whole scene, it cut the computation time dramatically. This feedforward processing requires 8s on the same CPU.

Kuan et al[28] developed a vision system for road following, in which a color image was acquired by a camera and road segments were extracted through color transformation. The system contained two steps: road image segmentation and geometric reasoning. The road image segmentation applied a pixel classification algorithm based on local intensity variation, to the transformed color image to segment the image into road and non-road regions. A geometric reasoning mode compensated the inefficiency of the road image segmentation mode when interpreting a scene, and performed shape analysis, using constraints to interpret a scene into meaningful regions such as road sides, road consistence, road continuity and junction. The vehicle has been tested successfully at a speed of 8 km/hr while following a planned route and avoiding obstacles, and at a speed of 19 km/hr when following roads while interacting with a digital terrain map.

2.3 Active Imaging for Vehicle Guidance

Active imaging technique is used to extract depth information. There are a number of definitions for *active imaging*. Generally, it is referred to as active imaging when: 1) an additional light source, either structured or unstructured, is required, 2) an object being viewed provides a light source, and 3) the camera dynamically observes a scene. The first category needs devices such as a laser beam, infrared rangefinder, ultrasound or grid of light, to project the signals onto the viewed objects. The second category does not require any device to create light, because the object is a light emitter. The third category allows a camera to view a scene in different viewing positions to obtain depth information solutions similar to the stereo imaging technique.

A typical example of the first type of active imaging technique is a vision system described in [29]. In this system, Mesaki et al used a CCD camera which was associated with a light source, to find retroflex landmarks for vehicle navigation. Small clusters of landmarks were arranged on the ceiling at each spot. A light source which came from a lamp driven by a battery was mounted near to the camera on the vehicle. They assumed

that there was no obstacle on the route defined by landmarks, and that only one cluster of landmark was seen at any instant. In detecting landmarks in an image, three steps were followed: binarization of the image, labeling of bright regions and extraction of landmarks using region parameters such as area, length and position. Afterwards, the location of landmarks was calculated using given parameters. The vehicle performed the recognition under each landmark cluster and used the unique arrangement of the landmark cluster to determine the next operation such as stopping, turning left, turning right and turning around. The detection process was completed in 1s.

Lamp array landmarks can be considered as the second type of active imaging technique because the landmark creates its own light. Hashimoto et al[30] proposed a simple method using a fluorescent lamp array on the ceiling for positioning and controlling a vehicle. A video camera was used to project a fluorescent lamp on a TV monitor screen.

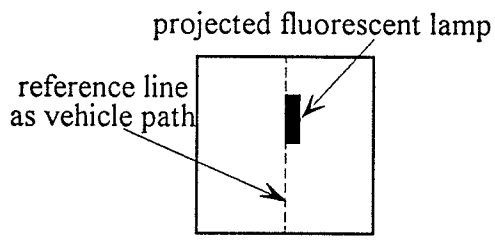


Fig. 2.6 Fluorescent lamp image on a screen

Fig. 2.6 shows a lamp image which was used to extract the position information of the vehicle. The center line of the image was used as a reference line to help the vehicle adjust its deviation with respect to the running direction. The camera image is preprocessed into binary images and the position of the fluorescent lamp is taken at its center of gravity. An analog circuit fulfilled the image processing task via photo-sensor arrays which are set up on the screen symmetrically with respect to the reference line of the screen, so that a continuous control of the vehicle can be achieved. In their experiments, three lamps were used and aligned with the running direction. The sensory

data and the landmark's position in the image were used to adjust the vehicle deviation with respect to its desired path. A computer was also used to keep the deviation of the vehicle within a few centermeters in the area where the image can not be obtained on the screen. The limitation of their proposal was that the arrangement of the lamp array in practice is random; this makes it difficult to apply the vision system.

Ishiguro et al[31] used the third type of active imaging technique to build a global map by integrating omni-directional views of an indoor environment at different locations. A single camera which can swivel about a vertical axis, was used to take stereo views. The camera focus was at a distance of R from the rotation center, and took consecutive images to form a panoramic image, from which the depth information can be derived. By measuring the angle of rotation of a feature point in the image plane from one picture to the next, the range information of the scene was attained. This range information was used to create a local map around the observing position. In order to build a global map, two omni-directional views were taken at different locations to obtain a stereo view. A global map of the environment was then obtained by integrating the local maps generated from the two panoramic views. A major limitation of the approach was the slowness in imaging.

Oh et al[32] described an active imaging technique to obtain the depth information of an object surface from stereo image pairs using a computational scheme.

2.4 Sensor-Based Guidance Navigation

Sensor based guidance navigation usually uses sensors, like ultrasonic, sonar or infrared to acquire local information of robot environments. Sometimes, a vehicle's guidance system integrates sensors for obstacle avoidance, and passive vision for obstacle localization. The most distinct significance of using sensors can be their low cost if accurate sensing data is not necessary.

Elfes[33] developed a sonar-based real-world mapping and navigation system for an autonomous mobile robot operating in an unstructured environment. An ultrasonic range

transducer was used to acquire data from the real world. The data was then interpreted and used to build a map of robot's operating environment, which contained the regions classified as empty, occupied and unknown. The system worked automatically, without any user provided-map and knowledge of the robot's surroundings, and operated successfully in indoor and outdoor environments.

Curran et al[34] described an algorithm to localize a mobile robot in a partially known environment. Their method integrated dead-reckoning, ultrasonic and infrared sensory data to estimate the position and orientation, and to further detect unexpected obstacles through a thresholding approach. A mathematical scheme were proposed to improve the position accuracy of the mobile robot by using sensory data to correct the integration error of odometry measurements. This system has been tested at a speed of 5 in/s on a hallway floor with the laboratory doors open. The total cycle time is approximately 150-200 milliseconds, including localization, motion control and communication to the robot.

Another example of sensor-based guidance system is by Cox[35]. The vehicle system was designed to operate autonomously within a structured office or factory environment. Its position estimation system compared sensory range data with a 2D map of the environment, which was used for position estimation in an absolute coordinate frame, to estimate the precise location of the vehicle. Two typical sensors were used: 1) an odometer giving the approximate position of the vehicle, and 2) an infrared rangefinder scanning the environment to obtain a range map of the environment. In order to locate the vehicle, the position estimation system first matched the obtained range information with the information on the 2D map, and then integrated the odometer data and matched positions, to obtain the precise location of the vehicle.

2.5 Neural Network for Visual Processing

Attempting to provide a human-like performance, neural network models for visual processing have been developed, loosely based on the studies of cooperative computation in biological nervous systems.

Mundkur and Desai[36] proposed a two-stage neural network approach using image and network decomposition, for applications in helping and guiding pilots of aircraft to land on runways. The system performed automatic target acquisition, identification and tracking by processing a sequence of images. Their method was proposed for solving a specific problem, i.e. locating a target in a given scene containing targets and background with some clutter. Firstly, an image is partitioned into manageable chunks, and therefore a neural network is partitioned into subnets to handle each of these chunks. Secondly, a modified network is used to prevent a subnet from giving a "valid" output to an image not in its training set, and to prevent two or more subnets from giving "valid" outputs simultaneously. The approach was simulated, and the percentage of recognition decreased with the image distortion.

Kim et al[43] used neural network concepts to propose an approach for the extraction of features from maps. The system consisted of three networks: necessary-feature, sufficient-feature and integrating. The necessary-feature network extracted road segments, and contained four processing stages: line thickness classification, line orientation extraction, parallel lines extraction and road skeleton extraction. In the sufficient-feature network, skeletons of parallel lines were directly extracted from the original image; and its outputs contained not only all of the road skeleton features but also additional false features. The integrating network combines outputs from the two feature networks, and extracts complete road information which have less noise. Experimental results showed that the performance of the system was adequate for a general map but not sufficient for a specific map area.

An automatic target recognition approach using neural net concepts was proposed in reference [37]. In this paper, Yang et al described a neural net system which consisted of two neural layers. The first layer, pattern matching, calculates the similarity between the input pattern and pattern stored in the data bank. This layer contains 256x256 input neurons and has a peak throughput of over 10 billion interconnections per second. The second layer, pattern classification, is an optical winner-take-all layer that selects the neuron with the maximum output value from the first layer to determine the category of the input pattern. The prototype of the system was under development then. The overall recognition speed of the prototype system was estimated as high as 100 targets per second.

Chapter 3

A Feature Recognizing Vision System to Help Guide an Automatic Vehicle

This chapter describes a target recognition system that enhances the guidance system of an automatic vehicle which can be used for part transfer in a manufacturing environment. The vision system is designed to determine the vehicle's location with reference to some guide marks as the vehicle navigates in its working space. Considering that manufacturing plants require automatic vehicles with a high degree of flexibility to meet the requirement for *flexible manufacturing systems*, the vision system should suit this requirement very well. The system is expected to be low in cost, fast in terms of computation time, operate efficiently under a variety of lighting conditions, and have a high degree of flexibility for modifying its parameters.

Section one describes the vision system and vehicle control. Section two states the objective and requirements of the vision system. Section three gives the overview of the vision system, and Section four describes the general principle of the recognition procedure.

More details about the principle of the recognition and the software implementation of the vision system, are described in Chapter 4 and Chapter 5 respectively.

3.1 Vision System and Vehicle Control

An automatic vehicle can be used for production automation in industrial and manufacturing environments. It plays a role for part transfer and material handling. Using an automatic vehicle can replace the manual handling of materials, substitute for a human operator in hazardous areas and provide a high degree of efficiency.

The vision system provides information to the control system of the automatic vehicle, and uses an on-board computer containing its software for the guidance of the

vehicle. The computer stores a memory map which states the path along which the vehicle is required to travel. The available information allowing the vehicle to know its location, is the position information with respect to a fixed coordinate frame, which is determined from the vehicle's wheels. This information is sufficient if a perfect rolling condition for the vehicle's wheels exists. However, the actual position and orientation of the vehicle are affected by the vehicle's wheel slippage, the rolling surface roughness and undulations. Because the vehicle accumulates its position errors while it is in motion, it is necessary to accurately update its position statement from time to time. The vision system provides this accurate information to the vehicle control system.

The technique for increasing the accuracy of the wheel position knowledge is to determine the wheel position relative to fixed marks in room from time to time. The absolute positions of these fixed marks are given by the internal map. The task to locate these marks with respect to the vehicle coordinate frame, is accomplished by a TV camera and a recognition algorithm that has been installed in the computer. Wang[20] has used this idea to propose a vision system which determined the vehicle's location with reference to a specially designed target. Our vision system was developed to find smaller targets and existing marks, and to recognize targets quickly and uniquely.

3.2 Objective and Requirement of the Vision System

As seen in section 3.1, the vision system works as an observer to locate guide marks so that the vehicle control system can determine the position and orientation of the vehicle. The vision system process consists of two major levels: target recognition and target localization. Considering the target recognition process is more important than the target localization which is done based on the degree of success the target recognition, the focus of the thesis is on the target recognition process. This recognition process must work effectively in a manufacturing environment.

3.2.1 The Objective of the Vision System

In Chapter 1, an imperative problem is posed that is to determine vehicle's position in real time while the vehicle is operating. This research aims to solve this problem, and is expected to achieve a practical solution for recognizing guide marks in a known manufacturing environment.

A majority of mobile robot guidance applications requires fast speed and moderate accuracy. The shorter the computation process is, the more frequently the vehicle can update its position information. In this context, the objective of the research is to develop a vision guidance system which can recognize a variety targets in a very short time and provide good performance with low cost. This process should not be affected by variation of illuminations and complexity of scene. This can be accomplished through a variety of techniques such as a user interface with software during system setup, and a logarithmic transformation. The target features are the combinations of different gray levels with sharp transition, and could be a few real-world targets shown from Fig. 3.2 a to c, and custom-made patterns shown from Fig. 3.2 d to f.

In determining the position and orientation of the vehicle, the vision system needs to know the orientation of two marks with respect to the vehicle, and the distance between one of the marks and the vehicle. As the marks' locations in a fixed coordinate frame are given by an internal map as mentioned before, the location of the vehicle in the same coordinate frame can be obtained. More details about the determination of the vehicle location are given in Appendix A.

3.2.2 Requirements of the Vision System

The system should work within a known typical manufacturing environment as stated earlier. The environment would typically have the following significant features:

- The floor is flat.
- The lighting conditions vary through the plant.

- The environment contains well defined, generally unobstructed travel routes, with a variety of machines and storage areas beside them.
- There exist permanent features that can be used as targets, and vertical surfaces to mount small targets on.

The vehicle moves freely on the floor of the environment, with its 2D motion approximately measured by an encoder on the rear wheels. The precise position and orientation of the vehicle still need to be updated from time to time using information obtained through the camera system.

The targets that will be used for the camera system are inexpensive because they are passive. This means that they do not create or radiate light and do not use special reflectors. These targets are stationary fixed marks, and their absolute locations of the targets are known. These target could be placed anywhere in the camera field of view.

A TV camera is used as the vision system's sensor since it costs less than any other optical sensors. The camera can be rotated using the map information to find the target of interest.

3.3 The Feature Recognizing Vision System

As described previously, the vision system to be developed uses passive targets and recognize unique targets in a very short time period, under a variety of illuminations, to help the vehicle precisely locate itself in manufacturing environments. The system should be able to work with camera-target distance range from 1m to 5m, and the processing time for recognition should be within 0.5 seconds.

This section provides a general overview of the system from the hardware employed to the recognition algorithm. First, the system overview describes the structure of the system. Then the system elements including the target, camera and image acquisition hardware are described. Finally, the vision algorithm and procedure for successfully recognizing a variety of targets are discussed.

3.3.1 System Overview

Fig.3.1 shows the configuration of the vision system. The vision system consists of a target, a TV camera for target sensing, an image acquisition and processing subsystem, and a video monitor.

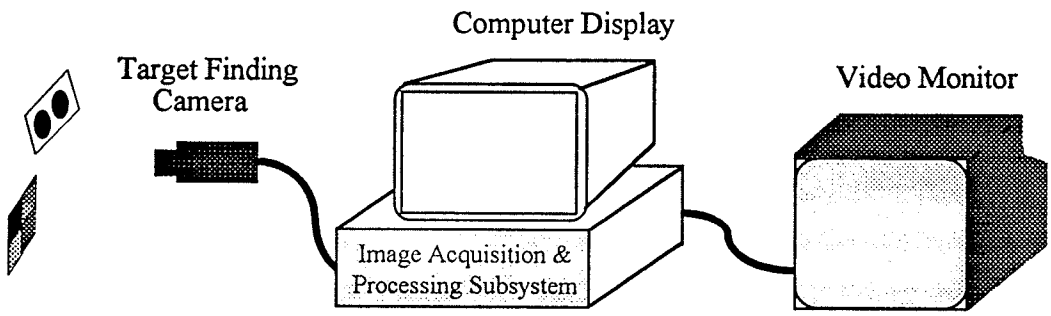


Fig. 3.1 Structure of the vision system

The targets used are custom-made gray patterns or objects in reality, which will be described in the next section. The camera is mounted on the top of the vehicle to capture pictures of scenes. It could rotate around the vertical axis if it is necessary, with the assistance of a mechanism. The image acquisition and processing subsystem contains a frame grabber for image capturing, a 80486 computer, and vision software. The frame grabber converts the picture of the scene into a 256 gray level digital image of 512x480 pixels. The computer used has a 12MB RAM and a 320 MB hard disk at a 33MHZ clock rate. It sends the command to capture an image of the scene, processes the image data and identifies the targets to provide information for the vehicle path planning. The vision software contains the code for the target recognition algorithm which is described later. The video monitor and computer display are used for displaying the captured image and the results of image analysis. They also help a human operator to set up the vision system as well supervise the vehicle operation during its navigation.

In this thesis we propose a scheme to recognize targets using the information of brightness transition ratio of adjacent gray regions. The recognition process contains four major steps:

1) gray scale manipulation,
2) edge detection,
3) logarithm difference of edge intensity measurement, and
4) feature identification.

The gray scale manipulation transforms a gray scale image into a logarithmic image which makes it easy for the user interface and possible to operate the system under different illuminations. Edge detection extracts edges of objects from the background and creates edge information in the form of structured database. The logarithmic difference of edge intensity measurement is performed after potential edges have been detected. The information obtained at this step is equivalent to a ratio of edge brightness transition, and is used for the feature identification process. The feature identification process matches the obtained information with the specifications specifying the desired target information in a data file. If the test succeeds, a target feature is claimed to be found, and the information of the target feature is stored.

3.3.2 System Elements

This section describes details of the system elements including the target and image acquisition system.

Targets

In real manufacturing environments, most articles are gray and vertically oriented. The targets used could possess similar characteristics, so that our vision system could pick up as many targets as it can. These targets must satisfy the following specifications: they must 1) be easy to set up, 2) comprise vertical or horizontal lines, 3) have smaller size in comparison with the target in [20], 4) have different gray shades of significant contrast,

and 5) have at least two different gray levels which include the background gray level, 6) have typical rectangular shape.

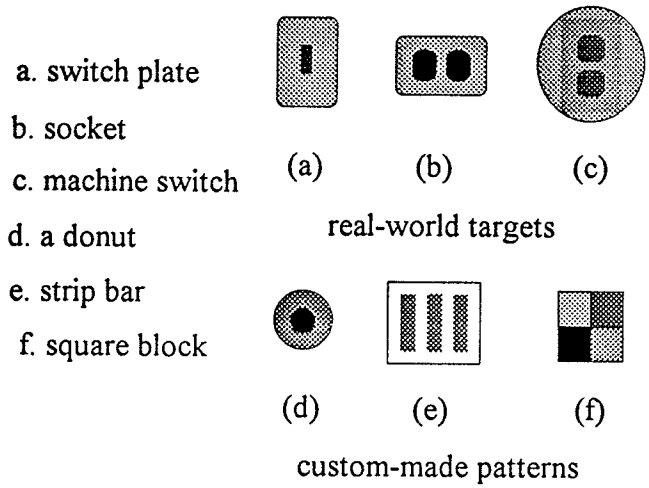


Fig. 3.2 Typical targets

The typical examples of targets are a switch plate, an electrical socket, vertical or horizontal bars, a circular machine switch as shown in Fig. 3.2. A custom-made pattern was created for testing the vision system; it has four 2 inch squares set in a 2x2 array and have four different gray levels, as shown in Fig. 3.2 f. Moreover, a target feature could contain many different gray level regions, and a circular object having adequate consistent features in the central part of the object can be also used as a target.

Image Acquisition System

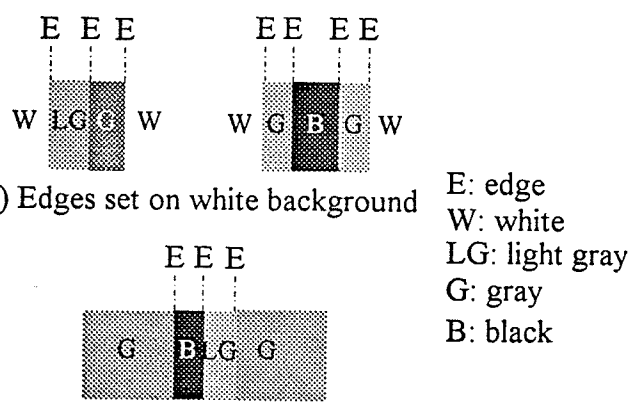
Image acquisition is implemented by a camera and a frame grabber. It performs the tasks of image sensing and digitizing. The TV camera mounted on the vehicle is a CCD camera model Pulnix TM-450, which is used to sense the scene. Its lens aperture is adjusted to the appropriate setting, which provides the most dynamic range of the target scene brightness without going into saturation. The rotation of the camera could be adjusted by a motor to suit the need for viewing around the vehicle while it remains

stationary. The camera used for the system test has a focal length of 16mm, and its viewing angles are 30.8° horizontally and 23.3° vertically.

The frame grabber (a commercial MATROX board) is used to digitize and store a picture to a computer. The images are captured through the frame grabber which is installed on a computer, and stored into a frame buffer which contains 512x480 bytes with 8 bits of gray level reading. The image grabbing speed is 1/30 second for each frame. The A/D conversion of the frame grabber is set up in a way such that it gives 255 and zero to the maximum video signal and zero video signal, respectively.

3.4 Recognition Algorithm

The recognition algorithm uses object gray level information to identify realistic objects which have similar features as shown in Fig. 3.3 within a digitized image domain. Fig. 3.3 illustrates target features representing object characteristics to be used for the recognition algorithm.



