12-1-2013

Computer-Based Stereoscopic Parts Recognition for Robotic Applications

Ahmad A. Fayed
University of Nevada, Las Vegas, fayeda@unlv.nevada.edu

COMPUTER-BASED STEREOSCOPIC PARTS RECOGNITION FOR ROBOTIC APPLICATIONS

By

Ahmad Fayed

Bachelor of Science in Mechanical Engineering
Al-Azhar University, Cairo, Egypt
1997

Master of Science in Mechanical Engineering
King Saud University, Riyadh, KSA
2007

A dissertation submitted in partial fulfillment
of the requirements for the

Doctor of Philosophy - Mechanical Engineering

Department of Mechanical Engineering
Howard R. Hughes College of Engineering
The Graduate College

University of Nevada, Las Vegas

December 2013

THE GRADUATE COLLEGE

We recommend the dissertation prepared under our supervision by

Ahmad Fayed

entitled

Computer-Based Stereoscopic Parts Recognition for Robotic Applications

is approved in partial fulfillment of the requirements for the degree of

Doctor of Philosophy in Engineering - Mechanical Engineering

Department of Mechanical Engineering

Georg Mauer, Ph.D., Committee Chair

Woosoon Yim, Ph.D., Committee Member

Brendan O'Toole, Ph.D., Committee Member

Evangelos Yfantis, Ph.D., Committee Member

Moses Karakouzian, Ph.D., Graduate College Representative

Kathryn Hausbeck Korgan, Ph.D., Interim Dean of the Graduate College

December 2013

ABSTRACT

COMPUTER-BASED STEREOSCOPIC PARTS RECOGNITION

FOR ROBOTIC APPLICATIONS

By

Ahmad Fayed

Dr Georg Mauer, Examination Committee Chair
Professor of Mechanical Engineering
University of Nevada Las Vegas

Most of robotic handling and assembly operations are based on sensors such as range and touch sensors. In certain circumstances, such as in the presence of ionizing radiation where most customary sensors will degrade over time due to radiation exposure, these sensors won't function properly. Utilizing two or more cameras (stereo vision) located outside the target zone and analyzing their images to identify location and dimensions of parts within the robot workspace is an alternative for using sensors. Object Recognition is affected by the light condition which oftentimes causes the gray-scale or red, green, and blue values to have a relatively small dynamic range. With this small dynamic range, edge detection algorithms fail to detect the proper edges and therefore cause improper image segmentation. To tackle this problem, a transformation on the (r,g,b) values of the pixels is introduced and applied prior to the edge detection and

segmentation process. A stereoscopic computer vision system with multiple cameras is then used to compute the distance of the object from the origin of a global Euclidean coordinate system with high resolution. As an application of computer vision, a classifier for testing remote solar panels for cleanness condition, and performing cleaning when necessary, is introduced. A classification algorithm consisting of: the classification vector, the metric used, the training of the classifier, the testing of the classifier, and the classifier is put into play for everyday use. A smart cleaning robot is being designed based on this system to perform the cleaning autonomously when necessary. Another application of computer vision is inspecting the degree of air pollution. A real time classification algorithm that uses a quantization algorithm based on prior calibration is applied to evaluate the quality of air. The intelligent system, based on this algorithm, classifies the air using a numeric system from 1 to 10 which is then transformed to a qualitative scale.

ACKNOWLEDGMENTS

First and foremost, all praise is due to Allah, the only God, the Compassionate, the Almighty Merciful, who kindly helped me throughout my life and to complete this work.

I would like to thank my examination committee chair, Dr. Georg Mauer, for introducing me to the field of computer vision.

I would like to express my sincere and deep sense of gratitude to my committee member, Dr. Evangelos Yfantis, for his sustained enthusiasm, motivation, exemplary guidance throughout the course of my doctoral research and for the time and efforts he spent to help me build a great background in the computer vision field. .

I would like to thank Dr. R. F. Boehm for the generous access he has provided to the solar panels at his UNLV lab, and his continuing support to complete the "Dust Detection in Solar Panels" part of this research. Many thanks to the NSF for their support and funding of the "Dust Detection in Solar Panels" and the "Pollution Detection" parts of this research.

I also would like to extend my gratitude to all of my family members who are always a great support for me.

TABLE OF CONTENTS

ABSTRACT iii

ACKNOWLEDGMENTS v

TABLE OF CONTENTS vi

LIST OF FIGURES ix

CHAPTER 1 INTRODUCTION 1

I. Stereoscopic Computer Vision 1

II. Computer Vision Applications 2

III. Research Objectives 3

IV. Contribution and Publications 4

V. Dissertation Outline 5

CHAPTER 2 BACKGROUND INFORMATION 7

I. Research Topic 7

II. Literature Survey 8

III. Camera Model 12

IV. Camera Calibration 15

V. Challenges for Image Segmentation and Object Recognition 17

A. Shadow Detection and Elimination 17

B. Edge Detection 19

C. Line Detection 21

CHAPTER 3 PRELIMINARY IMAGE PROCESSING TASKS 24

Abstract 24

I. Camera Calibration Procedure: 24

II. Testing Images 28

III. Image preparation 30

IV. Edge Detection after Eliminating Shadow Effects using Image Shifting 30

V. Line Reconstruction using Hough transform 35
VI. Corner extraction and Line joining 36
A. Extracting Corners from incomplete line segments 37
B. Joining fragmented line segments 40
VII. Conclusion: 45

CHAPTER 4 IMAGE SEGMENTATION, OBJECT DETECTION, AND DISTANCE ESTIMATION 46
Abstract 46
I. Introduction 47
II. Multi Camera Vision System with Two-Mode Non Invasive Laser 50
III. Applications 69
IV. Conclusions 71

CHAPTER 5 A CAMERA SYSTEM FOR DETECTING DUST AND OTHER DEPOSITS ON SOLAR PANELS 72
Abstract 72
I. Introduction 73
II. Background Information 78
A. The color of the Sky 78
B. Light and Electric Energy 80
III. Classification Algorithm 82
IV. Conclusion 92

CHAPTER 6 POLLUTION DETECTION IN URBAN AREAS USING THE EXISTING CAMERA NETWORKS 94
Abstract 94
I. Introduction 95
II. Background information 98
A. The color of the Sky 98
B. Light and Air Pollution 100
III. Classification Algorithm 101
IV. Conclusion 112

CHAPTER 7 CONCLUSION AND FUTURE WORK 113
Future Work 116

BIBLIOGRAPHY 117

VITA 124

LIST OF FIGURES

Figure 2.1 The pinhole camera model with focal length f and principal point (u_o , v_o) 13

Figure 2.2 Pinhole Camera Model – xz Plane 13

Figure 2.3 Mapping a straight line into parametric spaces 22

Figure 3.1 A set of 15 checkerboard pictures taken to perform camera calibration 25

Figure 3.2 Marking the four extreme corners of the checkerboard image, pic01.jpg 26