(a) Edges set on white background

(b) Edges set on gray or dark background

Fig. 3.3 Typical target features to be identified

The recognition process first extracts useful edges by scanning horizontally and compares extracted edge information with the desired target feature. It consists of four major steps as mentioned earlier and described next.

3.4.1 Gray Scale Manipulation

The gray scale manipulation is a process for providing the vision system with a high degree of flexibility under different lighting conditions and for the use of a variety of targets.

It is developed for a variety of reasons. Basically, this step is considered due to the existence of the gamma circuit in a camera, as illustrated in Fig. 4.2. The γ circuit is used to fit the video monitor visual output and increase the camera input brightness range. This γ circuit which is described in detail in section 4.1, provides an output signal to be the input object brightness to the power of γ. Since gamma varies with different light intensities arriving to the camera sensor, the output signal from the camera, which is measured by a frame grabber, needs to be measured to find the relationship between the input and output signals with different incoming light intensities.

In order to provide a high degree of flexibility for operating under a variety of lighting and using various targets which satisfy the target requirements as stated earlier, a logarithmic transformation is adopted. The logarithmic transformation converts a digital image into a logarithmic image in which the brightness gray level is a linear logarithmic function of the input object gray level. It provides the following merits:

1) The transformation allows a constant brightness increment for different brightness levels with constant brightness ratios, to represent a brightness transition ratio between adjacent areas.
2) A simple subtraction in logarithmic scale presents the same result as a division in linear scale so that the calculation time can be reduced.
3) The transformation also makes it easy for a user or an operator to read edge brightness ratios on a gray level curve when setting up target finding information for a specified mark.

To handle the γ effect and the logarithmic transformation in the system, a conversion table which provides sufficient accuracy and speed, is created and used to convert light intensities on the camera sensor to the output signals to be used.

More details of this manipulation process are described in section 4.1 giving the logarithmic transformation along with experimental results. Data of the conversion table is listed in Appendix C.

3.4.2 Edge Detection

Edge detection first detects raw edges within an input image and gives exact edge locations. It has two major steps: 1) the differentiation and thresholding step, and 2) the one pixel edge determination.

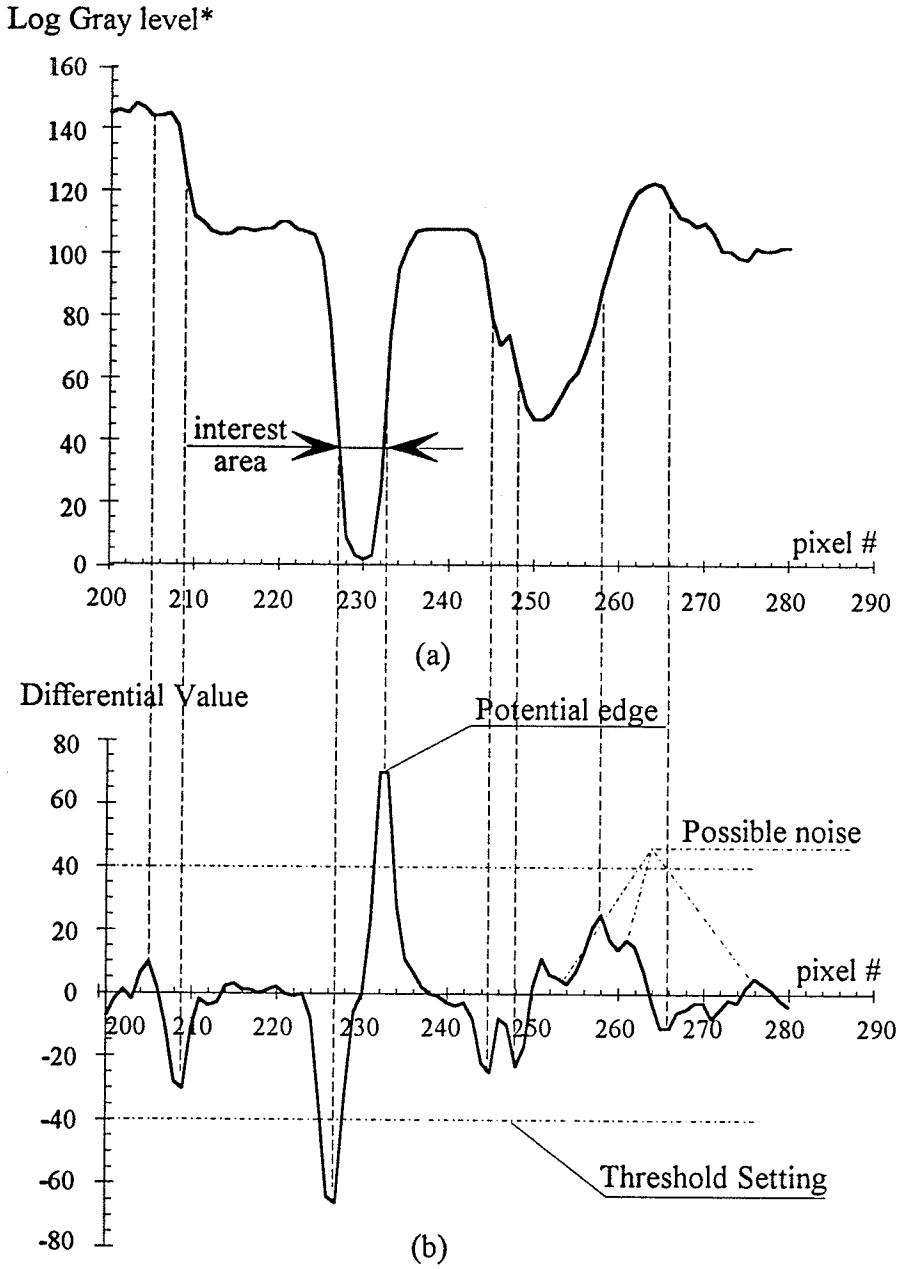


Fig. 3.4 (a) gray level profile and
(b) corresponding differential profile

*Note: the scale is timed by an multiplier(148) and added an offset(-101.4).

For the first step, the raw edges can be detected by applying a first differential operator to the input image. Fig. 3.4 shows a gray level and a differential profile along a horizontal scanned line of an image. Fig. 3.4 (b), it can be observed that a spike indicating

a change of slope in the gray level graph, could indicate the existence of a possible edge, or noise which are some dirt or whatever being caused by the system for physical reasons.

Since a true edge usually has much larger spike than noise, a thresholding operation is applied to each differential value in order to enhance potential edges and eliminate noise. Fig. 3.5 shows a discrete profile at the area of interest shown in Fig. 3.4 (a). A threshold at a value of 40 is applied to the discrete differential value at each pixel. The pixels whose differential values exceed the threshold value, indicate the presence of possible edges. However, one or more pixels remain at each potential edge point.

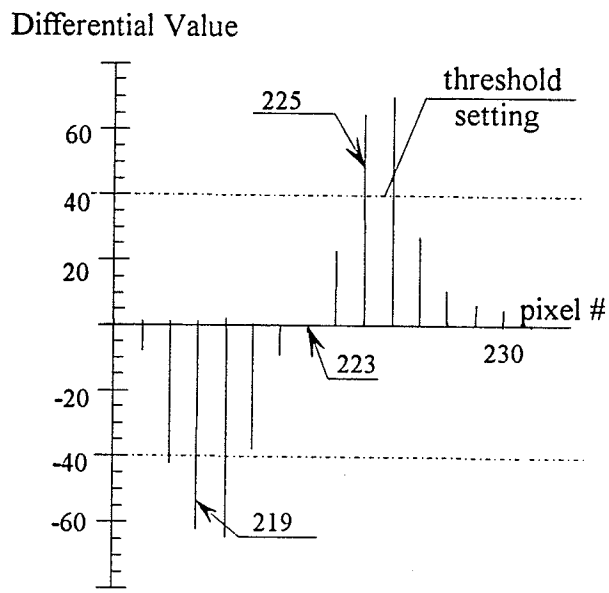


Fig. 3.5 Thresholding Differential Value

The second step determines the most probable location of an edge from the detected edge pixels in the first step. The quantity of pixels detected at each edge point, may be an odd or even number. The pixel at the center of an odd quantity of pixels or left of center of an even quantity of pixels, is selected as the edge location. For example, if two pixels have been selected in the first step as the representation of a potential edge, in the second step, the location of the first pixel will be chosen to be the edge position, as seen in Fig.

3.5, Pixel No. 225. If three pixels representing a possible edge, the location of the middle one will be chosen to be the edge position, as seen in Fig. 3.5, Pixel No. 219.

Once a potential edge has been found, the coordinates of the edge in the image domain and its brightness transition direction which indicates an edge transiting from light area to dark area or vice versa, are stored in a structured array.

3.4.3 Log Difference of Edge Intensity Measurement

After edge detection, one pixel information for each edge has been obtained. Further unique information at each found edge, is required in order to provide sufficient edge information for the process of feature identification. The process of logarithm difference of edge intensity measurement provides such information based on the success of edge detection. It measures the *brightness transition ratio* at a found edge point, i.e. the difference of the edge brightness transition on a logarithmic scale.

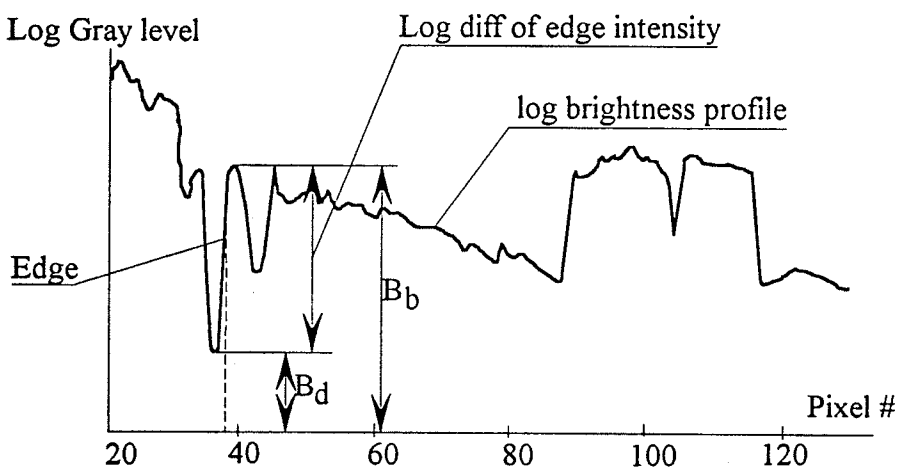


Fig. 3.6 Illustration of Log difference of edge intensity

Fig. 3.6 illustrates the log. difference of edge intensity on a log gray level profile of a scanned line, with an edge indicated. The logarithm difference of edge intensity at the edge point, is the brightness variation on the bright side and the dark side, i.e. the difference of B_b and B_d in Fig. 3.6.

From the gray level profile as shown in Fig. 3.6, it can be observed that the brightness intensity change around the edge of interest is a large variation rather than a sharp variation. This makes the brightness on the bright and dark sides of the edge, B_b and B_d, difficult to determine. Thus, both are required to be set arbitrarily. B_b is set as the average of the 3rd and 4th pixels' brightness on the bright side of the edge, and B_d is set in a similar way.

For the convenience of the later process, the logarithmic brightness intensities B_b and B_d, which are denoted as the highest and lowest edge intensities, are stored with the previous edge information.

3.4.4 Feature Identification

The final step, feature identification or recognition, refers to matching the processed target feature with the target feature described in a data file. The processed target feature is collected based on a cluster of edge information from the previous steps, which includes edge position, edge brightness transition direction, the highest and lowest edge intensities of the edge. The feature identification step relates this edge information to the stored target feature information. The data in the data file is defined on the concept of a primary edge, with a minimum and a maximum allowance for individual edge information.

A particular target feature to be recognized is shown in Fig. 3.7, with the illustration of all edge information obtained from previous steps. The process checks if each edge of a potential feature can be found so that a target feature is or not claimed to be found. Following the searching order and direction specified in the data file, the process tests each edge being processed against the criteria defined in the data file.

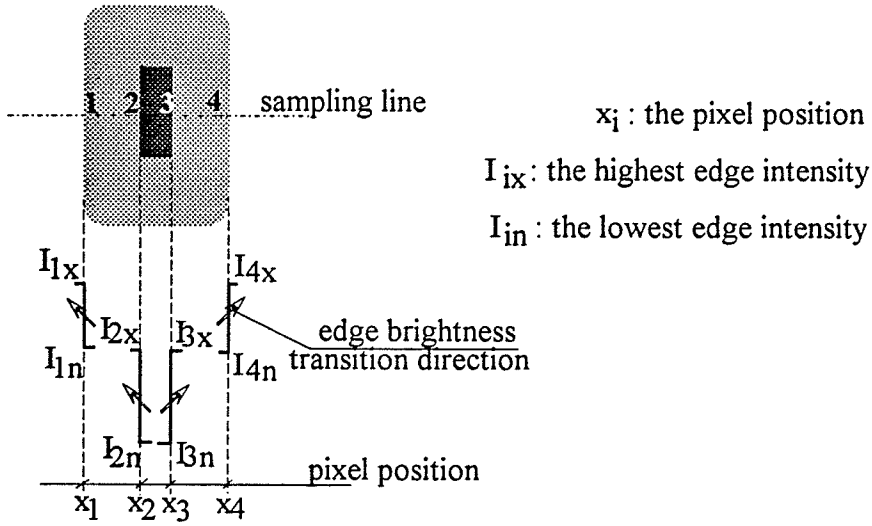


Fig. 3.7 A target feature to be recognized

Firstly, the feature identification process has to figure out which edge could be the primary edge by comparing each edge information with the primary edge information specified in the data file. Once the primary edge has been successfully found, the searching for the second edge resumes and checks if the edge being tested satisfies the requirement of the relationship between the primary edge and second edge, which are defined in the data file. The searching for the third or fourth edges follows after finding the primary and the second edges, and uses similar concepts.

Once the potential target feature has been found, the search repeats the same search strategy vertically using another identifier to search the vertical target feature in a local area where the horizontal target feature has been found. Should the search succeed, we consider the target to have been uniquely recognized.

This completes the description of the recognition process. More details on the recognition procedure are given in Chapter 4 and Chapter 5 respectively. Chapter 4, information extraction, describes the first three steps, and Chapter 5, recognition implementation, describes the fourth step and the software implementation.

Chapter 4
Information Extraction Processing

Chapter 3 has given the overview of the whole system including system requirement and target features. This chapter describes the information extraction processing in more detail, which includes the first three blocks shown in Fig. 4.1. Chapter 5 describes the details about the recognition implementation.

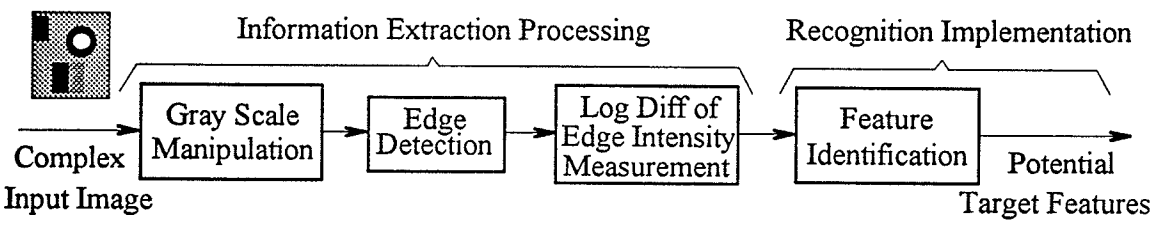


Fig. 4.1 The Recognition Procedure

The information extraction processing converts the real object brightness intensity information into the useful feature information for feature identification. It consists of three major phases (see Fig. 4.1). The first phase is *gray scale manipulation*, in which the input image gray levels are scaled using a conversion table. The second phase is *edge detection*, which extracts the edges of potential targets. The third phase is *log difference of edge intensity measurement*, which measures the edge brightness transition ratio for edges found in the previous step, and produces feature vectors.

4.1 Gray Scale Manipulation

The gray scale manipulation is developed for several reasons. The initial purposes of the manipulation are to correct the camera's gamma coefficient which affects the picture display on the CRT tube of the video monitor, and reduce the effect of variation in ambient lighting. It converts the image stored in the frame buffer to a logarithmic image by

using a conversion table. For ease of understanding, the TV camera structure and video image display are described along with a discussion of the gray scale manipulation.

Fig. 4.2 illustrates the structure of a camera which consists of a sensor, a sensor amplifier, a gamma circuit (γ) and a γ signal amplifier. The sensor produces an electrical signal proportional to the intensity of light falling upon it. This signal is then amplified by the sensor amplifier. The amplified signal is modified by a gamma circuit and is amplified again by the γ signal amplifier to produce a signal suitable for a video monitor.

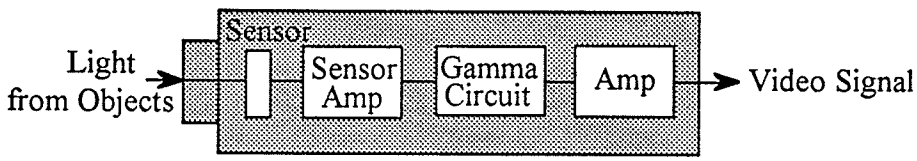


Fig. 4.2 Structure of a Camera

The reasons for having a γ circuit are as follows:

1) It compensates for the CRT non-linear characteristic response of the voltage to brightness intensity, which is shown in Fig. 4.3. The output brightness intensity of a CRT is related to the input voltage to the power of $1/\gamma$. To have a linear characteristic of output brightness intensity against object brightness, a signal from a camera sensor must undergo gamma correction by a γ circuit which is approximately the square root of the object brightness intensity.

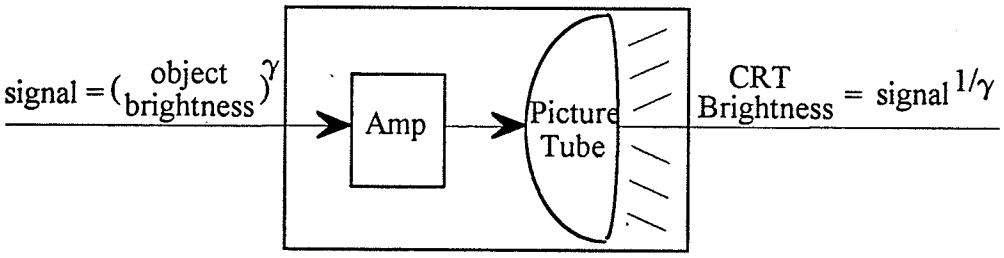


Fig. 4.3 Diagram of a video CRT display

2) The gamma circuit allows a larger brightness dynamic range when compared to linear output from the camera.

The output signal from a gamma circuit takes the form of:

$$\text{Signal} = K^*(\text{Brightness})^{\gamma} \quad (0<\gamma<=1,\ \gamma \text{ typically } \cong 0.5), \quad (4\text{-}1)$$

If γ=1, the circuit does not affect the output signal. If γ=0.5, the output is proportional to the square root of the object brightness.

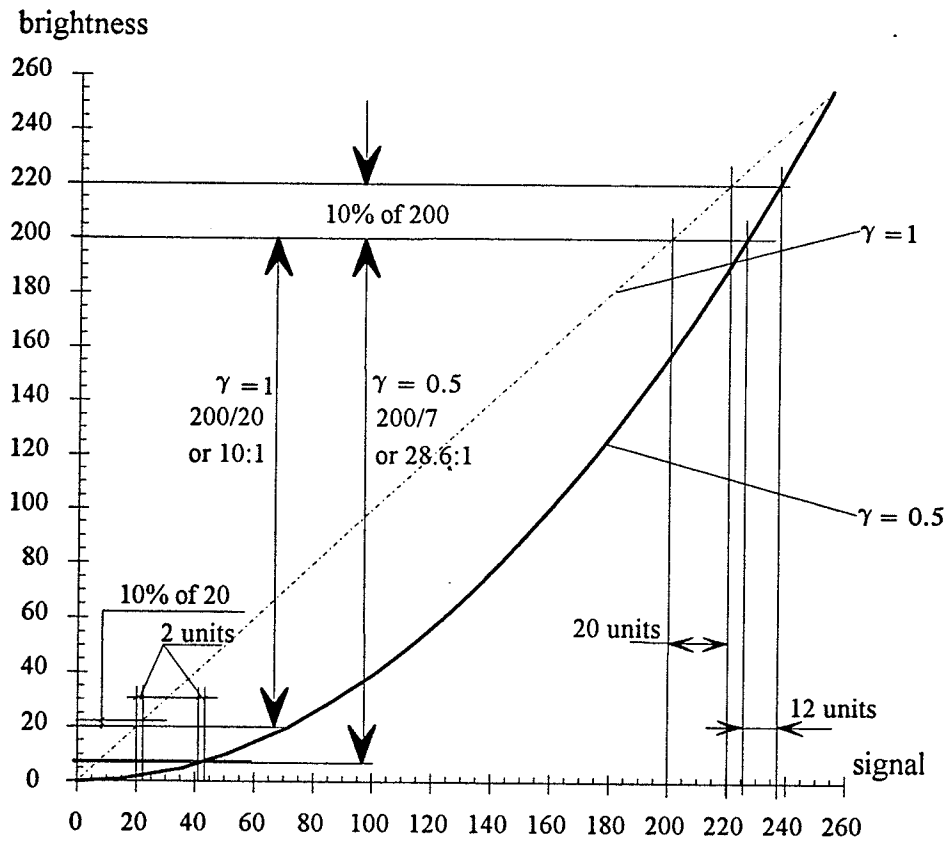


Fig. 4.4 Curves of object brightness vs. camera output signal for γ=1 and γ=0.5

Fig. 4.4 shows curves of object brightness vs. output signal for γ=1 and γ=0.5 respectively. The input brightness level is given a range of 0 to 255, and the minimum and maximum related signals for γ=1 and γ=0.5 are set at 0 and 255. The frame grabber has 255 steps, changing 1 step for each unit change in the signal. At a high brightness of 200 units of gray level, a 10% brightness

increase gives a 20 step change in the frame grabber for $\gamma=1$, and gives a 12 step change for $\gamma=0.5$. At a lower brightness of 20 units of gray level for the same percentage brightness increase, a 2 step change will be obtained for $\gamma=1$. If we set this 2 step change as the minimum resolution for image processing, a square root circuit can handle a minimum brightness which is far below 20 units of gray level. From Fig. 4.4, this brightness is about 7 units of gray level. This indicates that for a specified minimum resolution, if the ratio of brightness change with a linear response is 10:1, the ratio with a square root circuit can be 28.6:1. This shows that because of the limitation of the frame grabber, the square root circuit provides a larger dynamic brightness range than a linear response.

In practice, the square root circuit shown in Fig. 4.4 is an ideal case. It has been found that the camera gamma changes with different brightness intensities; i.e. the output signal of the gamma circuit changes with a smaller input signal being given a larger gamma. Therefore, the relationship between the output signal and the input signal must be measured. To do this, an experiment is conducted by adjusting the incoming light intensity to the camera sensor over its full dynamic range and measuring the output signal through the frame grabber.

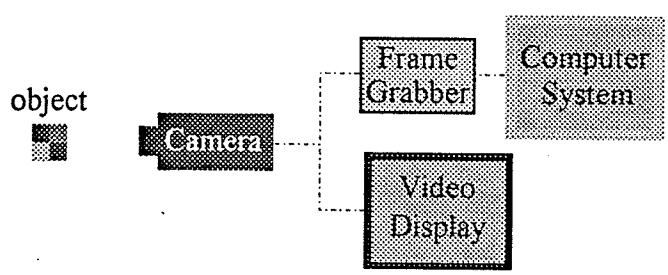


Fig. 4.5 The layout of experiment

Fig. 4.5 shows how the experiment is set up. The input signal entering the camera comes from an object and the output signal from the camera is digitized by a frame

grabber. The measured output signal is taken at a designated constant brightness area without changing the camera's location.

In the succeeding description, the brightness reading refers to the digitized output signal obtained from the frame grabber, and the incoming light intensity which is equivalent to the object brightness mentioned before, refers to the light arriving at the camera sensor.

The incoming light intensity can be changed by adjusting the lens opening of the camera. The intensity of the light entering the camera sensor is proportional to the size of the camera lens opening, which is proportional to the inverse of the square of camera aperture index. For example, an aperture index of 2.8 allows twice the amount of incoming light intensity as an aperture index of 4. If the maximum brightness reading gives the sensor full brightness for the maximum size of lens opening, i.e. the intensity of light on the object is set to give a reading of 255 from the frame grabber for an aperture index of 1.4; for a specified aperture index, the incoming light intensity on the camera sensor (O_i) is represented as:

$$O_i = \frac{510}{(\text{index})^2} \qquad (4\text{-}2)$$

where index is the camera aperture index.

To obtain a reasonably complete set of readings, a reading is taken every half aperture index mark. Based on the equation (4-2) and the experiment, the light intensity on the camera sensor and the measured output signal from the frame grabber at the corresponding aperture index, forms the second and third columns of Table 4.1.

Aperture Index*	Object Brightness (O_i)	Brightness Reading (R_i)	Log Brightness (C_i)
1.4	255	255	255
1.7	180	237	232
2	127	208	210
2.4	90	176	188
2.8	64	148	166
3.4	45	121	143
4	32	95	121
4.7	22	70	99
5.6	16	47	76
6.4	11	29	54
8	8	16	32
9.5	6	9	10

Table 4.1 Light Intensity Experimental Data
and Logarithm of Brightness

Now, in order to provide a high degree of flexibility under different lighting and for the use of a variety of targets, a transformation is considered to modify the brightness reading from the frame grabber for the following reasons:

1) The edge brightness transition ratio which is a measure of the contrast between adjacent areas, is a necessary parameter for our recognition purpose.
2) The illumination varies throughout manufacturing environments.
3) A human operator needs to read brightness ratios easily when setting up system parameters for recognizing target features.

A logarithmic transform is adopted to meet these challenges. This transformation gives the following advantages:

1) Performing a simple subtraction in logarithmic scale achieves the same result as performing a division in linear scale.

---

* the data of this column is approximately set as the camera aperture is adjusted smoothly, therefore, the data in 3rd column may slightly vary.

2) The transformation allows a particular brightness increment to represent a brightness ratio regardless of variation of illumination.

3) The gray level display of the transformation allows an operator sets up target finding information more easily because of 2) above.

Fig.4.6 (b) shows two profiles in logarithmic scale of the linear profiles in Fig.4.6 (a), and illustrates the brightness increments (R_1 and R_2) in the area of interest are the same for different brightness levels with constant brightness ratios. Fig. 4.6 (a) shows two profiles in linear scale at two different average gray levels. The solid line represents a profile at the average brightness level of 200 and the dash-dot line represents a profile at the average brightness level of 60 which is 30% of the brightness of the solid line. The brightness increments (B_1 and B_2) for both profiles at the shown area of interest, are observed to be different, even though both brightness ratios are the same. The following description shows how this transformation is accomplished.

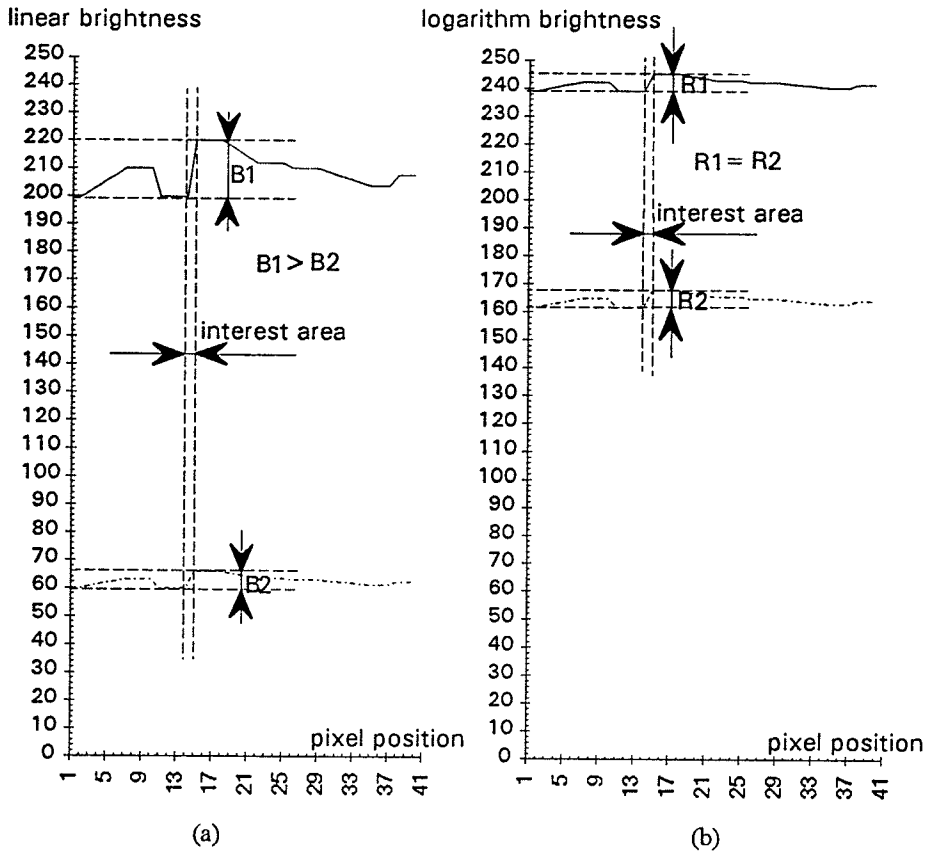


Fig. 4.6 Illustration of logarithmic process with different lighting conditions

The converted output brightness (C_i) is obtained by

$$C_i = A*Log(O_i) + B \qquad (4\text{-}3)$$

where O_i is an object brightness, and A and B are constants.

In equation (4-3), if the maximum and minimum converted brightness are known, the constants A and B can be set. In order to give a maximum usable brightness range for image analysis, the maximum converted brightness is set to be 255, which is the maximum brightness reading, and the minimum converted brightness is arbitarily set to 10, which considers the resolution limit described earlier. Substituting these two values and their corresponding object brightness into equation (4-3), A and B can be calculated as 148 and -101.4. In Table 4.1, the fourth column showing logarithmic brightness, is calculated by inserting the data from the second column with the determined A and B into equation (4-3).

From Table 4.1, one can see that the object brightness and brightness reading are not easy to represented by a simple mathematical relationship. For the simplicity of image transformation, a conversion table, which uses piecewise linear interpolation between two rows of Table 4.1, is created to convert the image stored in the frame buffer into a logarithm image. The data of logarithm brightness from 10 to 255 in the table is calculated by equation (4-3). For those brightness reading below 10, they are arbitrarily left unchanged. In practice, to ensure the vision system works properly, an image which have a brightness area below 10 units of gray level after the conversion, is ignored for further processing. Appendix C provides the entire data of the table.

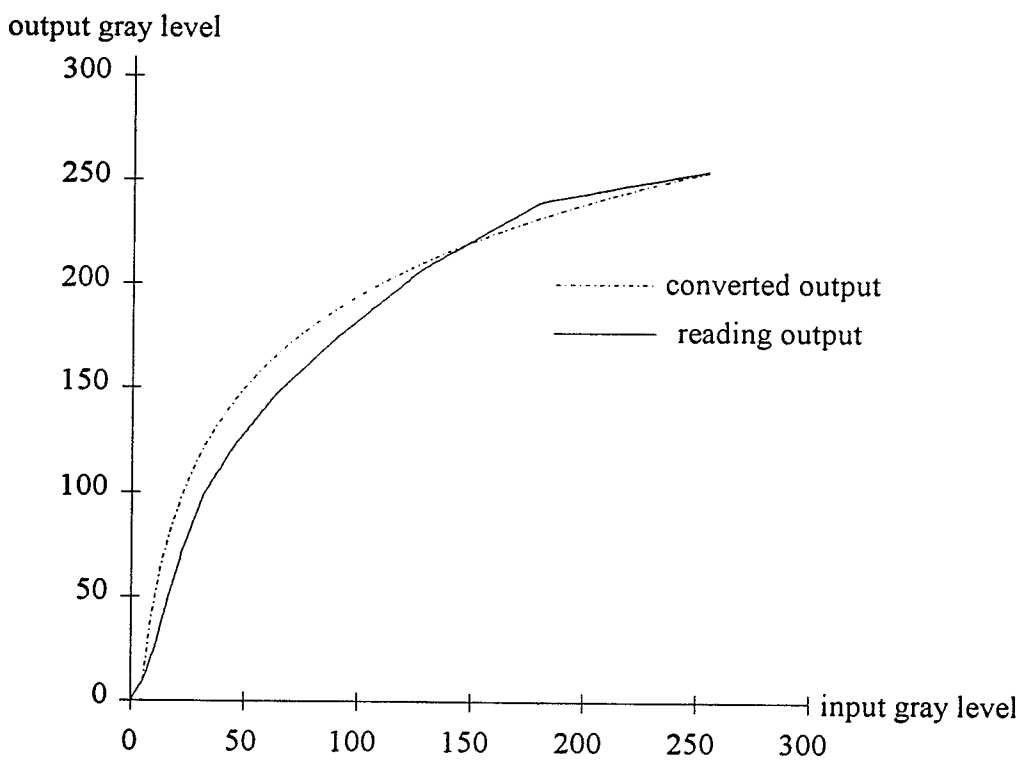


Fig. 4.7 Curves of output brightness vs. input brightness

Using the data in Appendix C, one can form two curves of output brightness vs. input brightness as shown in Fig. 4.7. When taking into account the aperture setting's

inaccuracy, the curve of reading output may need to be smoothed. If this is the case, the curve of converted output may be slightly closer to the reading output curve.

Note that the data in the second and third columns of Table 4.1, may vary slightly if images are grabbed by different frame grabbers or cameras. Therefore, an experiment relating the input of the camera sensor and the output through the frame grabber, is required, as will be discussed in section 6.3. The new conversion table to be used is generated by using the described transformation.

4.2 Edge Detection

In order to extract target features, the edges of targets need to be extracted first. The purpose of edge detection is to provide primary information for image analysis. The process of edge detection comprises two steps: 1) differentiation and thresholding; and 2) one pixel edge determination. The result of the process includes the location of the edges in the image domain and the edge brightness transition direction.

4.2.1 Differentiation and Thresholding

In theory, an edge point can be detected by using first differential operation. To detect a vertical edge, a horizontal sweep is digitized and discretized by the differential operation as:

$$G(x) = I(x+1) - I(x-1) \qquad (4\text{-}4)$$

where x is a pixel location along the row, G(x) is the differential value at the location of x, I(x+1) and I(x-1) are the brightness intensities corresponding to the locations of x+1 and x-1. An edge occurs where the value of G(x) reaches its local extreme value. In the case of a horizontal edge, a vertical sweep is digitized and similar operation is applied.

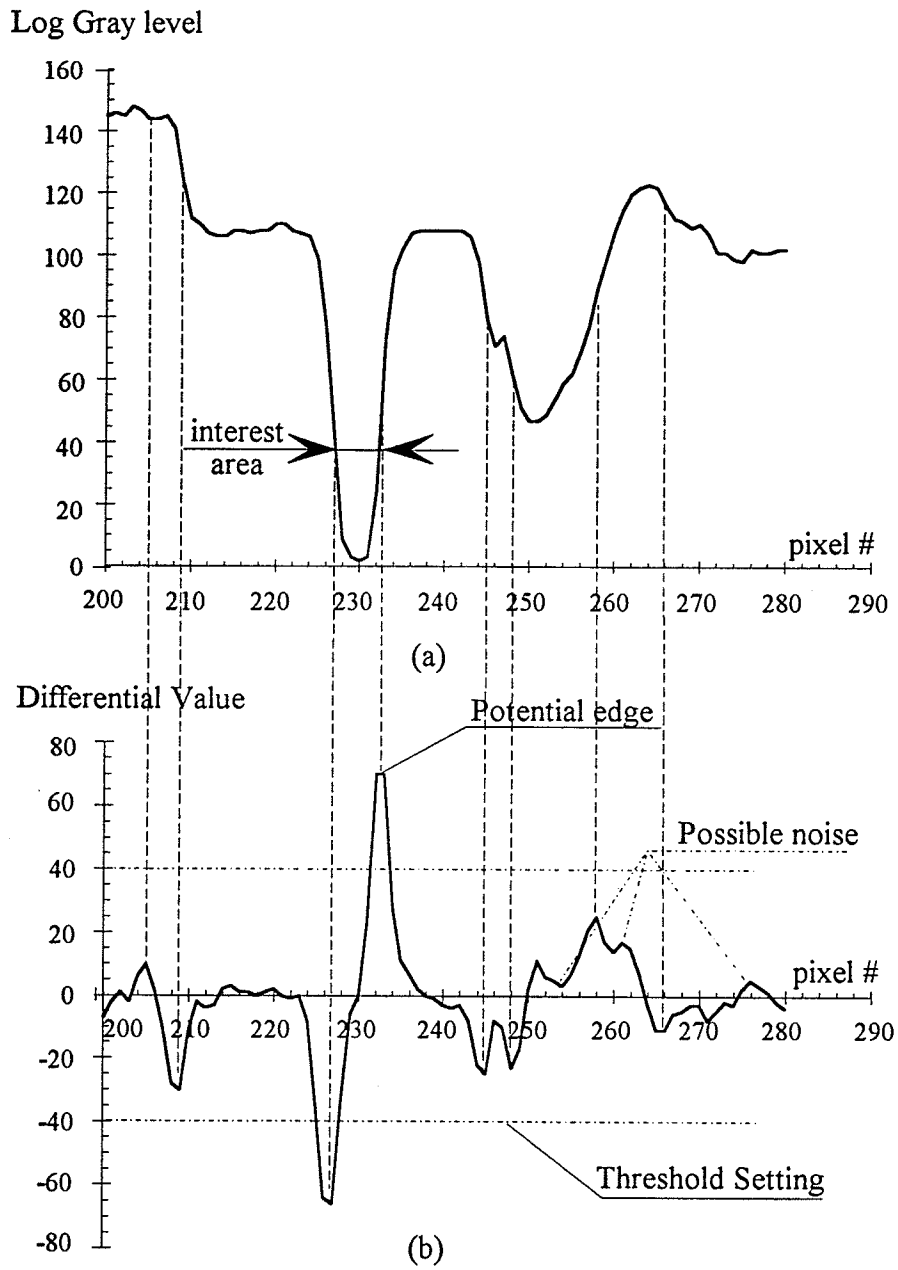


Fig. 4.8 (a) gray level profile and
(b) corresponding differential profile

Fig. 4.8 shows a gray level profile and a differential profile along a horizontal scan line of the image. From these, one can observe that a change of differential direction could indicate the existence of an edge but could also indicate noise caused by a mottled surface or system. This effect can be seen from the low peaks and other signal shapes appearing in

Fig. 4.8 (b), further indicating that the differential operator is sensitive to noise. This also shows that an edge location can not be precisely determined by simply using a differentiation operation.

To remove possible noise indicated as lower peaks, a threshold value, given as 40 units in Fig. 4.8 (b), is applied to each digitized differential value, so that they will not be considered as potential edges. The threshold value is set by scanning a line of image and observing the differential signals on the background of intrest. The threshold value is set higher than noise signals and lower than signals which have greater brightness ratios, so that the vision system can handle an input image without losing too much information.

After the process of differentiation and thresholding, one or more pixels have been detected at each potential edge, and the information of pixels being considered as possible edges is stored in a vector structure. This information includes: 1) the pixel's position referring to the coordinates of a pixel in the image domain, and 2) the pixel's brightness transition direction which is indicated the pixel's brightness changing from dark to bright area or vice verse.

4.2.2 One Pixel Edge determination

At the end of the edge detection process, one or more pixels exist at a potential edge point. A single pixel statement for the representation of a potential edge is necessary for later processing steps. The one pixel edge determination finds the most probable location of an edge from the detected edge pixels in the previous step, so that the multi-pixel situations can be clarified.

The quantity of pixels detected at each edge point may be an odd or even number. The pixel at the center of an odd quantity of pixels, or left of center of an even quantity of pixels, is selected as the edge located. For example, if two pixels have been selected as the representation of a potential edge, the location of the first pixel will be chosen as the edge location; if three pixels have been detected, the middle one, i.e. the second pixel, will be

chosen as the edge. For software implementation of the process, the center pixel of a group is determined by equation 5-5 which presents the above stated method.

$$x = x_i + Q(\frac{x_t - x_i}{2}) \qquad (4\text{-}5)$$

x is the selected pixel position, x_i and x_t are the first and the last pixel locations, where their differential magnitudes exceed the given threshold value, while $Q(\frac{x_t - x_i}{2})$ is the quotient of $\frac{x_t - x_i}{2}$.