Figure 3.3 Resulting boundaries of the calibration grid on pic01.jpg 26

Figure 3.4 Extracted corners for pic01.jpg 26

Figure 3.5 Camera extrinsic parameters with Camera centered 27

Figure 3.6 Sample testing images for prismatic and cylindrical objects 29

Figure 3.7 Image of the prismatic object, prism01.jpg 31

Figure 3.8 Results of Image cropping and Canny edge detection, prism01.jpg ... 31

Figure 3.9 Image of the prismatic object, Prism08.jpg 31

Figure 3.10 Results of Canny edge detection, Prism08.jpg 32

Figure 3.11 Image with applying histogram equalization and shifting techniques 33

Figure 3.12 Histogram of image Prism08.jpg with equalization and shifting techniques 34

Figure 3.13 Edge maps of image Prism08.jpg with equalization and shifting techniques 35

Figure 3.14 Extracted lines (6 segments in 5 sides), prism01.jpg 36

Figure 3.15 MATLAB function for determining the pairs of intersecting lines 38

Figure 3.16 MATLAB function for determining intersection point of extracted lines 39

Figure 3.17 Extracted line segments, using Hough transform, Prism06.jpg 40

Figure 3.18 Two fragments of a line segment 41

Figure 3.19 Original and equalized-shifted images, Prism03.jpg 42
Figure 3.20 Detected edges and extracted lines, Prism03.jpg 42
Figure 3.21 Extracted lines with coordinates, Prism03.jpg 43
Figure 3.22 Extracted lines after joining two collinear fragments, Prism03.jpg ... 43
Figure 3.23 MATLAB function for determining intersection point of extracted lines 44
Figure 4.1 Histogram of a prismatic object showing a small dynamic range, form 103 to 170. 52
Figure 4.2 Perspective projection 54
Figure 4.3 Cross configuration of the vision system 56
Figure 4.4 Parallel arrangement of the four cameras 57
Figure 4.5 Perspective projection of camera 1 and camera 2 59
Figure 4.6 Perspective projection of camera 3 and camera 4 61
Figure 4.7 Perspective projection of camera 1 and camera 3 62
Figure 4.8 Perspective projection of camera 2 and camera 3 63
Figure 4.9 Perspective projection of camera 2 and camera 4 64
Figure 4.10 Perspective projection of camera 1 and camera 4 66
Figure 5.1 Image of a clean panel 84
Figure 5.2 Image of a dusty panel 85
Figure 5.3 Probability density function for the R, G, B channels, Observe the shift of the red, green, and blue modes 85
Figure 5.4 Individual probability density functions for R, G, B channels of both clean and dusty images 86
Figure 6.1 Image of a very clean sky 102
Figure 6.2 Image of a clean sky 102
Figure 6.3 Probability density function for the R, G, B channels, Observe the shift of the red, green, and blue modes 103
Figure 6.4 Individual probability density functions for R, G, B channels of both the very clean and the clean sky images 104

CHAPTER 1

INTRODUCTION

I. Stereoscopic Computer Vision

Robotic Parts Handling and Assembly is a fairly well researched field, but little of the work has used vision as the primary sensor. Most of Handling and Assembly process are being done using a fixed environment setup, pre-programmed manipulators, manual and remote control. Most of robotic handling and assembly operations are based on sensors such as range and touch sensors. In certain circumstances, such as in the presence of ionizing radiation where most customary sensors will degrade over time due to radiation exposure, these sensors won't function properly [1]. Utilizing two cameras (stereo vision) located outside the radiation zone and analyzing their images to identify location and dimensions of cylindrical parts within the robot workspace is proven to be reliable in a hot cell [2]. Object Recognition is affected by the light condition which often times causes the dynamic range of the gray-scale or red, green, and blue values to have a relatively small dynamic range. With this small dynamic range, edge detection algorithms fail to detect the proper edges and therefore cause improper image segmentation. To tackle this problem, a transformation on

the (r,g,b) values of the pixels is introduced and applied prior to the edge detection and segmentation process. Multiple cameras are then used to compute the distance of the object from the origin of a global Euclidean coordinate system with high resolution. The introduced algorithm involves camera calibration, triangulation, and least square and is faster and more accurate than the epipolar method.

II. Computer Vision Applications

As an application of computer vision, a classifier for testing remote solar panels for cleanness condition, and performing cleaning when necessary, is introduced. The classification algorithm consisting of: the classification vector, the metric used, the training of the classifier, the testing of the classifier, and the classifier is put into play for everyday use. A smart cleaning robot is being designed based on this system to perform the cleaning autonomously when necessary. Another application of computer vision is inspecting the degree of air pollution. A real time classification algorithm that uses a quantization algorithm based on prior calibration is applied to evaluate the quality of air. Currently, commercial cameras transmitting MPEG-4 or H.264 frames connected to the circuit board where the classification takes place are used. The algorithm is loaded from an in-board flash memory into an Advanced RISC Machine (ARM)

processor on the board, and operates on incoming data. The intelligent system, based on the proposed algorithm, classifies the air using a numeric system from 1 to 10 which is then transformed to a qualitative scale.

III. Research Objectives

The objective of this project is to:

1) Develop the theory, which allows for reliable and accurate recognition and localization of objects in an interior environment regardless of the light condition, object color, and shadow existence. The theory is based on a system of 4 identical cameras, with the same orientation and parallel lenses, in a cross configuration along with non-invasive laser in the middle. The laser projects a tiny point on the object that can easily be identified in each of the camera image spaces, and creates an undisputed common point. The theory will mathematically formulate the relations between the 2-D information in each image plane and the actual 3-D position and compute the minimum geometrical description of the target object that allows for reliable robotic handling.

2) Introduce classification algorithms that can be used in camera systems to detect certain conditions of the target objects and hence trigger a smart robot to perform certain tasks. Two classification algorithms are introduced: the first allows camera system to detect dust and other deposits on remote solar panels,

and the second utilizes existing camera networks to detect air quality and possible pollution.

IV. Contribution and Publications

Following are the main contributions and publications associated with this dissertation:

1- The development of the theory of using 4 cameras in a cross configuration along with a non-invasive laser pointer in the center of the cameras that is used for accurate camera calibration, image segmentation, object detection and distance estimation. The following paper (IJARS-S-13-00211) was submitted for publication in the International Journal of Advanced Robotic Systems:

A. Fayed, Image Segmentation, Object Detection, Robot Vision, and Distance Estimation, International Journal of Advanced Robotic Systems, IJARS-D-13-00070, SUBMITTED July 19, 2013.

2- The development of a classification algorithm that uses a camera system for detecting dust and other deposits on solar panels. A smart robot is being developed to carry out the cleaning process based on the output of the aforementioned classification algorithm. The following

paper (ICDSC13-paper36) was submitted to the Seventh ACM/IEEE International Conference on Distributed Smart Cameras:

A. Fayed, A Camera System For Detecting Dust And Other Deposits On Solar Panels, 7th ACM/IEEE International Conference on Distributed Smart Cameras (ICDSC13), Oct 29 – Nov 1, 2013, Palm Springs, CA, SUBMITTED.