The reasons for adding this process are several. They are mainly related to the edge brightness ratios and edge sharpness. Some of them can be given from the evidence of Fig. 4.9. This figure shows ambiguous cases in which a one pixel statement could not be obtained at each potential edge unless one arbitrarily specifies a pixel as an edge.

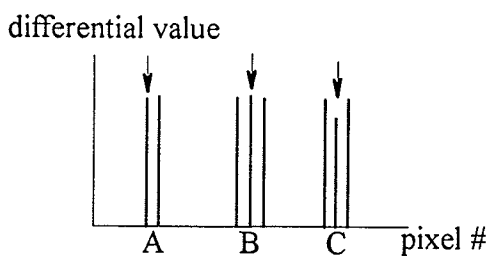


Fig. 4.9 Multiple pixel at an edge point

In Fig. 4.9, several areas of interest have been detected as potential edges after thresholding. Class A has two pixels having similar differential values standing for the presence of a potential edge. The occurrence is caused by two facts, one is because the edge is detected based on the central differential operation, the other reason is by the reflection from a rounded or non-sharp edge. Class B has three pixels having similar or equal differential values in a row, which is caused by a more rounded edge. Class C has two pixels having similar differential heights, which are separated by a pixel having lower differential height. This occurrence is due to the reflection of a non-sharp edge. An

exception to the cases shown in Fig. 4.9, is a rarely observed case in which there is a significantly large spike at one end of the pixel group. Because this occurs seldom and could be caused by an edge reflection, the pixel is still best selected by using the above process. Using the one pixel per edge process, the edges' locations for the cases illustrated in Fig. 4.9 are determined at the locations as pointed out by arrows.

Once an edge has been found, its location and brightness transition direction are stored in a vector structure for further processing.

4.3 Log Difference of Edge Intensity Measurement

From the edge detection process, edges have been extracted. The information for each edge is still not sufficient for the feature recognition process. Log difference of edge intensity measurement provides the unique information representing edges' characteristics. The measurement is equivalent to a measure of edge brightness transition ratio at an extracted edge. Therefore, a simple subtraction in logarithmic scale achieves the same results as a division in linear gray scale.

The logarithm difference of edge intensity (ΔB_x) is the logarithmic brightness variation on the bright side and dark side of the detected edge located at x. Fig. 4.10 illustrates the log difference of edge intensity on a log gray level profile of a scanned line, with an edge indicated. The log difference of edge intensity (ΔB_x) can be represented by the following:

$$\Delta B_x = B_b - B_d \qquad (4\text{-}6)$$

where B_b is the brightness on the bright side of the edge and B_d is the brightness on the dark side of the edge.

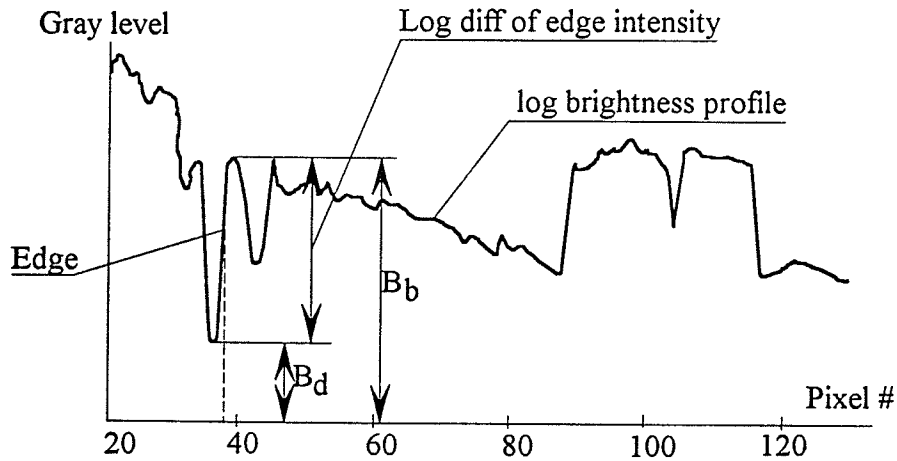


Fig. 4.10 Illustration of Log Diff of edge intensity

As shown the profile of the log brightness in Fig. 4.10, the brightness intensities at the edge transition area vary as a ramp, rather than sharply. This makes the bright side and the dark side brightness B_b and B_d next to the edge not easily determined.

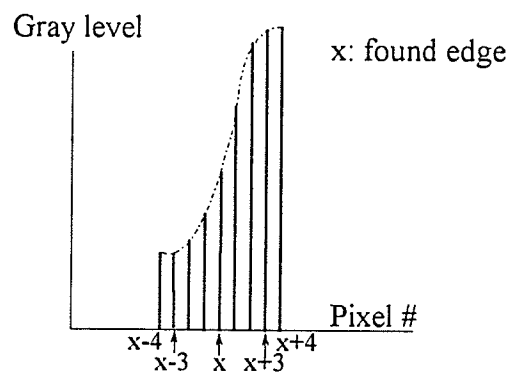


Fig. 4.11 Determination of B_b and B_d

In order to determine B_b and B_d, the profile around the found edge as shown in Fig. 4.10 is clipped horizontally and digitized as shown in Fig. 4.11. Through observations of image analysis experiments, B_b is arbitrarily set as the average of the third and fourth pixel's brightness on the bright side of the found edge and B_d is set in a similar way. B_b and B_d at x can be represented by equation (4-7).

$$B_r = \frac{B_{x+3} + B_{x+4}}{2}$$

$$B_l = \frac{B_{x-3} + B_{x-4}}{2} \qquad (4\text{-}7)$$

where B_{x+3}, B_{x+4}, B_{x-3} and B_{x-4} correspond to the brightness respectively at locations of x+3, x+4, x-3 and x-4.

After the measurement, an adjustable threshold is applied to the log difference of edge intensities to eliminate unreasonable edges having low contrast, so that edges having high contrast are processed for feature identification. For the convenience of the later process, the highest logarithm edge intensity is referred to the brightness (B_b) of the bright side, and the lowest logarithm edge intensity is referred to the brightness (B_d) of the dark side. The highest and lowest edge intensities are stored to be the information of logarithm difference of edge intensity for a found edge, and are used to calculate the logarithm average edge intensity which is introduced in the next step, feature identification.

At this point of the process, the detailed edge information has been extracted. This includes edge position, edge brightness transition direction, and the highest and lowest edge intensities. The next step, feature identification, relates the found features with the stored target features, will be described in Chapter 5.

Chapter 5

Recognition Implementation

Chapter 4 described the information extraction processing. This chapter describes the final phase of the recognition procedure in detail, which is the feature identification process, and then shows how the system software programming is implemented.

From the process of information extraction, the detailed edge information has been obtained. The *feature identification* process compares the obtained edge features with the desired target feature to be identified to decide if the desired target exists in the real scene.

5.1 Feature Identification

Feature identification or recognition refers to matching measured parameters with image data stored in a target information file which is used to represent the target feature to be found. The measured parameters are the information generated from the previous steps, which include the edge position, edge brightness transition direction, the highest and lowest logarithmic edge intensities. The data in the target information file, which is similar to the measured parameters, is defined on the concept of a primary edge with a minimum and a maximum allowance for individual measured parameter. The recognition process tests each edge against the criteria defined in the information file.

Fig. 5.1 shows a line of primary information for a switch plate. The edge positions are represented by x_1, x_2, x_3 and x_4. The edge brightness transition directions are indicated by arrows in such that a left-up arrow represents a brightness transiting from a light to dark area and a right-up arrow represents a brightness transiting from a dark to light area. The lowest and highest logarithmic edge intensities are represented by I_{1n}, I_{1x}, I_{2n}, I_{2x}, I_{3n}, I_{3x}, I_{4n} and I_{4x}. Using the lowest and highest log edge intensities, one can derive the log average edge intensity, which is the average of the lowest and highest log edge intensities.

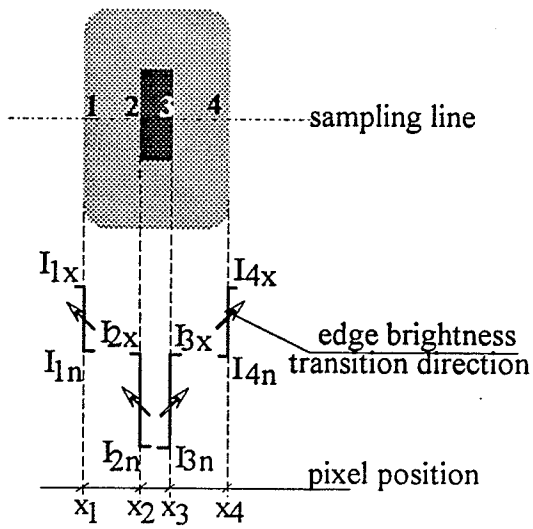


Fig. 5.1 illustration of obtained data

The information for the feature of the switch plate shown in Fig. 5.1, is stored in the target information file. All numerical quantities of the information are given with a minimum and a maximum range of allowance. The data in the target information file is shown in Fig. 5.2 and is as follows:

For the primary edge:

1. the log difference of edge intensity (h_1) which has to be higher enough to be distinct from other weak edges.
2. the edge brightness transition direction (left-up arrow).

For the second edge:

1. the sign indicating search direction (as shown in numerical sequence) and the distance (l_1) in pixel number from the primary edge.
2. the log difference of edge intensity (h_2).
3. the edge brightness transition direction (right-up arrow).
4. the difference ($H_1{}^2$) between the log average edge intensity relative and primary edge's log average edge intensity.

The data for the third edge contains:

1. the sign indicating search direction, and the pixel distance ratio (l_2/l_1) which is defined by the pixel distance (l_2) from the primary edge against the pixel distance (l_1) from the primary edge to the second edge, with a small amount of pixel adjustment that makes it easy to recognize two edges which are close to each other.
2. the log difference of edge intensity (h_3).
3. the edge brightness transition direction (left-up arrow).
4. the difference (H_1^3) of log average edge intensity related to log average edge intensity of the primary edge.

The data for the fourth edge is the same as the data for the third edge.

Note that the library data file can still be used if it has data for only one of the 3rd or 4th edges.

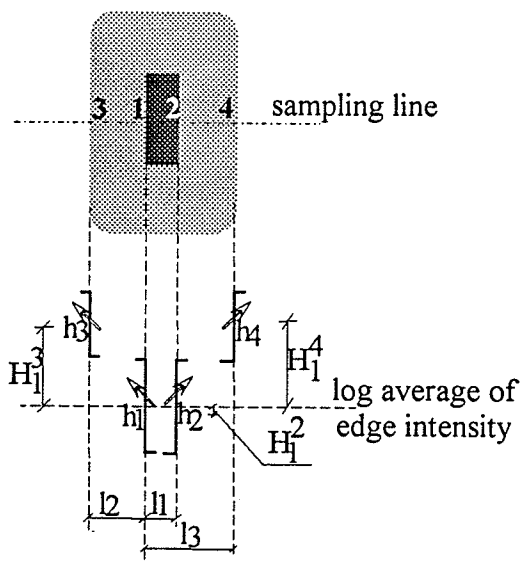
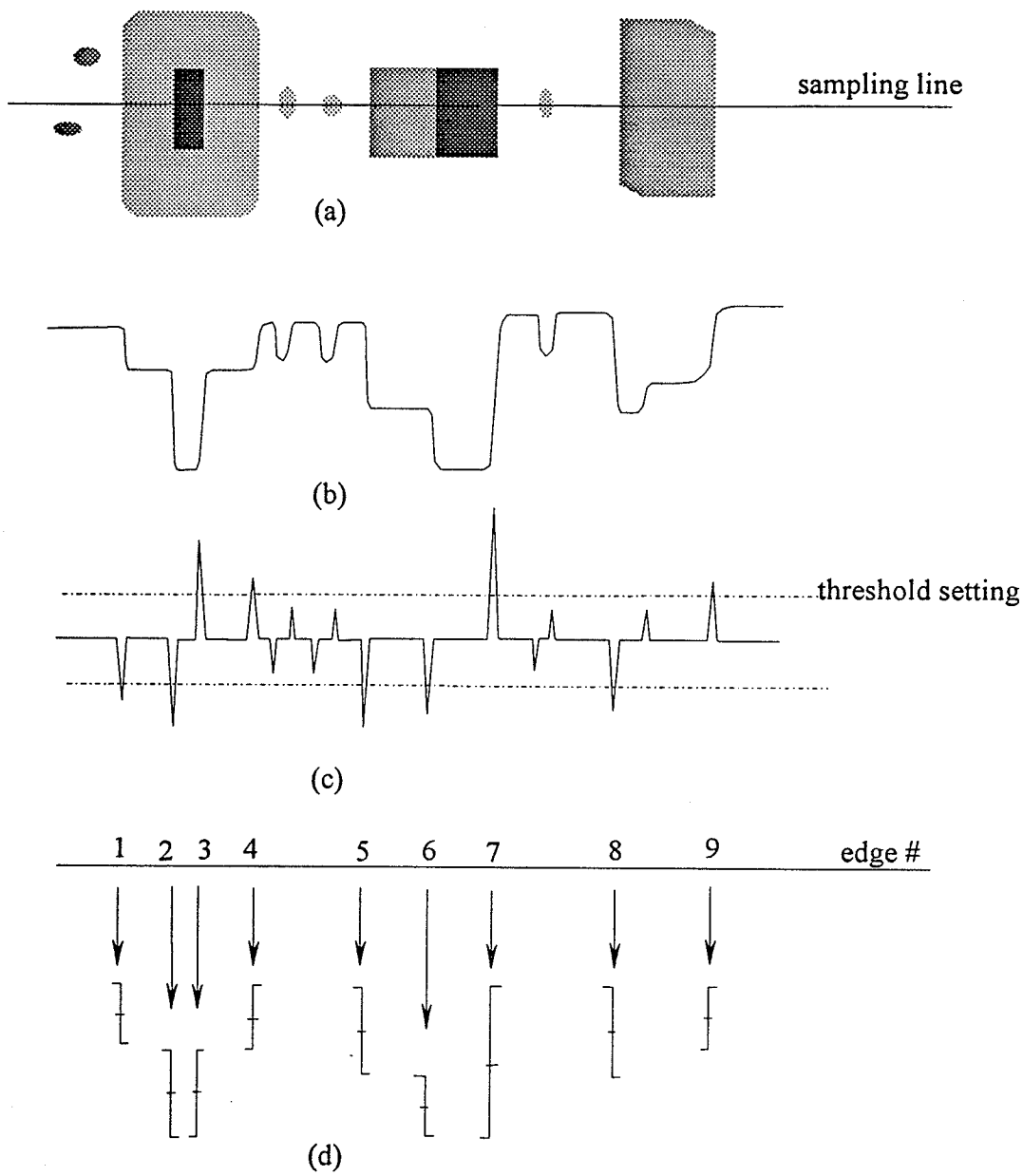


Fig. 5.2 data given in the target information file

In describing the matching process, one can refer to Fig. 5.3. In Fig. 5.3, (a) shows an image picture scanned by a camera, (b) shows a gray level profile for a sampled line

across the image, (c) shows its differential profile and (d) shows the obtained edge information after thresholding. The comparison of the target feature is performed as simple additions and subtractions to the logarithmic brightness information. The process for finding the feature of a switch plate in Fig. 5.2 is as follows:



(a) image picture. (b) gray level profile across a line.
(c) differential profile. (d) obtained information

Fig. 5.3 The illustration of the information obtained across a line of an image

Initially, the process considers every possible edge as the primary edge across a line of image. Every edge is tested, until an edge whose log difference of the edge intensity and edge brightness transition direction fit the given data for the primary edge specified in the library, has been found. Considering the obtained edge information shown in Fig. 5.3 (d), the primary edge could be at edge #2 or edge #6. Edge #8 and edge #5 could be chosen as primary edged depending on the range of acceptable values for the primary edge.

Once the primary edge has been found, searching for the second edge resumes by comparing the measured and the defined parameters. The edge to be searched must be on the right side of the primary edge and within a minimum and maximum pixel distance. In the event of the edge being tested falling within the defined pixel distance range, the log difference of edge intensity is measured and the brightness transition direction is checked. If the brightness direction is correct and the log difference of edge intensity is a fit, the difference of log average edge intensity related to log average edge intensity of the primary edge is verified against the defined data.

In relating to Fig. 5.3 (d), if edge #2 is selected as the primary edge, the second edge might be found at edge #3. Edge #7 could also be found as the second edge if the primary edge is chosen at edge #6. Suppose the pixel distance between edge #6 and edge #7 is a fit, the maximum log difference of edge intensity for edge #7 is not within the defined limit, so it would not be accepted as the second edge. Edge #9 could be second edge if the primary edge is at edge #8.

Searching for the third edge begins after finding the primary and second edges. To fit the specified information data, the possible third edge must be on the left side of the primary edge; and it must also be within a distance which is defined by the minimum and maximum ratios of the measured pixel distance between the primary and third edges to the measured distance between the primary and second edges. A small pixel number adjustment is also available. The relationship is represented by equation 5-8:

$$l_{1m}*R_{min} - Px < l_{2m} < l_{1m}*R_{max} + Px \qquad (5\text{-}8)$$

showing that the measured distance between the primary edge and third edge (l_{2m}) must be greater than the measured distance between the primary and second edges (l_{1m}) times the minimum ratio (R_{min}), minus pixel distance (Px), and less than the related maximum ratio plus Px.

If the pixel distance satisfies equation (5-8), the log difference of edge intensity, the brightness transition direction and the difference of log average edge intensity are tested against the specified criteria. If one looks at the possible match from Fig. 5.3 (d), edge #1 can be considered as a candidate for the third edge in reference to primary edge #2, provided the distance ratio is set to 2.0. Edge #5 could be chosen as a possible third edge, if edge #6 has been chosen as a primary edge and if edge #7 had been identified as a second edge. Edge #7 fails when considered with edge #8 as primary edge and edge #9 as secondary edge because the log difference of average edge intensity related to edge #8 and the edge brightness transition direction do not fit.

Searching for the fourth edge could be performed either before or after the searching of the third edge. In Fig. 5.3 (d), edge #4 matches parameters defined for the fourth edge related to the primary edge #1. The matching result tells that edges #1, #2, #3 and #4 possess the feature for the switch plate in the library file. Their positions and brightness transition directions are stored in a structured array, and are useful for further target verification. Other edges which do not fit defined features are discarded.

In this process several constraints employed to create the library file are used for target verification. They are as follows:

1. The minimum level of edge brightness transition ratio: this constraint makes the search ignore any edge which is small enough to be caused by a mottled surface or other surface noise.

2. Edge brightness transition direction assumption: it specifies the target appearance we are looking for. Fig. 5.4 illustrates two types of object appearance: (a) one with a gray rectangle in a black ellipse background, and (b) one whose gray levels are opposite to (a). Both will be recognized if the transitions between black and gray are used for the feature to be recognized, without using the edge brightness transition direction statements. This constraint isolates a target feature from others which have a negative contrast and an identical spatial relation as the target feature.

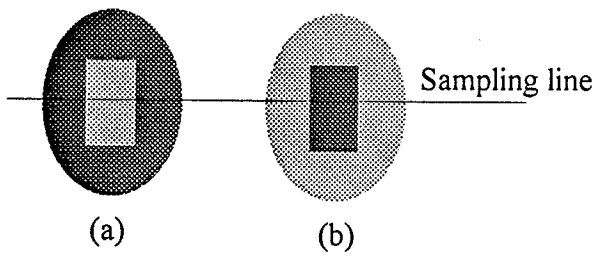


Fig. 5.4 Illustration of edge transition direction

3. The difference of log average edge intensity related to the primary edge: this measurement relates the contrast ratio relation of the tested edge feature to a specific edge so the general brightness level does not affect the recognition process.
4. Spatial relations between edges: it provides geometric ratios to remove the effect caused by different object and camera distances, and still defines the uniqueness of the target feature.

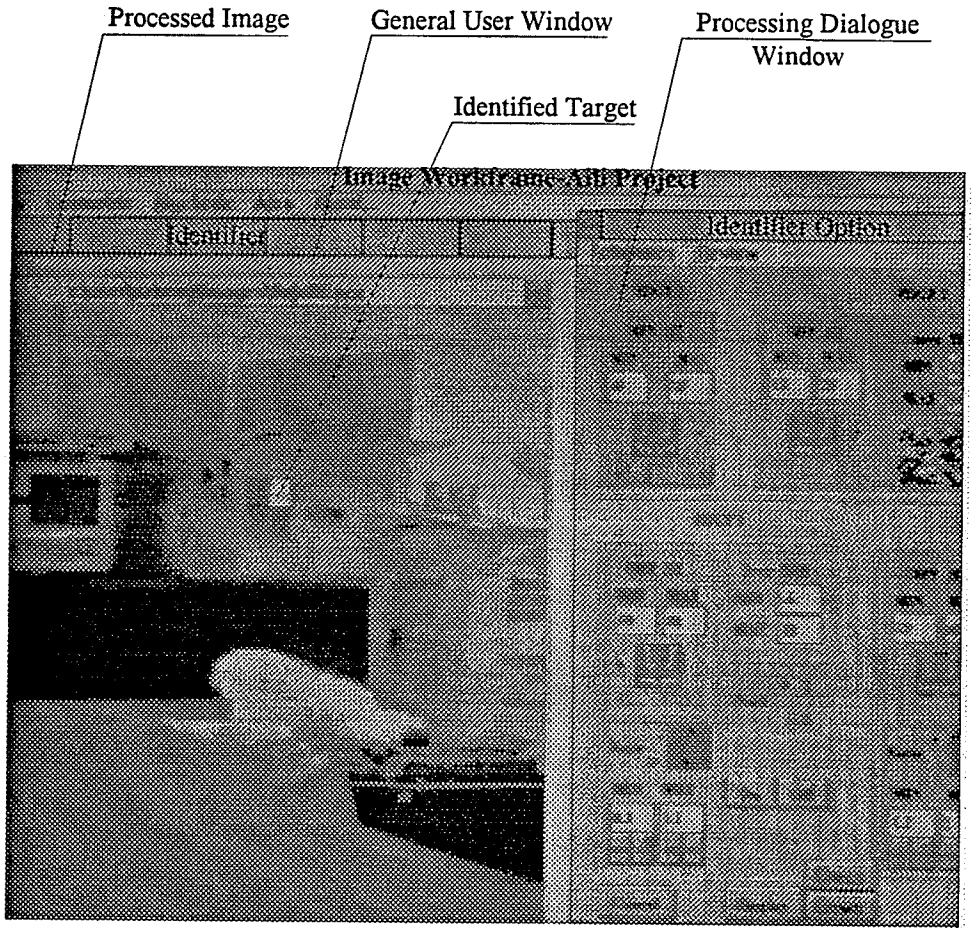
5.2 Software Implementation

The software of the system is written in the C language. It is combined with the MS-Windows application programming interface, and executed in the MS-Windows environment. The software code is designed to be modular, and the running program does

not interfere with other applications. Fig. 5.5 gives an example of the executive program running under MS-Windows on a 80486 PC computer.

The executive program runs within a general purpose user window that provides supports for the interfacing of software with video hardware such as cameras and image capture boards.

The processing dialogue window, part of the executive process, contains the signature of a target feature, which is used to match any potential targets. The target feature contains information describing two to four edges. The image to be processed is displayed in a format of array with 512x480 resolution with 8 bit gray level.



The processing dialogue window displays and interfaces with the library data file. It contains four quadrants to display information of a target feature. In four quadrants, each shows the ideal information from the library for each edge.

Fig.5.5 Program running under Windows

(a higher quality graph is available in Appendix D)

5.2.1 Flow Chart of the Recognition Algorithm

The recognition procedure for a sampled line can be described by the block diagram as shown in Fig. 5.6. It contains the following major parts:

1) image transformation; 2) finding potential edges; 3) measuring edge intensities and 4) matching target features.

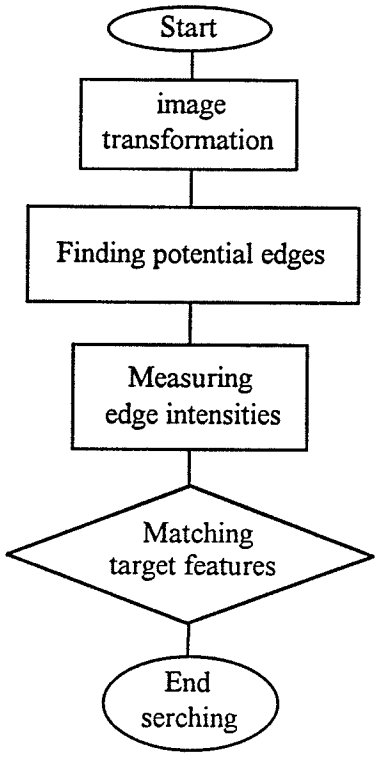


Fig. 5.6 General flow of the recognition process

The first block corresponds to the discussion in section 4.1. It performs a brightness conversion by applying a logarithmic gray scale conversion table to the input image and generates an output image which has logarithmic gray level.

The second block relates to the discussion in section 4.2. At the end of this step, potential edges are created and their positions are saved as structured vectors along with their brightness transition directions for later processing.

The third block of the diagram represents the log difference of edge intensity measurement which is stated in section 4.3. The obtained data from this step are the highest and lowest edge intensities, stated in the logarithmic scale. This information will be collected in a structured vector along with edge positions and edge brightness transition directions for the final feature verification.

The final step, shown as a diamond, is the matching of target features. This step compares the measured parameters with the criteria defined in a target information file in

order to find possible target features that fit the specified features. Edge groups that fit the target statement defined, have their information stored.

5.2.2 Programming Implementation

In programming the recognition algorithm, the details of Fig. 5.6 are shown in the flow charts of Fig. 5.7 and Fig.5.8. Fig. 5.7 includes the first three steps of the recognition process, and Fig. 5.8 shows the process of the feature identification. The target searching implementation follows:

The search starts from the left top to right bottom of an image in order to process image data line by line, so that any potential target feature in the scene would not be missed.

A slice of image is scanned, and the gray levels of it are scaled to logarithmic gray levels. The differential value at each pixel in the image is calculated, and then the possible edge is selected by applying a threshold to each differential value. The edge determination process is performed on pixels whose differential values exceed the threshold. After edge determination, potential edges are produced and their information data including their positions and brightness transition directions are kept in a structured array. The log difference of edge intensity measurements follow after the potential edges have been found, and the measured information will be kept until feature identification begins. The flow chart of Fig. 5.7 corresponds to this part of the search.

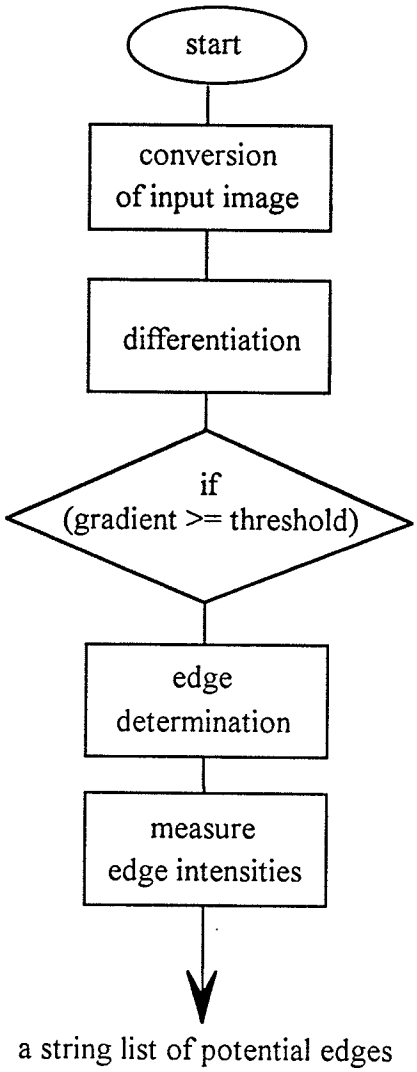


Fig. 5.7 Flow chart to obtain edge information

From the above search, a string list of potential edges are created. The target feature matching process applies criteria, called identifiers, to the potential edges being tested, so that any potential target features which match defined features are identified. This scheme is shown in Fig.5.8.

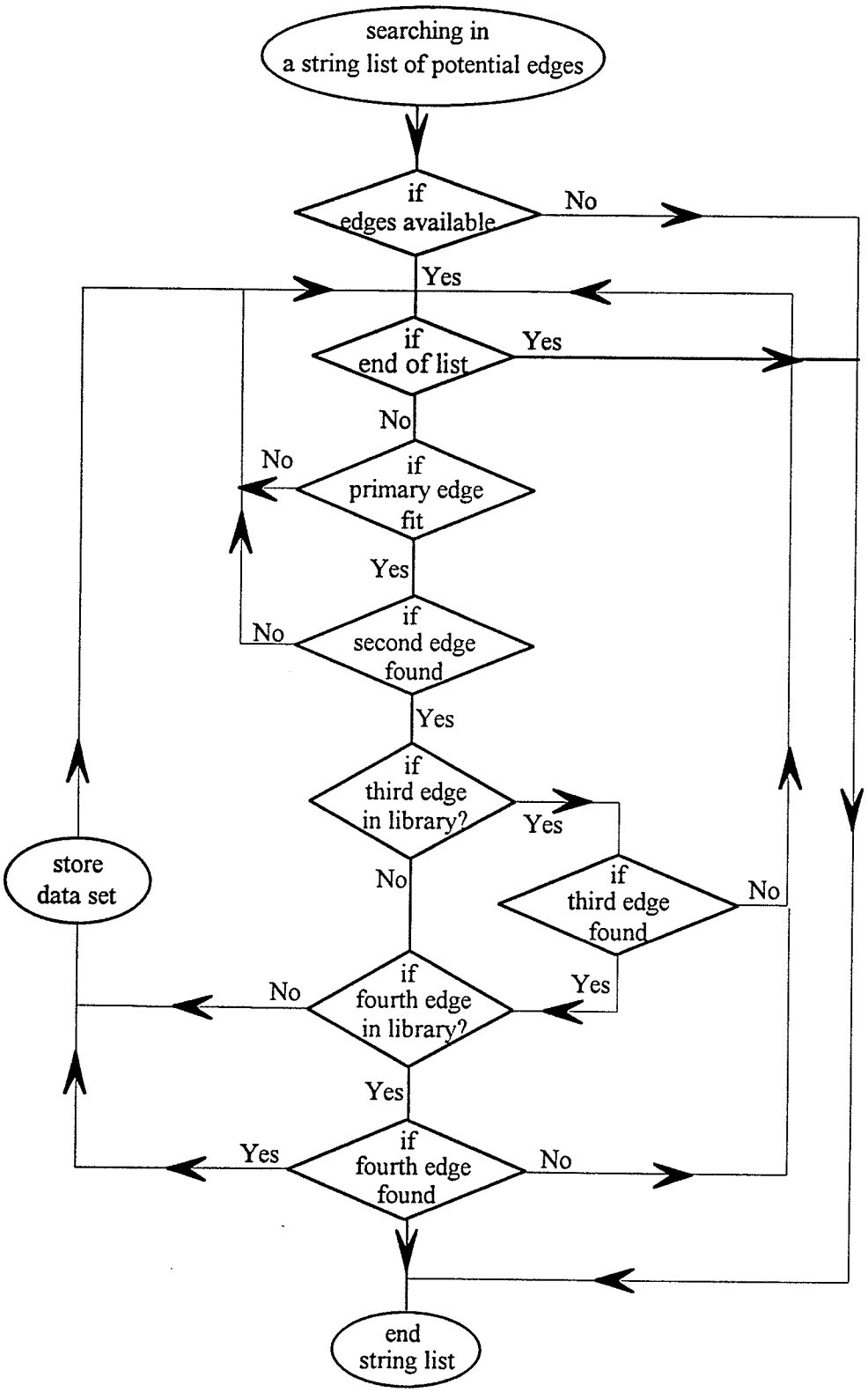


Fig. 5.8 Flow chart of target feature matching

The target feature matching scheme searches a primary edge from the string list until an edge matching the library information has been found. Once the primary edge has been found, the search for the possible second edge begins along the direction specified in the target information file. If no suitable match occurs within the allowable pixel distance range specified, the search looks for the next potential primary edge and repeats the same search scheme for the primary and second edges. The potential third or fourth edge's search follows the successful completion of the second edge search. If the data of the third and fourth edges defined in the information file are matched by the measured data, the data set is saved; otherwise the search process repeats for the next potential edge.

The search scheme continues along every scanned line till the last line of the image has been searched. To prove the target uniqueness, a different identifier is applied vertically to the local areas where the horizontally identified features have been found. The same scheme as for the horizontal search is used for the vertical search. The final decision is made based on the number of lines having the required feature at the same horizontal or vertical positions. If the numbers for both identifiers exceed certain numbers, the target is claimed as a found target having the same feature as desired target feature.

So far the theory of the recognition procedure has been described. In the next chapter, a number of experiments using several targets are demonstrated and their results are discussed to show the suitability of the vision system.

Chapter 6

Experimental Results

In Chapter 5, we have described the recognition algorithm of the vision system along with its software implementation. In this chapter, we present the results of experiments conducted under various conditions to test the feasibility of the vision system. The targets used for these experiments are, a gray pattern which has been used to verify the behavior of the system during the system development period, and some objects in the real world such as an electrical outlet and a switch plate.

The information is presented in the following order: the test environment and method, the image processing analysis result along a scanned line, the calibration of the image acquisition system, the test results, and finally the discussion of the test results.

6.1 Test Environment and Method

To validate the system for feature recognition purposes, the test environment allows the following conditions to be created:

1. variations in lighting conditions.
2. presentation of objects having a variety of shapes.

The experiments are conducted to verify the vision system's performance in this typical environment, with different pictures taken at different camera locations and orientations. The software program of the system searches for desired targets appearing in the camera field of view.

In experiments, a scene is captured under a variety of lighting conditions, with different camera locations and orientations. The amount of light reaching a camera sensor can be adjusted by 1) adding or reducing the lighting in an environment, and 2) adjusting the camera lens opening size. A scene taken at different locations and orientation can be obtained by three ways: 1) by changing the camera distance to the object, 2) by moving

the camera horizontally with an offset from the previous location, and 3) by changing the camera horizontal and vertical viewing angles.

6.2 Image analysis result before feature identification

Fig. 6.1 illustrates a line of image analysis across a scanned image, which has been displayed on the computer monitor. The upper area shows a graph of the gray level intensities of the horizontal line in logarithmic scale; the middle area presents a graph of its differential profile; and the lower area shows a graph about the log difference of edge intensities which exceed a threshold setting.

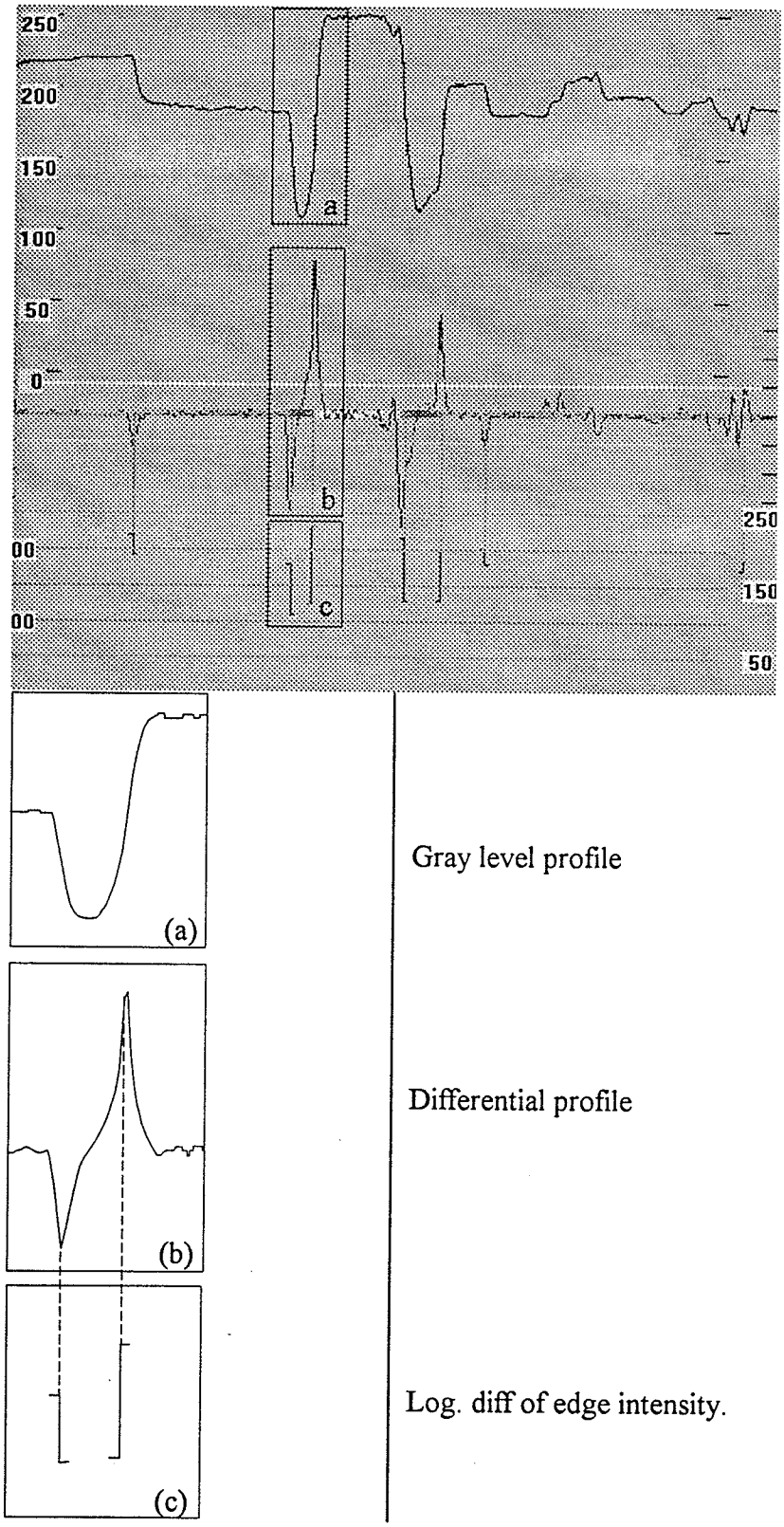


Fig. 6.1 A line of image analysis

6.3 Calibration of Image Acquisition System

As described in section 4.1, this calibration is necessary when images are grabbed by different frame grabbers or cameras. The major steps include the calibration of the frame grabber and calibration of the entire image acquisition system. The calibration of the frame grabber ensures the dynamic reading range of a frame grabber and the dynamic sensing range of the camera match each other. The calibration of the entire image acquisition system creates a table information relating the input light intensities to the frame grabber outputs.

The calibration of a frame grabber is done by checking the lowest and highest readings from the frame grabber, corresponding to the maximum and minimum amount of light intensity arriving to the camera. From the frame grabber point of view, the minimum light intensity could be when the camera lens opening is closed representing extremely low light condition; the maximum light intensity could be the highest light level it can read before saturation. The lowest and highest readings are set at approximately about 0 and 255.

The calibration of the entire image acquisition system first creates the table by having different sets of readings presenting different intensities of light to the camera sensor and reading the frame grabber outputs, and then checks if the outputs against the incoming light intensities are smooth.

To acquire different sets of readings for the calibration, the following steps must be followed:

1. Take an image using the camera being calibrated, and adjust its aperture and focus to an appropriate setting, so that the brightness reading at the brightest area to be measured is approaching 255.

2. Select a few (e.g. 2) areas having different brightness intensities, and take an average of several brightness readings in each constant brightness area to obtain a more reliable reading.
3. Change the camera aperture index to give a factor 2 of the incoming light intensity, without changing the camera's location, and repeat step 2 for the same reading locations until all readings have reached zero.

After these reading sets have been collected, they are scaled and graphed. If these curves fit to each other and follow the shape of a single smooth curve, this indicates that the incoming light intensity matches the brightness reading and further indicates the manual settings of the lens are reasonably accurate. Otherwise an adjustment of the camera lens setting is needed to obtain smooth readings.

This calibration is a rough measure of how the output brightness responds to the input light intensity, and is of sufficient accuracy for the recognition process.