3- The development of a classification algorithm that uses a camera system for detecting air quality and pollution in urban areas using the existing camera networks. The following paper (IJMT10157) was accepted for publication in the International Journal of Multimedia Technology and is currently IN PRESS.

A. Fayed, Pollution Detection in Urban Areas Using the Existing Camera Networks, International Journal of Multimedia Technology (IJMT), IN PRESS.

V. Dissertation Outline

The outline of this dissertation is as follows. Chapter 1 is an introduction. Chapter 2 is background information. Chapter 3 is about the challenges in edge detection and the proposed techniques to overcome them. In chapter 4 the theory of the image segmentation, object detection, robot vision, and distance estimation is developed. Chapter 5 describes the camera system for detecting dust and other deposits on solar panels. Chapter 6 gives the details about the proposed

pollution detection in urban areas using the existing camera networks. Chapter 7 is the conclusions.

CHAPTER 2

BACKGROUND INFORMATION

I. Research Topic

The use of Robotics in material handling and assembly is one of the numerous applications of robotics in the industry. Robotics used to be programmed to do specific tasks under supervision of human and then developed to do these programmed tasks with certain kind of safety sensors that allow them to fail safe. With the developments of sensing devices such as sonar, infra-red and laser range sensors, it was possible to have an adaptive robotics handling and manipulations with a very minor or absolutely no human interference. With a combination of these sensors and controlling software the autonomous robotics came to existence.

Under certain circumstances such as exposure to ionized radiation, the operational performance of electronic devices including sensors will degrade, [1]. This can take place mainly in space environment and nuclear applications. For this reason, researches on utilizing cameras instead of sensors are gaining more importance. Besides, the information content of camera image (e.g. shape, structure, and color) is considerably greater than that of any other sensors whenever it is retrievable. However, retrieving this content is not a single

problem but a large number of complicated problems/processes These processes including, but not limited to, segmentation, object recognition, 3D reconstruction, tracking, fixation, feedback, [3]. Additionally, in contrast to laser scanners, cameras do not require moving objects and are thus robust and wear-free, [4]. Another advantage of using cameras over using sensors is reducing cost and energy needed in the operation. This will be of a great value if accompanied by an optimized code that will finish the image processing task in a significantly small time.

The scope of this research and introduced theory would include developing a stereo vision system for the purpose of identification and localization of objects with different geometry in different environments.

II. Literature Survey

Nowadays, robots are performing several general purpose tasks including but not limited to food packaging, car production and assembly, mass productions of Printed Circuit Boards (PCBs), health care providing and performing guided surgery. Robots are also utilized in navigation and autonomous applications such as Automated Guided Vehicles (AGVs), [5]. Robots are generally utilized to perform a repeated tasks more accurate and faster than human can do which make them more cost and time efficient than

any other alternatives. Most importantly, robots are utilized to perform tasks that humans would not feel comfort doing them, tasks in which humans life will be in danger when performing them, tasks which are taking place in a risky or hazardous environment, or tasks in environments that are physically inaccessible by humans. These tasks include but not limited to Space exploration, rescue operation, handling and cleaning hazardous materials such as explosives or nuclear materials, or sometimes in the battle field to detonate a buried improvised explosives [6]. When it comes to independency and autonomy, robots can either be programmable where they follow a predetermined sequence of movements, remotely controlled by human operator, or, most recently, independently full autonomous.

Originally, robot control and guidance fields were considered part of the Automatic Control and Artificial Intelligence researches since they were meant to perform fixed or repeated tasks. Later they were equipped with sensors to move them more towards the autonomy. With the advance in research in the area of Computer Vision and Stereo Vision, a full autonomous robotic manipulation becomes more viable.

Vision is one of the most beneficial processes when applied to machines and robotics despite its complication and its need for a huge computing power. Continuous researches in the nature of Human Visual System (HVS) and other

biological vision systems contributed to the huge development in the computer vision field. Work in Computer Vision field is greatly inspired by the biological vision of humans, animals such as eagle and fish and even insects such as housefly [7, 8, 9, 10]. Human was thought to have the 3D perception of scenes because of having two eyes but this is not true because humans still can have the 3D perception with single eye vision. It is the nature of human eye retina along with brain that can interpret the scene with its 3D content. Any object can cast infinite number of 2-D images into the retina as the object's position, pose, lighting, and background vary relative to the viewer. Indeed, although we typically see an object many times, we effectively never see the same exact image on our retina twice. [11, 12].

The regular cameras can only produce 2D images which is not enough to get the 3D information without further complicated processing. The two most common methods of geometrical modeling of a scene are: a) using multiple 2D passive images of the scene which also called stereo vision [13], and b) using 3D point clouds like LiDAR (Light Detection And Ranging) or LADAR (LAser Detection And Ranging) data [14]. A sequence of images taken by a single camera makes it easier to estimate some of the 3D image contents with limitations [15, 16] . A multiple views taken by multiple cameras to the same scene is thought to be the best to get a 3D reconstruction of the scene [2, 7].

Focusing in scenes containing planes and parallel lines, Criminisi et al 1999 [16], described how aspects of the affine 3D geometry of a scene can be measured from a single perspective image. They utilized the minimum geometric information determined from the image, namely the vanishing line of a reference plane, and a vanishing point for a direction not parallel to the plane. Their approach can be used for manually analyzing scenes for forensic science, virtual modeling and modeling paintings.

In his research, Prudhvi Gurram, 2009 [14] investigated the automation of 3D object modeling from aerial video imagery by applying the concept of stereo vision. He utilized multiple aerial video frames to form a stitched stereo mosaics that can provide the elevation information of the scene and act as good 3D visualization tools. The resulting 3D information is used to identify man-made buildings using probabilistic inference provided by Bayesian Networks.

Tobias Kotthäuser, 2007 [17] built a MATLAB® application that successfully used stereo vision for reconstruction and manipulation of cylindrical objects in specific environment that meets certain conditions of illumination, shadows and background. The application could determine the location and minimum geometrical aspects of the cylindrical object that are required for the handling. The cylindrical objects were painted with matt-finished black and were placed in an environment with white background at a special lighting setup that

will mostly eliminate shadows. The cylindrical objects could be recognized, and picked up while in different orientation, positions, elevation, within the defined work region which is about the size of a Letter size paper.

III. Camera Model

The famous perspective pinhole camera model, Figure 2.1, is used to model a center projection and to map the 3-D world coordinates to 2-D image coordinates. The center of projection is point *O* of the camera frame *F*. The camera image plane *uv* is parallel to the *xy* plane and displaced from it with a distance equal to the focal length, *f*, from *O* along the *z* axis which is also called the optical axis or the principal axis. The intersection of the image plane and the *z* axis is the principal point. The *u* and *v* axes of the 2-D image plane coordinates are parallel to the x and y axes of the 3-D camera frame coordinates, respectively. The coordinates of the principal point in the image plane are $[u_o, v_o]^T$.

Let *P* be an arbitrary 3-D point located in the positive side of the *z* axis and *p* is its projection on the image plane. The coordinates of *P* in the camera frame is $[x_p, y_p, z_p]^T$ and its projection can be presented by the intersection of the line connecting *O* and *P* with the image plane, Figure 2.2. The projection of *P* on the camera plane can be denoted as $\left[\frac{f\,x_p}{z_p}, \frac{f y_p}{z_p}, f\right]^T$ or $[x_c, y_c, f]^T$.

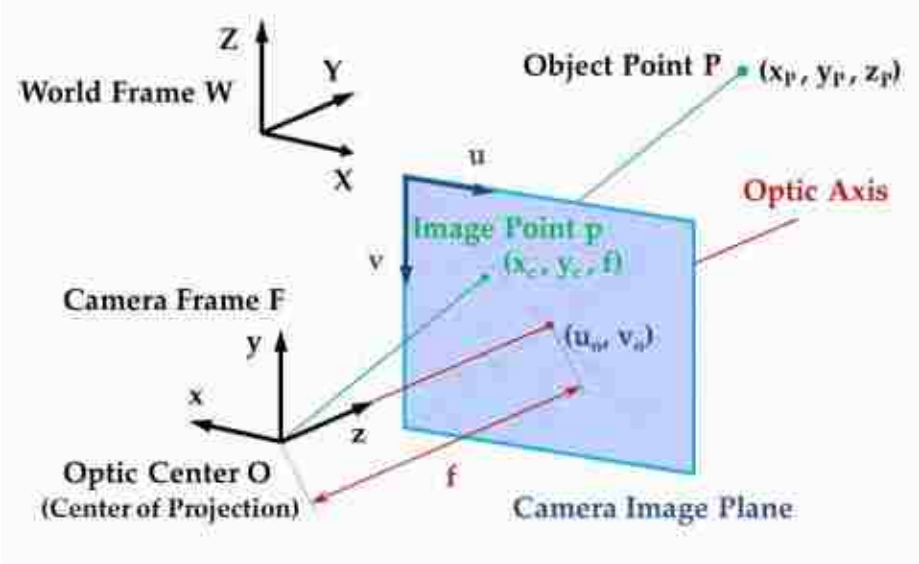


Figure 2.1 The pinhole camera model with focal length f and principal point (u_o , v_o)

Ignoring the final coordinate, the central projection can be defined as the 3-D to 2-D map:

$$\begin{bmatrix} x_p \\ y_p \\ z_p \end{bmatrix} \rightarrow \begin{bmatrix} \frac{f\,x_p}{z_p} \\ \frac{f\,y_p}{z_p} \end{bmatrix} = \begin{bmatrix} x_c \\ y_c \end{bmatrix} \tag{2.1}$$

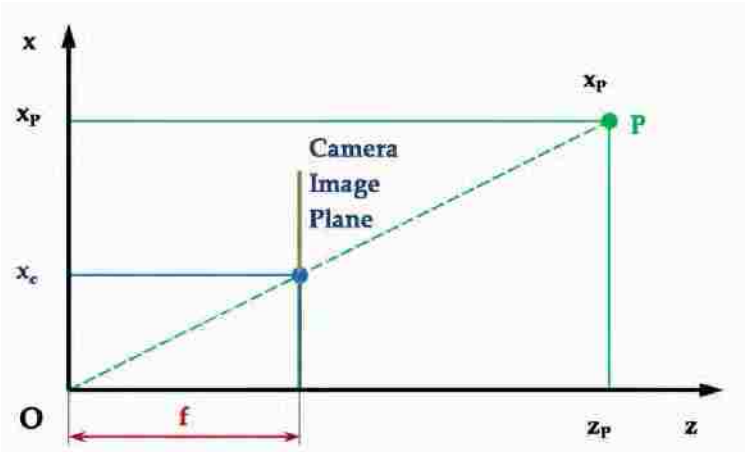


Figure 2.2 Pinhole Camera Model – xz Plane

The coordinates of P in the world frame are $[X, Y, Z]^T$. The coordinates of p in the image plane is $[u, v]^T$ and they can be solved from the homogeneous coordinates given by the transformation:

$$\begin{bmatrix} u \\ v \\ 1 \end{bmatrix} \propto \begin{bmatrix} \lambda u \\ \lambda v \\ \lambda \end{bmatrix} = F \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} = PM \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} \tag{2.2}$$

where F is the Perspective Transformation Matrix (PTM),

$$P = \begin{bmatrix} f & \alpha_c f & u_o & 0 \\ 0 & sf & v_o & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \tag{2.3}$$

where λ is is a scale factor, s is the pixel aspect ratio, α_c is the pixel skew coefficient. Parameters s, f, α_c, u_o, v_o are called the intrinsic camera parameters and can be evaluated through the camera calibration process as described in the following section. M is a 4x4 matrix describing the mapping between the world coordinates and camera coordinates. M is decomposed as follows:

$$M = \begin{bmatrix} \mathrm{R} & T \\ 0 & 1 \end{bmatrix} \tag{2.4}$$

where

$$T = \begin{bmatrix} T_x \\ T_y \\ T_z \end{bmatrix} \tag{2.5}$$

describes the translation between the two frames, and R is a 3x3 orthonormal rotation matrix that defines the rotation of each of the axes between the two frames.

IV. Camera Calibration

Camera calibration is an important process that is needed to get both the intrinsic and extrinsic camera parameters [18, 19, 20, 21, 22]. These parameters are essential to form the transformation between image plane coordinate (2-D pixel coordinates) and the 3-D camera frame coordinates and world frame coordinates and compensate for pixel skew, pixel aspect ratio, and lens distortion. The intrinsic camera parameters include the focal length in pixels, f, the principal point coordinate (also known as camera center point), cc=(u_o, v_o), the skew coefficient, α_c, that defines the angle between the x and y pixel axes, and the distortion coefficient, k_c, that defines both radial and tangential distortion components. These parameters build the camera matrix that relates the 2-D pixel coordinates (image plane coordinates) of any point in space to the 3-D coordinates with respect to the camera reference frame. If P =(x_p, y_p, z_p)^T is the coordinates of point P in the camera reference frame that projects as p =(x_c, y_c, f)^T in the image plane with respect to the camera reference frame, and as p=(u, v) with respect to the image plane frame. The normalized image projection, p_n is given by:

$$p_n = \begin{bmatrix} \frac{x_p}{z_p} \\ \frac{y_p}{z_p} \end{bmatrix} = \begin{bmatrix} x \\ y \end{bmatrix} \tag{2.6}$$

Consider the lens distortion model [23, 24] that includes both radial and tangential distortion coefficients, a five terms distortion coefficient vector, *k*, is formed using second and fourth order terms for radial distortion, and two components for tangential distortion and one sixth order radial distortion term in the fifth entry of the vector. The distortion coefficient vector can be visualized as:

$$k = [k_{r,2} \quad k_{r,4} \quad k_{t1} \quad k_{t2} \quad k_{r,6}] \tag{2.7}$$