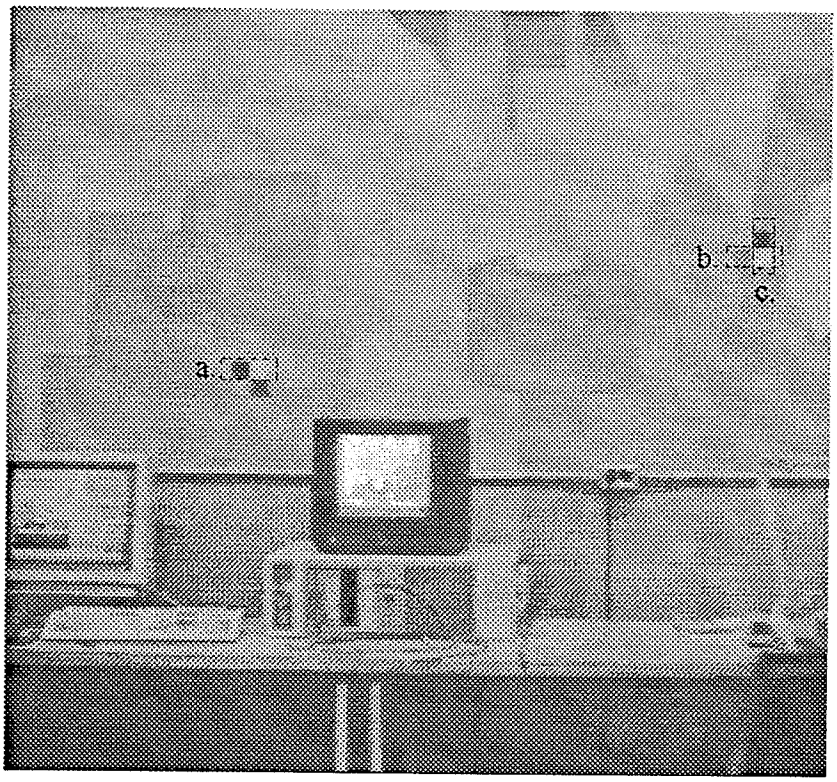
6.4 Test Results

The aim of the experiments is to verify if the system works well regardless of the variation of lighting and camera locations. The tests took place at the Vision Laboratory in the University of Manitoba. The camera lens used has a focal length of 16mm. In the tests, the camera orientation is adjusted manually so that a scene of interest can be taken from different viewing positions. A total of 50 images have been used to test our system functions; these images include images used during the system development phase. Here, a few images containing the targets of interest, are presented to demonstrate the performance of the system, and more images are available in Appendix D.

6.4.1 Recognizing Gray Targets

In order to validate the system, the primary tests use the gray pattern described in section 4.3.2. A number of images were taken from different camera locations and orientations, with different levels of illumination.

Fig. 6.2 shows an image taken from 5.5m away from the target plane; it contains two identical targets with different target orientations. The orientation of the pattern on the right-top of the image is 90° clockwise to that of the one on the left-bottom. The areas a, b and c are of interest and need to be recognized. With the same scene and same distances, more experiments were conducted, using images taken at different horizontal and vertical viewing angles and horizontal viewing locations. The system performance showed that all areas of interest were recognized respectively by the system with all the views, with separate identifiers for areas a, b and c.

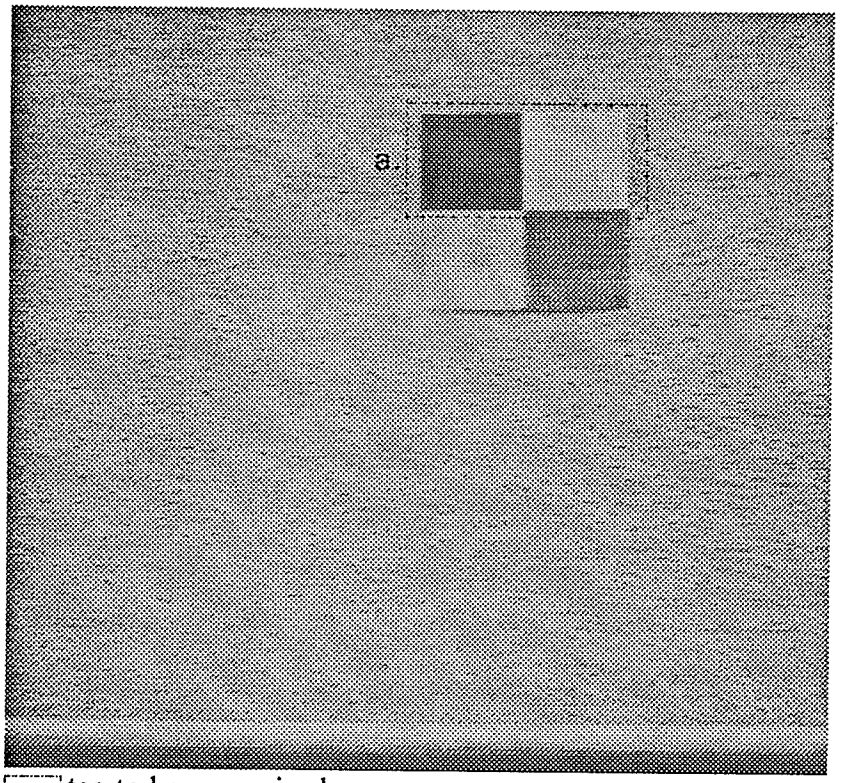


tag to be recognized

Fig. 6.2 An image contains two identical gray patterns with different target orientations

Fig. 6.3 contains the same target as that shown on the left-bottom of Fig. 6.2 a, except that it is at a distance of 1m. The system was capable of recognizing the same area

(a) as one in Fig. 6.2, using the same identifier as used for Fig. 6.2 a. Using the same identifier, the system was also able to recognize the same area (a) from an image which was taken from the left side of the same scene at the same distance .



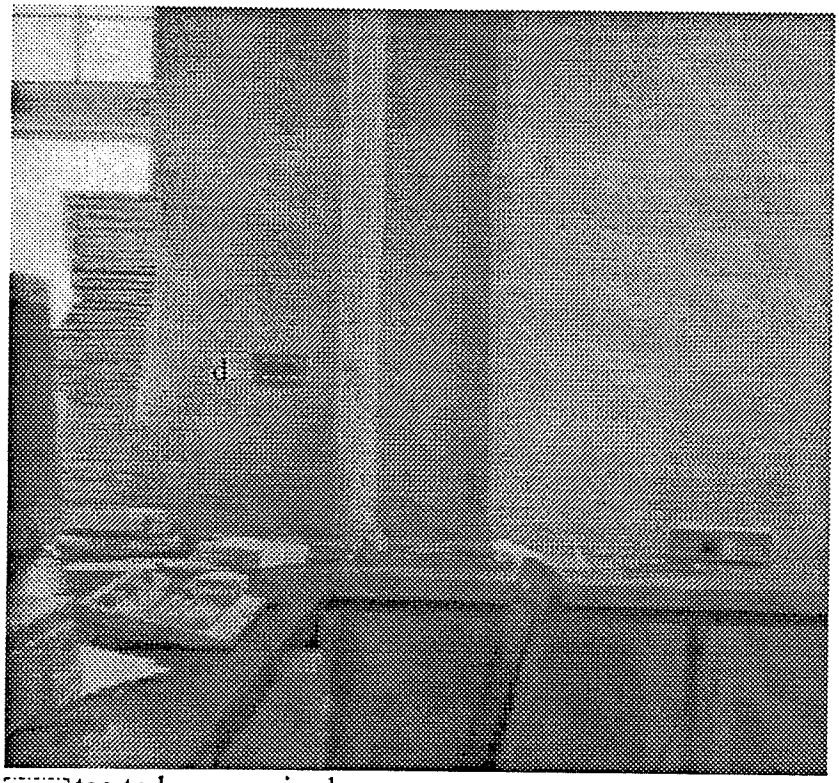
tag to be recognized

Fig. 6.3 An image contains the identical target to Fig. 6.2 a and was taken at a distance of 1m

In addition, other experiments have also been conducted under different lighting conditions, using target pictures done at many of locations used for the standard bright level picture. The results showed the system worked well by using the same identifiers as previously used. The comparison of images having different levels of illuminations can be seen in Appendix D.1 and D.2.

6.4.2 Recognizing an Electrical Outlet

An electrical outlet fixed on a wall of the vision lab, was also used as a target to confirm the possibility of applying the vision system to manufacturing environments. The electrical outlet being recognized has two plugs in it. The feature to be identified which is marked with d, is across the middle region of the outlet, as shown in Fig. 6.4.



tag to be recognized

Fig. 6.4 An image including an outlet at a distance of 5.3m

Fig. 6.5 shows the feature of an electrical outlet to be recognized. The search sequence specified in the target information file is set as follows: the primary edge is set at the left-side rim of the left plug, the second edge is set at the most left outside rim of the outlet plate, the third edge is set at the right-side rim of the right plug, and the fourth edge is set at the most right outside rim of the outlet.

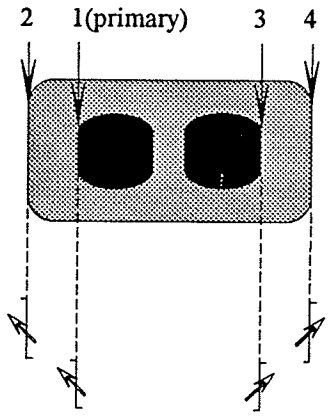


Fig. 6.5 An outlet feature to be recognized

The experiments consist of two steps:

1. use an image taken at 3.5m to set up the experiment first.
2. use an image taken at 5.3m without changing lighting and the same image as step 1 with a brighter illumination.

The system successfully recognized the electrical outlet in step 2 using only one identifier created in step 1. In both steps 1 and 2, the same outlet was recognized as a target. This result indicates that the system can ignore features that are mixed with the features used for recognition, and further indicates that the system has the potential to use realistic objects for references to automatic vehicles. Fig. 6.4 shows one of images used for the first step test. Appendix D.4 shows the image used for the second step test.

6.4.3 Recognizing a Switch Plate

During the development of the system, a switch plate was used to test the system. The vision system tried to recognize this type of target through a number of different conditions. Fig. 6.6 shows one of the images containing a switch plate that is fixed on the wall of the vision lab, which was taken at 3.9m from the target plane.

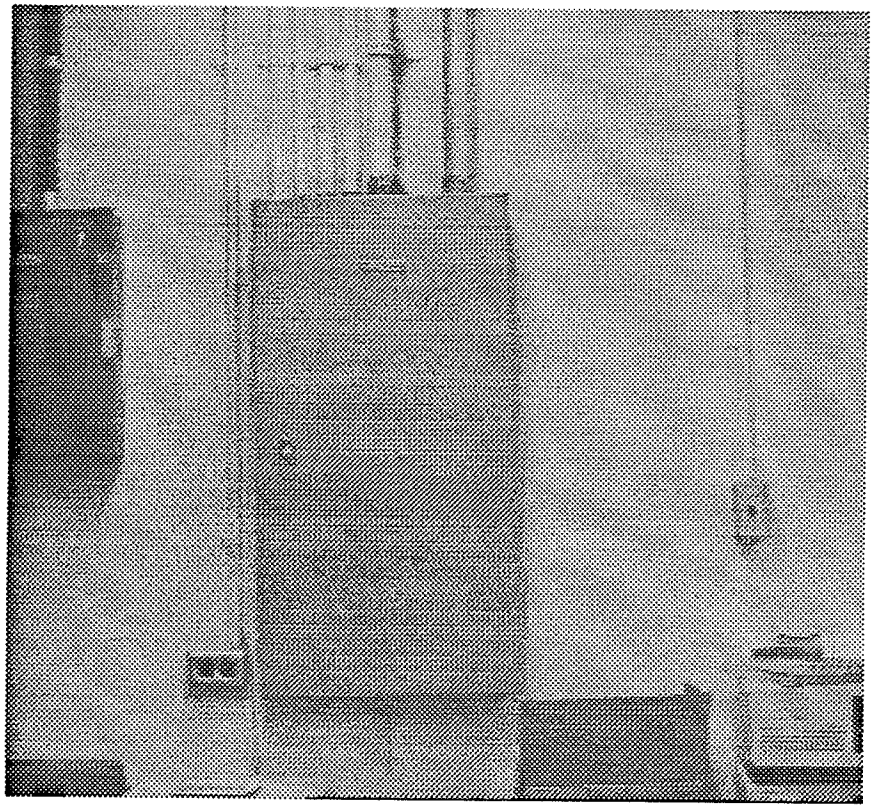


Fig. 6.6 An image containing a switch plate

The system successfully recognized the same switch plate using the same target information identifier when the camera was set at 3.9m and 2.3m respectively. With the same scene under partial lighting conditions, similar experiments were conducted using light coming from the ceiling to the left or right of the camera. It was found that the reflection from a shin object of the light incident upon on it could affect the recognition results, as will be discussed later. With the lighting coming from different directions, the system successfully recognized the same switch plate using different target information identifiers, but failed when using a single target information identifier.

6.4.4 Recognizing other alternatives

In addition to the above mentioned targets, the system also was tested to recognize other targets such as a wood framed blackboard corner and a marker on the back of a machine.

Fig. 6.7 shows an image containing a blackboard corner on the wall of the Vision Lab. In order to recognize the blackboard corner, the system used two different identifiers horizontally and vertically. If both them can be found in the image, a further process relating the geometry relation of the horizontal and vertical features, could be performed to confirm the existence of the potential corner.

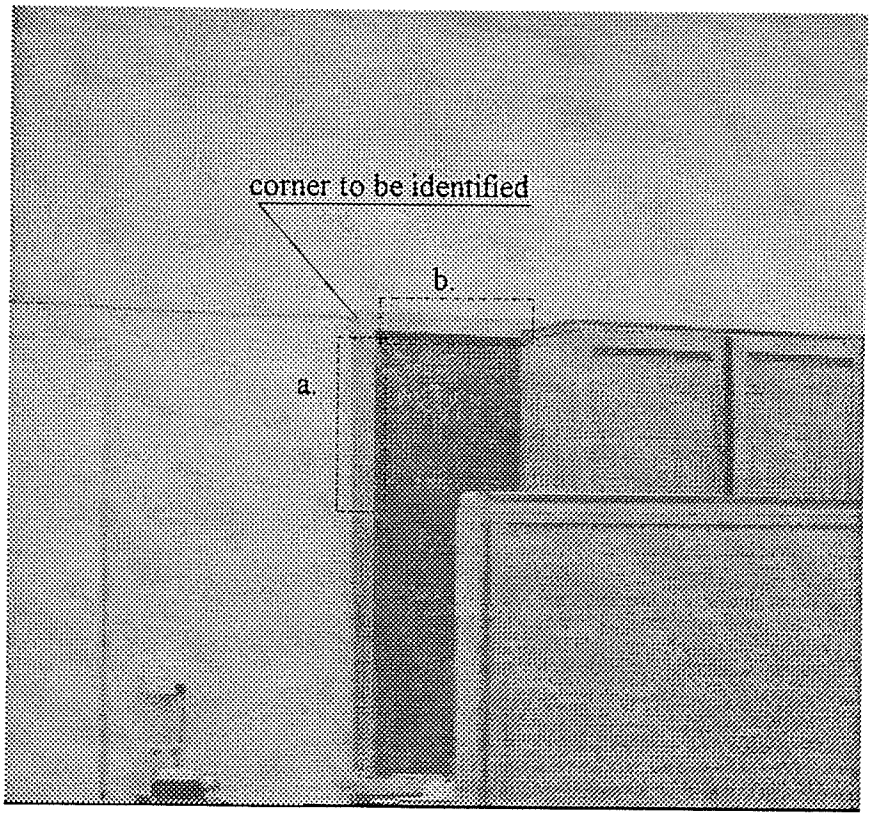


Fig. 6.7 An image containing a corner of a blackboard

A mark sticker on the back of a computer which is similar to a mark on manufacturing machines, was used as a target to test the system. Fig. 6.8 shows this type of target. The vision system recognized such mark in horizontal scan at distances of 1.5m and 3.6m, with the existence of other features. The system can confirm the uniqueness of found target if another geometry statement is used for a vertical scan recognition with the distance between edges set relating to the distance found in the horizontal scan.

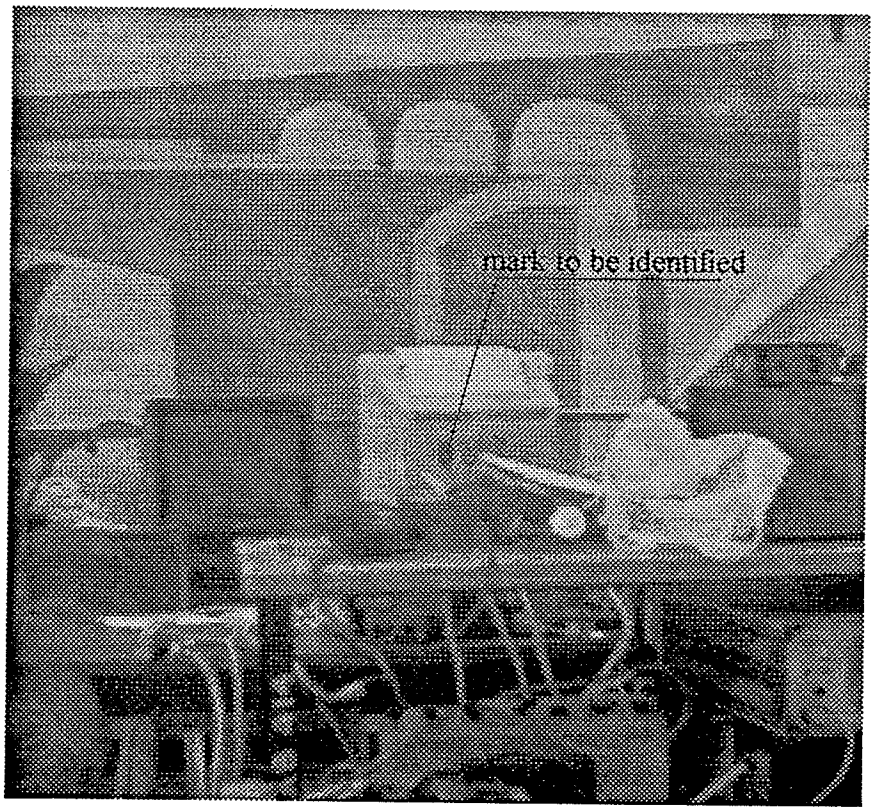


Fig. 6.8 An image containing a machine mark

6.5 Discussion of the Test Results

6.5.1 The Processing Time

The total processing time, composed of the logarithmic transformation time and the recognition process time, is approximately 7.8 seconds if the recognition process scans every line. This includes 5.4 seconds for the logarithm transformation and 2.4 seconds for the recognition process. If the program runs every third line, the processing time is about 2.4 seconds which includes 1.7 seconds for log transformation and 0.7 seconds for the recognition process.

Currently, these results are obtained on a 33 MHz 80486 PC without putting any efforts to speed up the processing time. If the system runs on new PENTIUM based PC , the processing speed would be about 8 times faster than the current speed, i.e. the

processing time is shortened to 0.24 seconds for every third line scan. This result indicates that the proposed system is promising for real time operation and can be used effectively in manufacturing environments.

6.5.2 Recognition Results

It should be noted here that the system output results could be affected by the following restrictions:

1. The image acquisition limitation caused by the lack of camera frequency response and lack of synchronization between the frame grabber and camera along a horizontal line.

   Because the camera has a limited frequency response, the shape of video signal is rounded, as seen in Fig. 6.9.

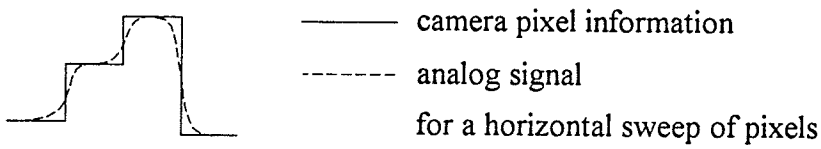


Fig. 6.9 Degradation of video pixel information caused by limited bandwidth of the analog video signal

The frame grabber follows the standard specification of NTSC (National Television System Committee) to receive camera video signals. This means the video format image is read out on a horizontal line-by-line basis. Each line is separated from the next by a horizontal synchronization pulse which indicates the start of a line, which means that the digitization of a frame grabber is synchronous with camera video signals in the vertical direction. Along video signal in the horizontal direction, a pixel clock in the frame grabber drives the A/D converter. There is no synchronization between the camera pixel to horizontal analog signal conversion and the frame grabber digitization.