Let $r^2 = x^2 + y^2$, the radial distortion vector is defined as:

$$p_{d,r} = \left(1 + \ k_{r,2} * r^2 + k_{r,4} * r^4 + k_{r,6} * r^6\right) * \ p_n \tag{2.8}$$

The tangential distortion vector is defined as:

$$p_{d,t} = \begin{bmatrix} 2k_{t1}\, x\, y + k_{t2}(r^2 + 2x^2) \\ k_{t1}\,(r^2 + 2y^2) + k_{t2}\, x\, y \end{bmatrix} \tag{2.9}$$

Therefore, total lens distortion is:

$$p_d = \left(1 + \ k_{r,2} * r^2 + k_{r,4} * r^4 + k_{r,6} * r^6\right) \begin{bmatrix} x \\ y \end{bmatrix} + \begin{bmatrix} 2k_{t1}\, x\, y + k_{t2}(r^2 + 2x^2) \\ k_{t1}\,(r^2 + 2y^2) + k_{t2}\, x\, y \end{bmatrix} \tag{2.10}$$

The final pixel coordinates p_{pixel} of the projection of *P* on the image plane is:

$$p_{pixel} = \begin{bmatrix} x_{pixel} \\ y_{pixel} \end{bmatrix} = \begin{bmatrix} f + p_d(1) + \alpha_c * p_d(2) + u \\ s * f * \ p_d(2) + v \end{bmatrix} \tag{2.11}$$

Therefore, the pixel coordinate vector, p_{pixel} and the normalized distorted coordinates are related to each other according to the linear equation:

$$\begin{bmatrix} x_{pixel} \\ y_{pixel} \\ 1 \end{bmatrix} = KK \begin{bmatrix} p_d(1) \\ p_d(2) \\ 1 \end{bmatrix} \tag{2.12}$$

Where *KK* is the camera matrix and is defined by:

$$KK = \begin{bmatrix} f & \alpha_c f & u_o \\ 0 & sf & v_o \\ 0 & 0 & 1 \end{bmatrix} \quad (2.13)$$

The extrinsic camera parameters include the 3x1 translation vectors, T_c, and 3x3 rotation matrices, R_c, that transform from the camera reference frame to a coordinate system attached to the calibration object (or in general, to the world reference frame using the relative position of the camera and the desired world reference frame). If $P_w = (X, Y, Z)^T$ is the coordinate of a point in the world reference frame, then the coordinate of the same point in the camera reference frame, $P = (x_p, y_p, z_p)^T$ is defined by:

$$P = R_c * P_w + T_c \quad (2.14)$$

V. Challenges for Image Segmentation and Object Recognition

In this section some of the tasks and challenges commonly faced in the area of image segmentation and object recognition are described. Solutions to selected problems are introduced in the next chapter.

A. Shadow Detection and Elimination

One of the challenging problems when dealing with images is the presence of shadow. It causes the false classification of cast shadows as objects, detection of extra false edges, or misclassification of shadow points as foreground, [25], [26]. This problem was addressed by many researchers in two

phases, namely; shadow detection and shadow elimination. While it was easy to detect shadows on most of the researches, depending on the efficiency of detection algorithm, there was a difficulty to get a completely shadow free images without degrading the image quality. Some shadow detection algorithms, [27], [28], were based on a sequence of images, which is commonly used in detecting moving shadows, while others were based on a single image. Based on a sequence of images, the shadow detection algorithms can be either deterministic or statistical [29]. Based on single image, the shadow detection and removal algorithm uses either an invariant color feature method, color statistical information, Gaussian Mixture Model (GMM), multi-label conditional/Markov Random Field (MRF) method [30], [31], gray scale pixel intensity information, [32], or energy function [33].

The main aspects of the shadow detection algorithms are the diversity and the dynamic nature of environments for which shadow detection is required in vision applications. The algorithms developed, therefore, often have specific strengths and limitations (e.g., indoor/outdoor only), and are designed for particular data domains (e.g., color/monochrome, pixel/transform). A particular algorithm may be optimal for a specific application and may perform effectively without modification. However, due to the complex nature of many environments, adaptive and/or hybrid forms of existing approaches may best be

able to meet the needs of dynamically changing conditions. There is still a need for further research in this direction which will require a comprehensive analysis of the specific environment, and its dynamic nature, prior to the determination of optimal combinations [26].

B. Edge Detection

Edge detection is one of the common image processing tasks. There are many methods to perform edge detection but they can be classified under two main categories; gradient-base and Laplacian-based. Gradient-based methods detect high spatial derivatives that mark strong intensity change or edges. The gradient is approximated by the following equations:

$$\nabla x = \frac{\partial f(x,y)}{\partial x} = \frac{f(x + d_x\,, y) - f(x,y)}{dx} \tag{2.15}$$

$$\nabla y = \frac{\partial f(x,y)}{\partial y} = \frac{f\left(x,\ y + d_y\right) - f(x,y)}{dy} \tag{2.16}$$

where *dx* and *dy* are the horizontal and vertical distance between two adjacent pixels. The gradient vector is

$$\nabla f(x,y) = \left(\frac{\partial f(x,y)}{\partial x}, \frac{\partial f(x,y)}{\partial y}\right) \tag{2.17}$$

The magnitude of the gradient can be calculated using the Euclidian norm

$$|\nabla f| = \sqrt{\left(\frac{\partial f(x,y)}{\partial x}\right)^2 + \left(\frac{\partial f(x,y)}{\partial y}\right)^2} \tag{2.18}$$

which is computationally expensive and can be approximated by the following norm 1

$$|\nabla f| = \left|\frac{\partial f(x,y)}{\partial x}\right| + \left|\frac{\partial f(x,y)}{\partial y}\right| \tag{2.19}$$

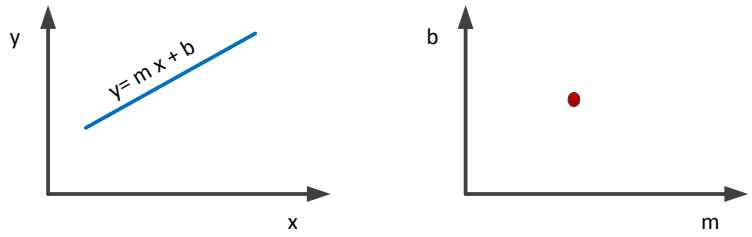
Among the several edge detection techniques, Canny's edge detection algorithm [34] is adopted for its known efficiency and accuracy. Canny proposed his algorithm based on the following criteria; low error rate, accurate localization of edge points, considering only one response to a single edge. Based on these criteria, Canny's technique has the following sequence; smoothing image to eliminate noise, finding image gradient to identify high spatial derivatives regions, inspecting each of these regions and eliminating all non-maximum pixels. The remaining pixels, that have not been eliminated, were reduced using a two threshold hysteresis. If the pixel gradient is less than the lower threshold, it is considered a none-edge. If the pixel gradient is greater than the higher threshold, it is considered an edge. If the pixel gradient is between the two thresholds, it is considered non-edge unless it is connected to a pixel with a gradient greater than the higher threshold. Canny used the calculus of variations to meet these criteria. This leads to an optimal function in Canny's detector that is given by the sum of four exponential terms, but it can be approximated by the first derivative of a Gaussian [34, 35, 36, 37].

C. Line Detection

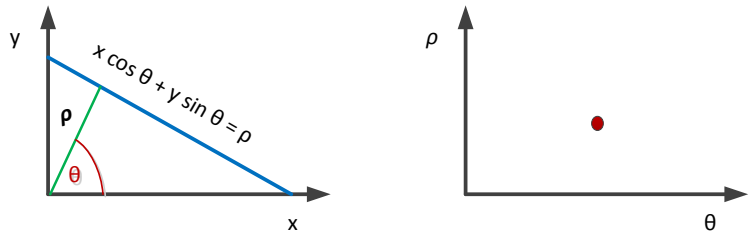
A common problem in computer vision is the detection of straight lines in digitized images. In a simple black and white image, the picture contains a number of discrete pixels lying on a white background. The problem is to detect the existence of groups of collinear or almost collinear pixels. This problem can be solved to any desired degree of accuracy by testing the lines formed by all pairs of pixels. However, this method is computationally expensive. [38]

Hough, 1962 [39] proposed a computationally efficient method for detecting lines in binary images. Rosenfeld [40] described a technique based on Hough's method [39] that replaces the problem of finding collinear points by a mathematically equivalent problem of finding concurrent lines. This method involves mapping each of the pixels into a straight line in a parameter space. The parameter space is defined by the parametric representation used to describe lines in the picture plane. Hough used the common slope-intercept parameters to represent his two-dimensional parametric plane. Since both parameters are unbounded the application of this technique was complicated. Duda and Hart [38] proposed an alternative parametric plane to eliminate this problem and interpreted Hough's method in two alternative ways that clarified its efficiency. They used angle-radius, Figure 2.3, rather than slope-intercept parameters which simplifies the computation, and extends the application of Hough's method to be

used for more general curve fitting. They also showed how to extend this method to find other general classes of curves in images. A variation of this approach was used by Griffith [41] to find long lines in a picture.



(a) slope-intercept space



(b) radius-angle space

Figure 2.3 Mapping a straight line into parametric spaces

Any line can be represented by its slope-intercept equation using x-y space as follow:

$$y = m\,x + b \tag{2.20}$$

where m is the slope, b is the intercept. The previous equation can be rewritten in the Hough parameter slope-intercept (m-b) space as

$$b = -xm + y \tag{2.21}$$

Now, x and y are constants, m is a variable, b is a function of m.

In the angle-radius (ρ-θ) parameter space, the line is represented by:

$$x\cos\theta + y\sin\theta = \rho \tag{2.22}$$

and any point in x-y space maps to a cosine curve. All points that belong to the same line map to the common intersection point of the cosine curves representing these points.

CHAPTER 3

PRELIMINARY IMAGE PROCESSING TASKS

Abstract

In this chapter, the calibration procedure that is used to evaluate and validate camera parameters is presented. The improvement on edge detection by means of eliminating shadow effects in engineering environment is discussed. In addition, algorithms are introduced to address some of the problems accompanying the application of the line detection using Hough transform.

I. Camera Calibration Procedure:

USB cameras with commercial specifications of $\frac{1}{7}$" CMOS sensor of 300,000 pixels, lens aperture of f/2.6, focal lengths F=2.8, f=4.9 at 30 mm, ∞. These cameras provide up to 640 x 480 Super VGA images with 30 frames per second. To get the exact parameters, calibration was done using the camera calibration toolbox for MATLAB introduced by Jean-Yves Bouguet, [42]. The details of the calibration process for one of the cameras are given below.