The above limitations causes the brightness between adjacent pixels in the image to blend each other and results in three pixels representing an edge transition vertically but five pixels horizontally. That explains why we move two pixels from the center of an edge when measuring log diff. of edge intensity. Also, if an identical differential threshold is applied to a horizontal and a vertical sweeps, edges having the same brightness ratios may be detected differently depending on the threshold setting.

2. The unbalanced illumination level on both sides of a shining object, or light falling upon the object affects the system recognition outcome. A typical example of this effect is shown in Appendix D.4, where the light comes from the right-top of the ceiling relative to the switch plate causing a shadow on the left side of the switch plate.

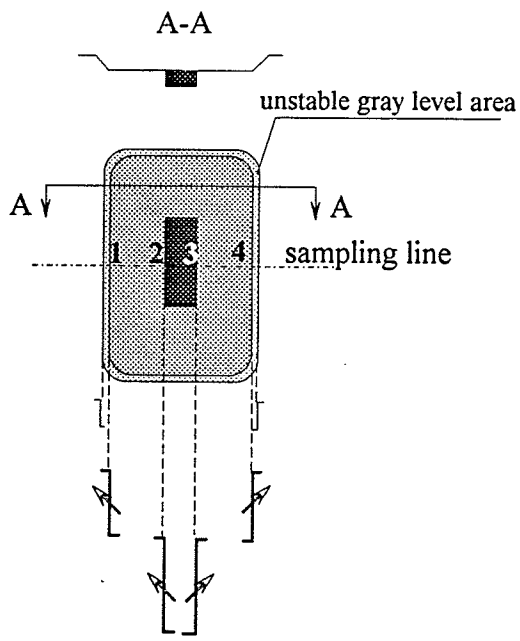


Fig. 6.10 The light incidence effect on a switch plate

Fig. 6.9 shows this effect on a switch plate. The brightness intensity of unstable gray level area depends on the light incidence direction. Since this

area is narrow, this light direction effect causes slight changes in the magnitude of edge brightness ratio and parallel position for the outermost edges no. 1 and 4. In order to achieve a better performance, it is suggested that the system use a few identifiers for one item when a single identifier of the same target does not work, so that the target could be recognized for these situations. A positive outcome of this is that the system could also note when some of the lights normally used are turned off.

Chapter 7

Discussion and Conclusion

This chapter discusses the proposed system's limitations, along with further work which would make the system more able to carry out the task of vehicle guidance. A summary of the system's features are given, followed by the conclusion.

7.1 Discussions

In Chapter 6, system test results were presented with a short discussion. This section discuss the hardware limitation to the system, and the possible improvements to the presently developed software.

7.1.1 Hardware Limitations

As stated in section 6.5.2, an image is digitized by a frame grabber that is not synchronized with the pixels in horizontal scan. If a synchronous frame grabber or a digitization camera becomes available, the image blur problem caused by the lack of synchronization can be solved. This would increase the reliability that the system precisely detect targets in the camera field of view. In addition, the currently used frame grabber digitizes an image in 8-bit or 256 levels. However, if a 10-bit frame grabber becomes available, the jumps in brightness reading at lower brightness level as described in section 4.1, would be reduced, and the system would have a greater dynamic brightness range which allows a higher brightness level and a lower brightness level.

7.1.2 Software Improvements

The system is limited to recognizing patterns having no more than four repetitive edges. In some cases, all repetitive edge features of a target need to be recognized for the unique recognition of the target. Examples are those object having symmetrical geometry features as shown in Fig. 7.1.

(a) (b)

A triple switch plate A symmetrical pattern

Fig. 7.1 Symmetrical geometry features

Fig. 7.2 shows the obtained edge feature for a triple switch plate. This type of recognition would be required to tell the difference between a switch plate having three switches with a plate having only the 1st and 3rd switches. This pattern problem is also illustrated in Fig. 7.3.

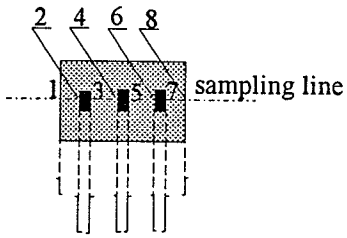


Fig. 7.2 Obtained features for a triple switch plate

In order to recognize this type of target uniquely, two techniques could be carried out based on the present software. The first technique could make the software capable to identify all repetitive edges of the pattern, i.e. all edge no.'s from 2 to 7 in Fig. 7.2 would be recognized at the same time.

The second technique could have a three step recognition process. The first step determines the existence of the switch plate with the two outside switches, e.g. the 1st and 3rd. The second step identifies two switches close together, e.g. the 1st and 2nd or 2nd and 3rd. In performing these two steps of recognition, the recognition process could use a primary identifier horizontally to find two vertical bars which are in certain distance from each other, and then apply a second identifier in regions determined by the first identifier.

If the second identifier is found in a region, a potential pattern feature required exists. The third step verifies the logical relationship for the switch plate.

An extension of this second technique allows a greater variety of targets to be recognized. This extension will be discussed below.

The present system, in some cases can not exclude a pattern which is not supposed to be recognized, if it is in some ways similar to a target which is recognized. Fig. 7.3 shows two similar patterns. If Fig. 7.3(a) is the pattern to be recognized, the system will recognize both patterns as the recognition process ignores features that are mixed with the features used for recognition. Therefore, the software also needs to be improved so the system would exclude any patterns that have extra features. This could be solved using a similar process as mentioned for Fig. 7.1 and 7.2. If the second step of recognition process finds black spots close together, the pattern is not considered the target of interest.

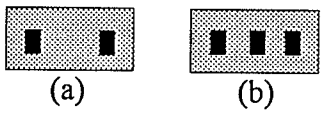


Fig. 7.3 Patterns to be distinct

In the experiments, only a few items have been tested as target candidates for the proposed vision system. More targets need to be explored when it is put into use. One of the improvements of the software to be done, is to build a structured statement for each type of target so that the recognition process can classify target groups. To do this, a general three step recognition which is similar to the recognition for the pattern shown in Fig. 7.2, could be carried out:

1. scanning the image using a primary identifier in a horizontal or vertical direction.
2. using one or more secondary identifiers, to confirm the existence of a pattern in the regions where the primary identifier has been found.
3. verifying the dimensions of the pattern and the geometric relationships.

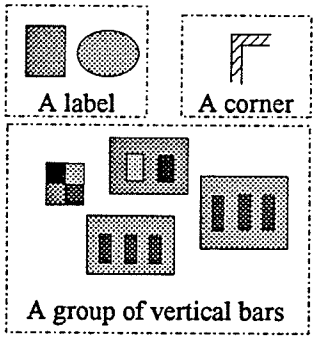


Fig. 7.4 Classification of each target group

Fig. 7.4 shows the possible classification of target groups. The classification can be grouped into at least three categories: machine labels having rectangular and elliptical shapes, corners and vertical bars.

Applying the general three steps to the machine label's recognition, the recognition process could first use a primary identifier horizontally to search primary pattern features such as the 1st and 2nd edges in the image domain, and use a second identifier vertically in the identified region to measure the height of the label. To uniquely recognize a label, a geometric ratio which relates the label width found in the horizontal scan to the height found in vertical scan, could be used to verify the uniqueness of the found pattern. The recognition process of a corner is similar to that for a machine label, except using a different measurement in the third step, i.e., using the relationship between the primary and secondary identifiers.

A similar recognition process to that applied to patterns shown in Fig. 7.2, could be applied to the recognition of vertical bars. The third level of the recognition for this type of target, could need the height information of a pattern to distinguish patterns having the same width but different heights. This height information could be obtained by the line number of the first identifier or by scanning the extent of the first identifier vertically using

a different identifier. A similar ratio as used for a machine label could also be used to verify the target dimension and geometrical characteristics.

Except for the above classification improvement, more user interface features could be added to the software for an effective communication between a user and the software. For ease of access to the desired each type of target group, a pictorial window or dialogue box which displays target icons could be coded into the software. By clicking the mouse on any icon, a user could specify the typical target to be searched. Additionally, the geometric relationship for defining a target could be specified by a pictorial interface. The geometric ratio for the unique recognition of a machine label is a typical example for using this interface type.

7.2 Conclusion

In this thesis, the creation of a vision system has been developed to recognize visual guide marks for an automatic vehicle which is used to move parts in a manufacturing environment. The features of the system and experimental results can be summarized as follows:

1. In general, the vision system can be used for the guidance of an automatic vehicle in manufacturing environments. It could also be used to help position a mobile robot moving parts between machines in a work or machining center.
2. It uses passive vision technique, i.e. a camera without any external lighting. A prior map of the environment specifies the path along which the vehicle is allowed to travel. A recognition procedure installed in a computer, fulfills feature recognition tasks.
3. The target used to verify the performance of the system is a pattern having four 2-in squares set in a 2x2 array. Potential targets could also be among realistic objects such as a switch plate and an electric outlet. The size of a target does not need to be specified.

4. An algorithm for recognizing multiple targets is presented. It extracts the useful feature information from the input object brightness information and matches the obtained edge features to the desired target feature to be identified. This algorithm suits the use in manufacturing environments very well.
5. The significant properties of the system are as follows:
   - The camera optical axis does not need to aim a target center and be perpendicular to the target plane. It can slant horizontally or vertically.
   - The content of a target feature to be recognized, is simply and easily specified by an operator.
   - The system is less affected by the variety of lighting conditions and camera to target distances.
   - It is very easy to interface the software with the environment. This is very convenient for an operator when the working environment's lighting condition has been changed or when a different target has to be introduced.
6. The system has been tested in a real-world environment under various illumination conditions, at different viewing locations including 1) camera is far or close to the target, 2) the images are taken with different horizontal and vertical view angles. The total processing time is 2.4 seconds if the recognition process scans every third line after the system has properly set up. This indicates that the proposed system can be developed to deal with real world environments efficiently and effectively.

The thesis concentrates on the application of a reasonable vision system to real time operation, and on the identification of as many as targets as possible. The target features are a set of edges which have a number of edges having significant brightness contrast. Further work would emphasize 1) the completion of the software to uniquely locate a target, 2) making the system more versatile so that it could recognize a target regardless

of where the illumination around a target is from, 3) improving the accuracy of the target location statement, 4) and making the system more human-like when instantaneously updating the vehicle position is necessary, without stopping the vehicle. All these will enhance the capability of the vehicle control while it is moving in its working area.

References

[1] M. Tajima, "Computer Controlled Wire Guided Vehicle in FMS", *Proc. 1st Int'l Conf. Automatic Guided Vehicle Systems,* June 1981, pp. 1-10.

[2] M. H. E. Larcommbe, "Tracking Stability of Wire Guided Vehicles", *Proc. 1st Int'l Conf. Automatic Guided Vehicle Systems,* June 1981, pp. 137-144.

[3] D. Elbracht and W. Plum, "Wire Guided Flexible Manufacturing Line Systems in Europe, USA and Japan", *Proc. 2nd Int'l Conf. Automatic Guided Vehicle Systems,* 1983, pp. 129-146.

[4] Tsumuta. T, "Survey of Automated Guided Vehicle in Japanese Factory", *Proc. IEEE Int'l Conf. Robotics and Automation,* 1986, pp. 1329-1334.

[5] R. A. Cooke, "Microcomputer Control of Free Ranging Robots", *Proc. 13th Int'l Symp. Industrial Robots,* Vol. 2, April 1983, pp. 209-120.

[6] M. Julliere, L. Marce, and H. Place, "A Guidance System for a Mobile Robot", *Proc. 13th Int'l Symp. Industrial Robots,* Vol. 2, April 1983, pp. 58-68.

[7] J. Albus, E. Kent, M. Nashman, P. Mansbach, and L. Palombo, "Six-dimensional vision system", *Robot Vision,* SPIE Vol. 336, 1982.

[8] R. A. Jarvis, "A Laser Time-of Flight Range Scanner", *IEEE Trans. Pattern Analysis and Machine Intelligence,* Vol. 5, 1983, pp. 505-512.

[9] D. L. Milgram and O. Firschein, "Stereo Digital Image Processing", *Wescon Tech. Papers, Vol. 23,* Sept. 1979.

[10] Stephen T. Barnard and Willim B. Thompson, "Disparity Analysis of Images", *IEEE Trans. Pattern Analysis and Machine Intelligence,* Vol. 2, No. 4, July, 1980, pp. 333-340.

[11] Ikuo Fukui, "TV Image Processing to Determine the Position of a Robot Vehicle", *Pattern Recognition Vol. 14,* 1981, pp. 101-109.

[12] Yubin Hung, Pen-Shu Yeh, David Harwood, "Passive Ranging to Known Planar Point Sets", *Proc. IEEE Int'l Conf. Robotics and Automation*, 1985, pp. 80-85.

[13] Magee, M. J. and Aggarwal, J. K., "Determine the Position of a Robot Using a Single Calibration Object", *Proc. First IEEE Conf. Robotics.* 1984, pp. 140-149.

[14] Tashiro Takeda, Atsuhiko Kato, Toru Suzuki, Mitsuo Hosoi, "Automated Vehicle Guidance Using Spotmark", in *Proc. IEEE Int'l Conf. Robotics and Automation*, 1986, pp. 1349-1353.

[15] Shigeki Ishikawa, Jideki Kuwamoto and Shinji Ozawa, "Visual Navigation of an Autonomous Vehicle Using White Line Recognition", *IEEE Trans. Pattern Analysis and Machine Intelligence*, Vol. 10, No. 5, Sept. 1988, pp. 743-748.

[16] Keith C. Drake, E. S. McVey, and R. M. Inigo, "Sensing Error for a Mobile Robot Using Line Navigation", *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. PAMI-7, no. 4, July 1985, pp. 485-490.

[17] E. S. McVey, Keith C. Drake, and R. M. Inigo, "Range Measurements by a Mobile Robot Using a Navigation Line", *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. PAMI-8, Jan. 1986, pp. 105-109.

[18] O. D. Faugeras and G. Toscani, "The Calibration Problem for Stereo", *Proc. IEEE Computer Vision and Pattern Recognition*, 1986, pp. 15-20.

[19] H. Y. Kwon, Z. Bien, D. K. Cho and I. H. Suh, "An Effective Navigation System Using Vision and Guide-marks for a Mobile Robot", *Proc. of the 11th Triennial World Congress of the Int'l Federation of Automatic Control*, Vol. 5, 1991, pp. 111-116.

[20] Shauna Hui Wang, "Machine Vision with its Application in to Auonomous Vehicle Navigation", *M. Sc. thesis*, The Dep. of Mech. & Ind. Eng., The University of Manitoba, 1993.

[21] Mansur R. Kabuka and Alvaro E. Arenas, "Position Verification of a Mobile Robot Using Standard Pattern", *IEEE Journal of Robotics and Automation,* Vol. RA-3, No.6, Dec. 1987, pp. 505-516.

[22] Toshio Fukuda, Shigenori ITO, Nobuyuki OOTA, Fumihito ARAI, Yasunori ABE, Kouetsu TANAKA and Yoshio TANAKA, "Navigation System Based on Ceiling Landmark Recognition for Autonomous Mobile Robot", *Proc. IEEE IECON*, Vol. 2, 1993, pp. 1466-1471.

[23] Saburo Tsuji and Jiang Yu Zheng, "Visual Path Planning by a Mobile Robot", *Proc. of 10th IJCAI*, 1987, pp. 1127-1130.

[24] K. Storjohann, Th. Zielke, H. A. Mallot and W. von Seelen, "Visual Obstacle Detection for Automatically Guided Vehicles", *Proc. IEEE Int. Conf. Robotics and Automation*, vol. 2, 1990, pp. 761-767.

[25] Nicholas Ayache and Francis Lustman, "Trinocular Stereo Vision for Robotics", *IEEE Trans. Pattern Analysis and Machine Intelligence*, Vol. 13, No. 1, Jan. 1991, pp. 73-84.

[26] Zhengyou Zhang and Olivier Faugeras, "A 3D World Model Builder with a Mobile Robot", *The International Journal of Robotics Research*, Vol. 11, No. 4, Aug. 1992, pp. 269-284.

[27] Allen M. Waxman, Jacqueline J. LeMoigne, Larry S. Davis, Babu Srinivasan, Todd R. Kushner, Eli Liang and Tharakesh Siddalingaiah,"A Visual Navigation System for Autonomous Land Vehicles", *IEEE Journal of Robotics and Automation.* Vol. RA-3, No. 2, April, 1987.

[28] Darwin Kuan, Gary Phipps and A. Chuan Hsueh, "Autonomous Robotic Vehicle Road Following", *IEEE Transaction Pattern Analysis and Machine Intelligence.* Vol. 10, No. 5, Sep. 1988.

[29] Yuji Mesaki and Isao Masuda, "A New Mobile Robot Guidance System Using Optical Reflectors", *Proc. IEEE Int'l Conf. Intelligent Robots and Systems,* July, 1992.

[30] Takeshi Hashimoto, Shigehiro Yamamoto, Takehiko Aso and Minoru Abe, "Vision-based Vehicle Navigation Using the Fluorescent Lamp Array on the Ceiling", *Mem. Fac. Eng., Kyoto Univ.*, Vol. 55, No.1, Jan. 1993, pp. 37-48.

[31] Hiroshi Ishiguro, Massashi Yamamoto and Saburo Tsuji, "Omni-Directional Stereo", *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 14, No. 2, Feb. 1992.

[32] Hae-Seok Oh, Chil-Woo Lee and Ishizuka Mitsuru, "Navigation Control of a Mobile Robot Based on Active Vision", *Proc. IEEE IECON'91*, pp. 1122-26.

[33] Alberto Elfes, "Sonar-Based Real-World Mapping and Navigation", *IEEE Journal of Robotics and Automation*, Vol. RA-3, No. 3, June 1987, pp. 249-265.

[34] A. Curran and K. J. Kyriakopoulos, "Sensor-Based Self-Localization for Wheeled Mobile Robots", *IEEE Proc. Int'l Conf. on Robotics and Automation*, Vol. 1, 1993, pp. 8-13.

[35] Ingemar J. Cox, "Blanche--An Experiment in Guidance and Navigation of an Autonomous Robot Vehicle", *IEEE Trans. Robotics and Automation*, Vol. 7, No. 2, April 1991.

[36] Wookhyun Kim, Yuzo Hirai, Takashi Furukawa and Hidenobu Arita, "A neural Network Extracting Road Segments from Maps using Necessary and Sufficient Features", *International Joint Conference on Neural Networks*, vol. 1, pp 665-670, 1991.

[37] X. Yang, R. A. Athale and N. P. Caviris, "Prototype Optical Neural Network System for Automatic Target Recognition", *Journal of Optical Memory and Neural Networks*, Vol. 2, Iss. 1, 1993, pp. 37-50.

[38] C. Ming, Wang, "Location Estimation and Uncertainty Analysis for Mobile Robot", *Proc. IEEE Int'l Conf. Robotics and Automation*, Vol. 2, 1987.

[39] A. M. Waxman, J. LeMoigne and B. Srinivasan, "Visual Navigation of Roadway", *Proc. IEEE Int'l Conf. Robotics and Automation*, Mar. 1985, pp862-867.

[40] I. Aloimonos, I. Weiss and A. Bandyopadhyay, "Active Vision", *Proc. 1st Int'l Conf. Computer Vision*, 1987.

[41] S. K. Premi and C. B. Besant, "A review of various guidance techniques that can be used by mobile robots or AGVs", *Proc. of 2'nd Inte'l Conf. Automatic Guided Vehicle Systems*, pp. 195-210, Junn, 1983.

[42] T. Arai, "Development of the measuring equipment for location and direction using ultrasonic waves and its evaluation", *Trans. SICE*, vol.18, No. 10 pp.1013-1020, Oct. 1982.

[43] P. Y. Mundkur and U. B. Desai, "Automatic Target REcognition using Image and Network Decomposition", *International Joint Conference on Neural Networks*, vol.1, pp. 281-286, 1991.

[44] Toshihiro Tsumura, "Survey of Automated Guided Vehicle in Japanese Factory", *IEEE 1986 International Conference on Robotics and Automation.*

Appendix A

Determination of the Vehicle Location

To determine the position and orientation of the vehicle, three possible methods could be theoretically used. The first method uses the orientations of three marks relative to the vehicle. The second method is as mentioned in section 3.2.1; it derives the orientations of two marks with respect to the vehicle, and the distance between one of the marks and the vehicle. The third method derives the distances of each of two marks with respect to the vehicle to determine the vehicle location.

In this section, the first method will be briefly discussed below as it is the most frequently used method for our application. The second and third methods deal with the distance of a mark and the vehicle. The distance measurement requires that the width or height of the target in the image plane be accurately measured. This is not possible for many of the target types.