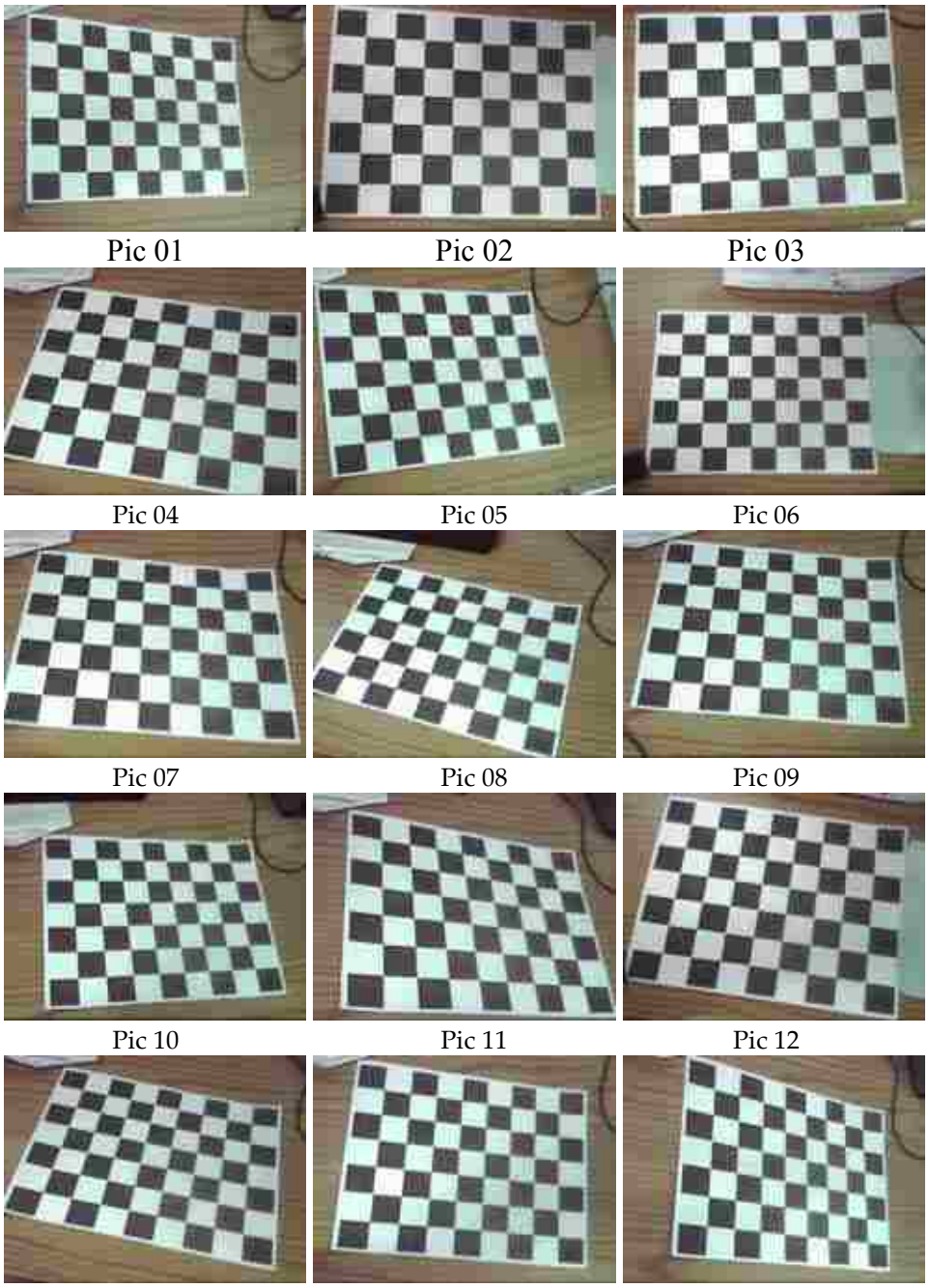


Figure 3.1 A set of 15 checkerboard pictures taken to perform camera calibration

A set of 15 images taken to a 30x30 mm checkerboard were used to perform the calibration, Figure 3.1. Steps of identifying grid and extracting its corners are displayed in Figure 3.2, Figure 3.3 and Figure 3.4.

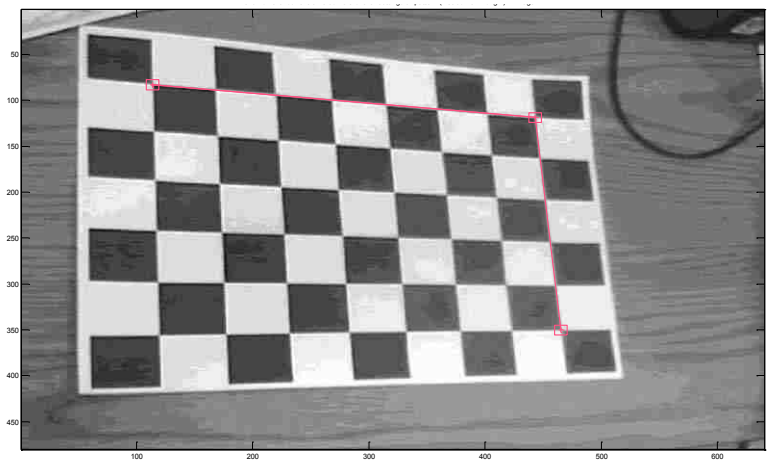


Figure 3.2 Marking the four extreme corners of the checkerboard image, pic01.jpg

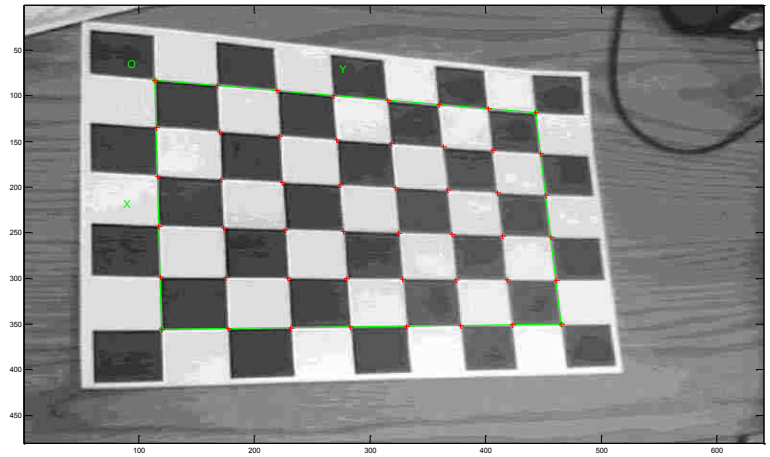


Figure 3.3 Resulting boundaries of the calibration grid on pic01.jpg

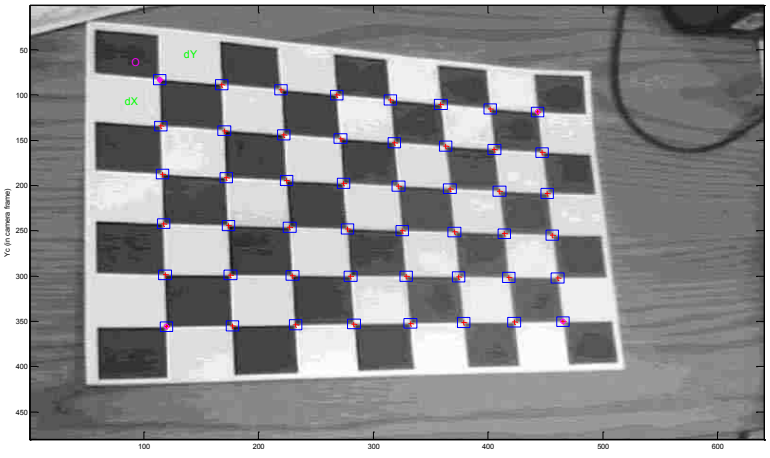


Figure 3.4 Extracted corners for pic01.jpg

Table 3.1 displays the values of optimized camera intrinsic parameters resulting from calibration with uncertainties:

Table 3.1 Camera intrinsic parameters resulting from calibration

Parameter	value
Focal Length:	$[f, s f]$ = [874.33369 873.87717] ± [8.80390 8.13899]
Principal point:	$[u, v]$ = [321.85898 280.41214] ± [5.49404 7.25276]
Skew:	α_c = [0.00000] ± [0.00000] => angle of pixel axes = 90.00000 ± 0.00000 degrees
Distortion:	k = [0.11593 -0.40386 0.00606 -0.00053 0.00000] ± [0.03236 0.28567 0.00228 0.00288 0.00000]
Pixel error:	err = [0.26596 0.31451]

The extrinsic camera parameters which give the relative positions of the checkerboard grids with respect to the camera can be visualized in Figure 3.5.