For the first method, assuming that three targets appear in an image. Fig. A.1 shows the top view of three targets and the vehicle. The location of the vehicle is determined by three variables: x, y and horizontal deviation angle α if assuming that the floor on which the vehicle travels is even.

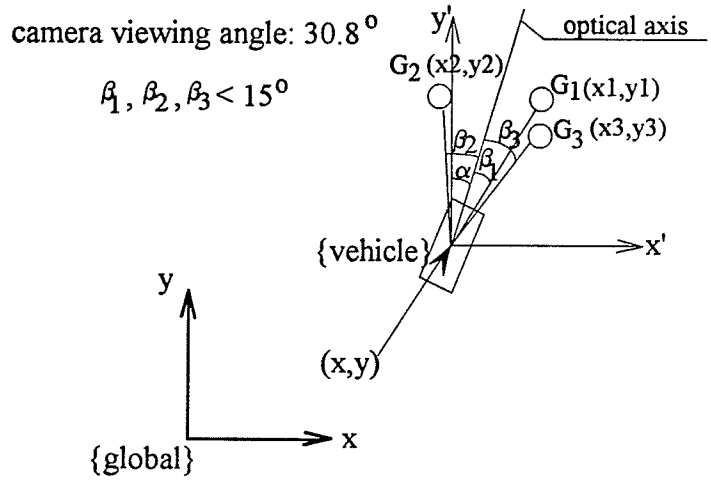


Fig. A.1 Top View of Targets and Vehicle

In Fig. A.1, β_1, β_2 and β_3 are horizontal deviation angles of the guide marks relative to the vehicle traveling direction respectively in a horizontal plane which is parallel to the floor plane. They can be derived by using the available information after image processing.

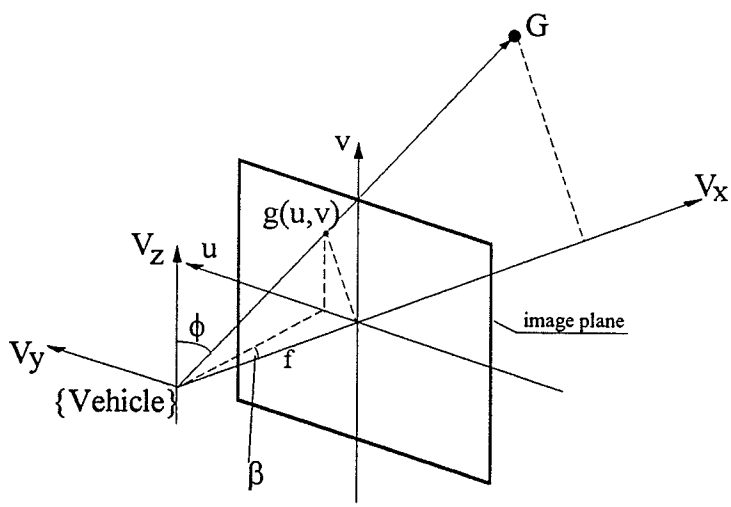


Fig. A.2 Mapping of a guide mark on image plane

Fig. A.2 shows a guide mark mapping onto the camera image plane. Wherein f implies the focal length of the camera lens, u and v are coordinates of the guide mark

center in the image plane. The horizontal deviation angle β of the guide mark can be determined using equation (A-1).

$$\beta = \tan^{-1}\left(\frac{u}{f}\right) \qquad \text{(A-1)}$$

The geometrical relationships between these marks and the vehicle are given as:

$$\frac{x_1 - x}{y_1 - y} = \tan(\beta_1 + \alpha) \qquad \text{(A-2)}$$

$$\frac{x - x_2}{y_2 - y} = \tan(\beta_2 - \alpha) \qquad \text{(A-3)}$$

$$\frac{x_3 - x}{y_3 - y} = \tan(\beta_3 + \alpha) \qquad \text{(A-4)}$$

With obtained β_1, β_2 and β_3 from equation (A-1) and known absolute coordinates of three marks, the variables x, y and α can be derived. The general solution for solving the variables x, y and α, is not described here.

The described method is based on the assumption that three marks are in the same scene. However, an image of a scene may contain one or two marks in some cases. It is necessary to develop a program which could also derive the vehicle's position from several consecutive images which are taken in very short interval and may contain just one or two targets. This could save the vehicle wandering time during the vehicle running.

Appendix B

References Reviewed in [20]

From M.Sc. thesis by S. H. Wang, 1993, The University of Manitoba

[1] Andresen, L. Davis, R.D. Eastman and S. Kambbampati, "Visual algorithm for autonomous navigation", *Proc. IEEE Int'l Conf. on Robotics and Automation*, 1985, pp. 656-661.

[2] Arkin and R. Murphy, "Autonomous Navigation in a manufacturing environment", *IEEE Trans. on Robotics and Automation*, Vol. 6, No. 5, 1990, pp. 445-454.

[3] Augusteijn and C. Dyer, "Recognition and recovery of the three dimensional orientation of planar point patterns", *Computer Vision, Graphics, and Image Processing*, Vol. 36, 1986, pp. 76-99.

[4] T. Barnard, "Interpreting perspective images", *Artificial Intelligence 21*, 1983, pp. 435-462.

[5] Brzakovic and L. Hong, "Road edge detection for mobile robot navigation", *Proc. IEEE Int'l Conf. on Robotics and Automation*, 1989, pp. 1153-.

[6] Courtney and J. K. Aggarwal, "Robot guidance using computer vision", *Pattern Recognition*, Vol. 17, No. 6, 1984, pp. 585-592.

[7] Dhome, M. Richetin, J. Lapreste and G. Rives, "Determination of the attitude of 3D objects from a single perspective view", *IEEE Transaction on Pattern analysis and machine intelligence*, Vol. 11, No. 12, 1989, pp. 1265-1278.

[8] Fischler and R. Bolles, "Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography", *Communications of ACM*, Vol. 24, 1981, pp. 381-395.

[9] Frohn and W. Seelen, "VISOCAR: An autonomous industrial transport vehicle guided by visual navigation", *Proc. IEEE Int'l Conf. on Robotics and Automation*, 1989, pp. 1155-1159.

[10] Goto and A. Stentz, "The CMU system for mobile robot navigation", *Proc. IEEE Int'l Conf. on Robotics and Automation*, 1987, pp. 99-105.

[11] Fukui, "TV image processing to determine the position of a robot vehicle", *Pattern Recognition*, Vol. 15, 1981, pp. 101-109.

[12] Horaud, "New methods for matching 3D objects with single perspective views", *IEEE Transaction on Pattern analysis and machine intelligence*, Vol. Pami-9, No. 3, 1987.

[13] Horaud, B. Conio, O. leboulleux and B. Lacolle, "An analytic solution for the perspective 4-point problem", *Computer Vision, Graphics, and Image Processing*, Vol. 47, 1987, pp. 33-44.

[14] Krotkov, "Mobile robot localization using a single image", *Proc. IEEE Int'l Conf. on Robotics and Automation*, 1989, pp. 978-983.

[15] D. Livine, "Computer determination of depth maps", *Computer Vision, Graphics, and Image Processing*, Vol. 2, 1973, pp. 131-150.

[16] J. Magee, "Determination the position of a robot using a single calibration object", *Proc. IEEE Int'l Conf. on Robotics and Automation*, 1984, pp. 140-149.

[17] Onofuchi and M. Watanabe, "A visual navigation system using a multi-information local map", *Proc. IEEE Int'l Conf. on Robotics and Automation*, 1990, pp. 767-775.

[18] Poggio, "Early vision: From computational structure to algorithm and parallel hardware", *Computer Vision, Graphics, and Image Processing*, Vol. 31, pp. 139-155.

[19] Sarachik, "Characterizing an indoor environment with a mobile robot and uncalibrated stereo", *Proc. IEEE Int'l Conf. on Robotics and Automation*, 1989, pp. 985-989.

[20] Sugihara, "Some location problem for robot navigation using a single camera", *Computer Vision, Graphics, and Image Processing*, Vol. 42, 1988, pp. 112-129.

[21] Turk and D. Morgenthaler, "Video road-following for the autonomous land vehicle", *Proc. IEEE Int'l Conf. on Robotics and Automation*, 1987, pp. 273-280.

[22] Triendl and David J. Kriegman, "Stereo vision and navigation within buildings", *Proc. IEEE Int'l Conf. on Robotics and Automation*, 1987, pp. 1725-1730.

[23] Tsuji, J. Zheng and M. Asada, "Stereo vision of a mobile robot: world constraints for image matching and interpretation", *Proc. IEEE Int'l Conf. on Robotics and Automation*, 1986, pp. 1594-1599.

[24] Wallace, K. Matsuzaki, Y. Goto, J. Crisman, J. Webb and T. Kanade, "Progress in robot road Following", *Proc. IEEE Int'l Conf. on Robotics and Automation*, 1986, pp. 1615-1621.

[25] Waxman, J. LeMoigne and B. Srinivasan, "Visual navigation of roadways", *Proc. IEEE Int'l Conf. on Robotics and Automation*, 1985, pp. 662-667.

[26] Waxman, J. LeMoigne, L. Davis, E. Liang and T. Siddalingaiah, "A visual navigation system", *Proc. IEEE Int'l Conf. on Robotics and Automation*, 1986, pp. 1600-1606.

Appendix C

Gray Level Table for converting input image

Input Visual Data is Converted into Logarithmic Data

read	convert	read	convert
255	255	234	229
254	253	233	228
253	252	232	228
252	251	231	227
251	249	230	226
250	248	229	226
249	247	228	225
248	245	227	225
247	244	226	224
246	242	225	223
245	241	224	223
244	239	223	222
243	238	222	221
242	236	221	221
241	234	220	220
240	232	219	219
239	232	218	218
238	231	217	218
237	231	216	217
236	230	215	216
235	229	214	216

read	convert	read	convert
212	214	185	196
211	213	184	195
210	212	183	195
209	212	182	194
208	211	181	193
207	210	180	193
206	210	179	192
205	209	178	191
204	208	177	190
203	208	176	189
202	207	175	189
201	207	174	188
200	206	173	187
199	205	172	186
198	205	171	186
197	204	170	185
196	204	169	184
195	203	168	184
194	202	167	183
193	202	166	182
192	201	165	181
191	200	164	180
190	200	163	180
189	199	162	179
188	198	161	178
187	198	160	177

read	convert	read	convert
158	176	131	152
157	175	130	151
156	174	129	150
155	173	128	149
154	172	127	148
153	171	126	147
152	170	125	146
151	169	124	145
150	168	123	144
149	168	122	143
148	167	121	142
147	166	120	142
146	165	119	141
145	164	118	140
144	163	117	139
143	162	116	138
142	162	115	137
141	161	114	136
140	160	113	135
139	159	112	135
138	158	111	134
137	158	110	133
136	157	109	132
135	156	108	131
134	155	107	130
133	154	106	129

read	convert	read	convert
104	127	77	103
103	125	76	103
102	124	75	102
101	123	74	101
100	122	73	100
99	121	72	99
98	120	71	98
97	120	70	97
96	119	69	96
95	118	68	96
94	117	67	95
93	117	66	94
92	116	65	93
91	115	64	92
90	114	63	91
89	114	62	90
88	113	61	89
87	112	60	89
86	111	59	88
85	110	58	87
84	110	57	86
83	109	56	85
82	108	55	84
81	107	54	83
80	106	53	82
79	105	52	81

read	convert	read	convert
50	79	23	45
49	78	22	43
48	76	21	41
47	76	20	39
46	75	19	37
45	74	18	34
44	73	17	32
43	72	16	29
42	71	15	26
41	69	14	23
40	68	13	20
39	67	12	17
38	66	11	13
37	65	10	10
36	64	9	9
35	63	8	8
34	62	7	7
33	61	6	6
32	59	5	5
31	58	4	4
30	57	3	3
29	55	2	2
28	54	1	1
27	52	0	0
26	51		
25	49		

Appendix D

Supplementary Images

This section gives several supplementary images to illustrate the performance of the vision system.

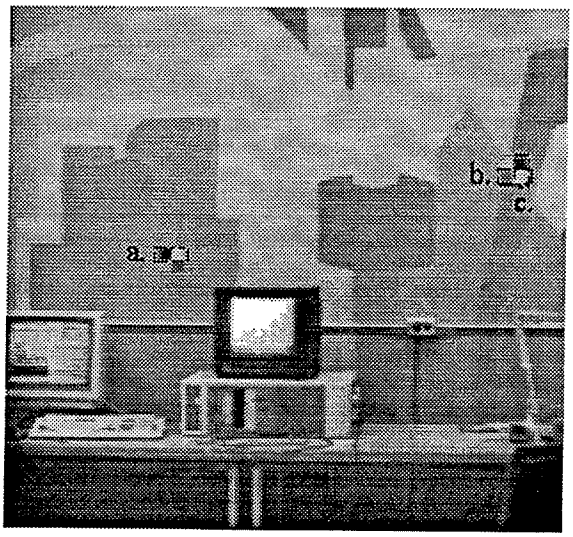


Fig. 6.2

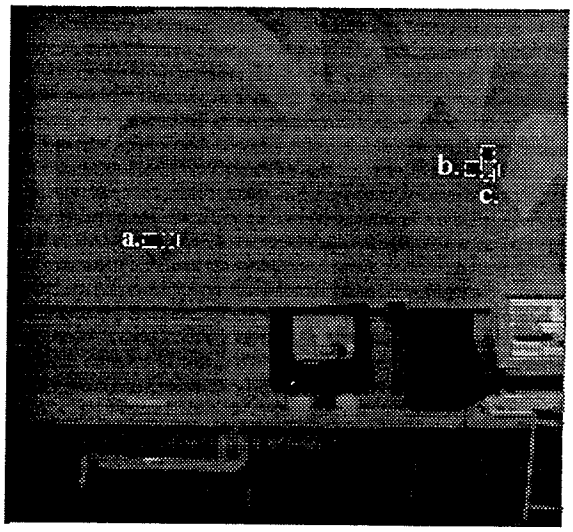


Fig. D.1
(Fig. 6.2 with a lower level of brightness)

Fig. D.2(Fig. 6.4 taken at 3.5m)

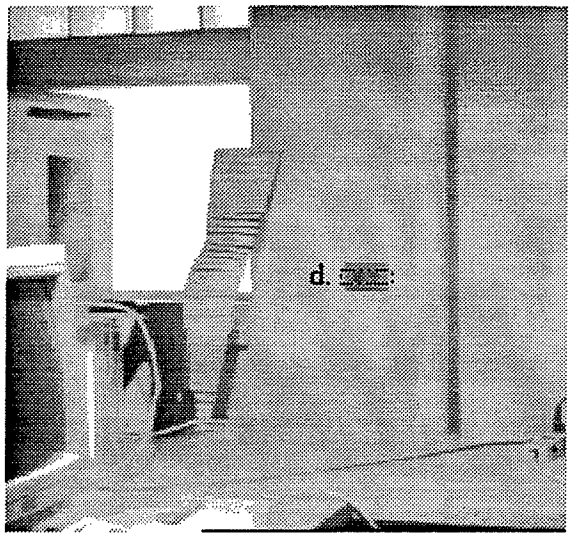


Fig. D.3

(Fig. D.2 with a higher level of brightness)

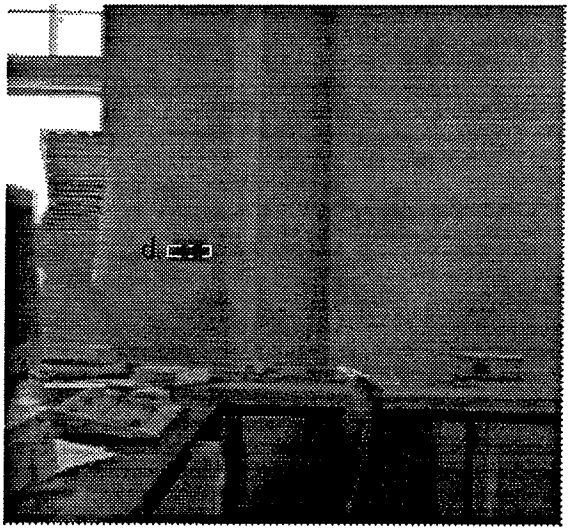


Fig. 6.4(taken at 5.5 m)

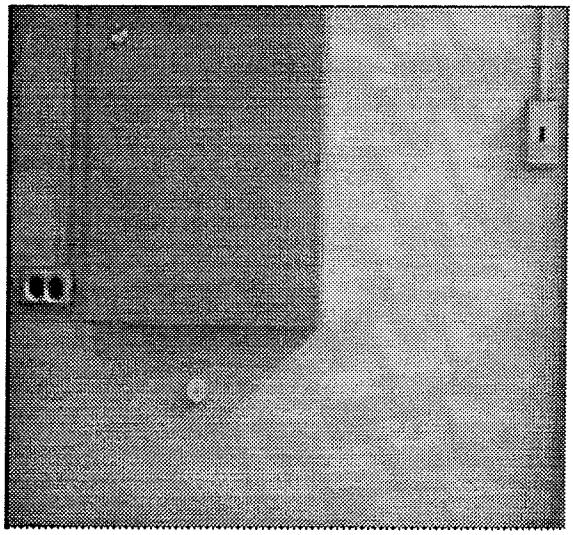


Fig. D.4

(light incident effect to a switch plate, with light coming from the right-top of the image viewer)

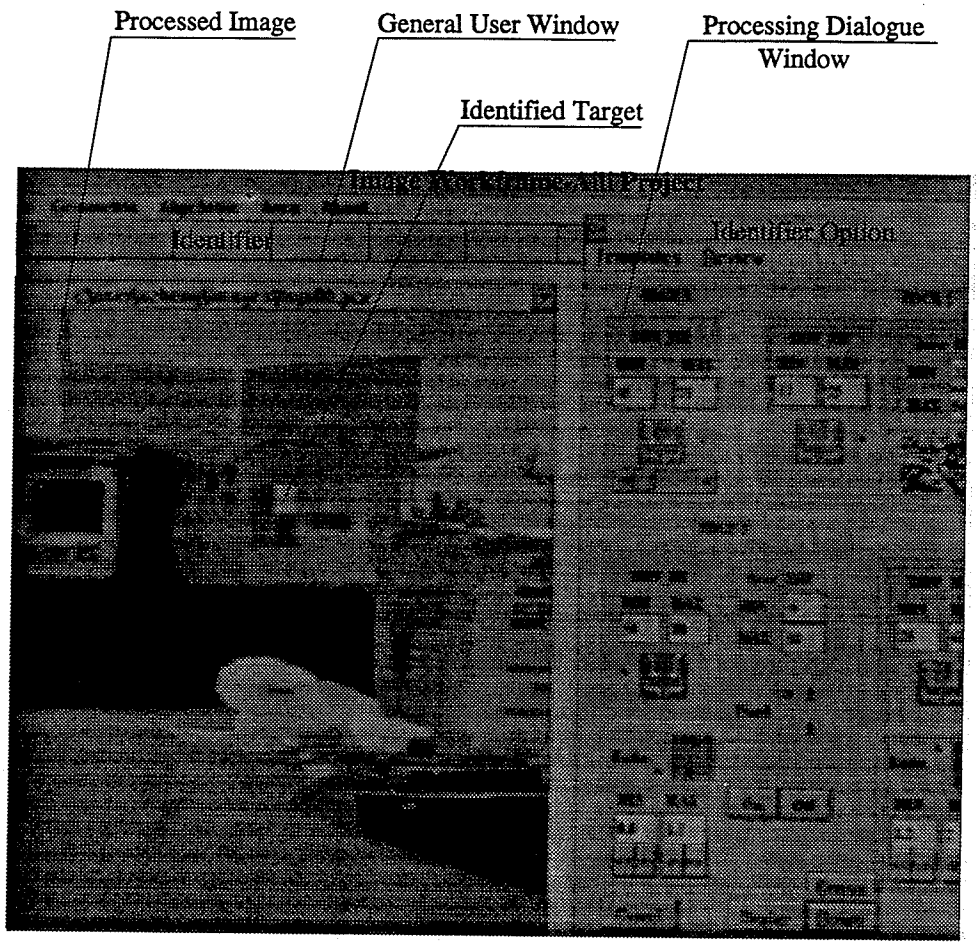

Processed Image
General User Window
Processing Dialogue Window
Identified Target
Identifier Option

Fig.5.5