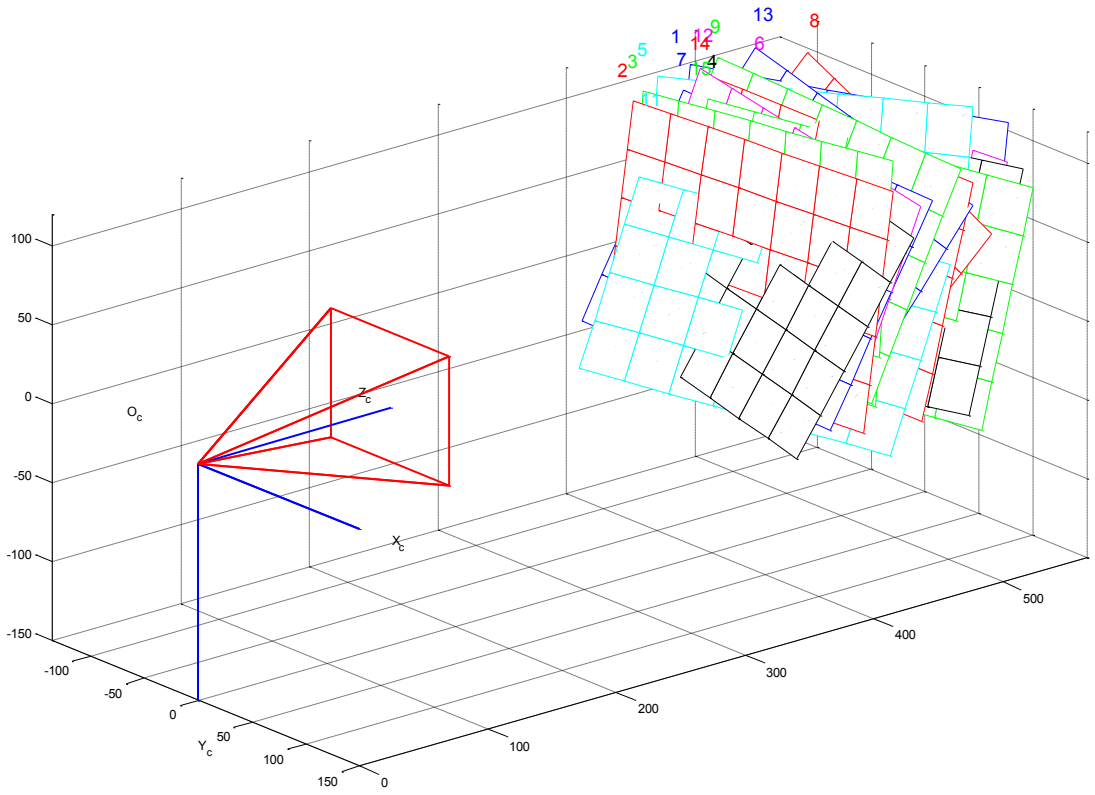


Figure 3.5 Camera extrinsic parameters with Camera centered

II. Testing Images

Several testing images were used to investigate possible problems and improvement on the edge detection, line extraction, and corner extraction processes. Samples of these images are presented in Figure 3.6. Color darkness and finish along with the lighting conditions has a great impact on the images taken. Figure 3.6, Pic 01 to Pic 04, shows an Aluminum prismatic object in its original metallic color in special lighting condition to eliminate shadows. In this color condition, light gray, shadows are close in color to the target object and hence they should be eliminated using a proper lighting. Eliminating shadows using four light from the four corners is a good solution with one disadvantage of increasing the brightness of the target object, or in other words increasing illumination and reflection on some of the object surfaces. A proper light filter is used to decrease light intensity and hence reduce the effect of shadow, illumination and reflection (Pic 02 to Pic 04). Figure 3.6, Pic 05 to Pic 08, shows an Aluminum cylindrical object in its original metallic colors. Pic 05 and Pic 06 are taken with the extra four corner lights. This lighting condition results in some shadows. With adjustment of the lights position Pic 07 was taken. It shows no shadow but too much illuminations and contrast.

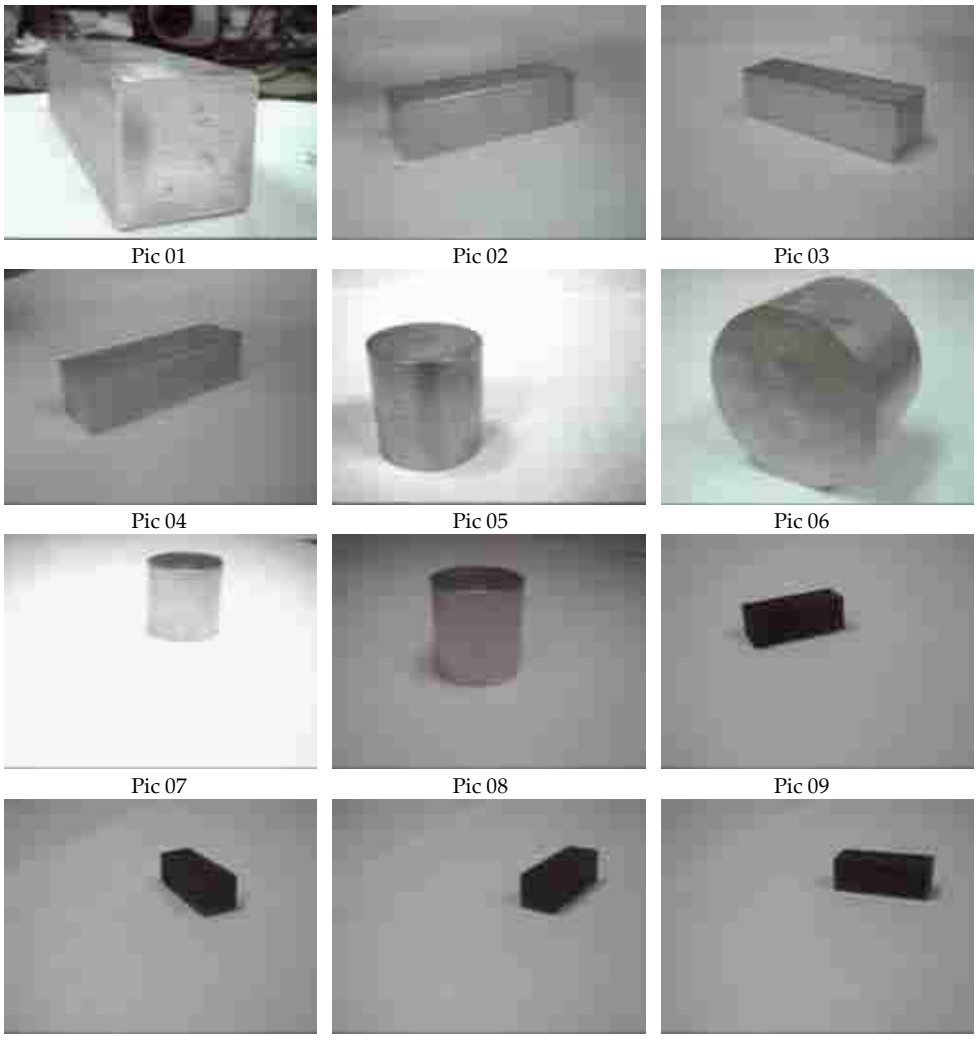


Figure 3.6 Sample testing images for prismatic and cylindrical objects

Pic 08 shows a good compromise after applying filters to the lights. It still suffers from the closeness of the objects color and the background color. Figure 3.6, Pic 09 to Pic 12, show a prismatic object with matt-finish black color in the regular room lighting condition. It shows some shadows but with high contrast with respect to the object itself. Primary image analysis was done based on these images.

III. Image preparation

Images for the prismatic objects were taken in different orientation to build and test the modules of detecting edges, extracting lines and corners. Before proceeding to edge detection and further phases, images were first transformed to a grayscale images using the standard NTSC formula which combines percentage of each color layer to form the gray scale image. To reduce the processing time, the input images were processed to crop the region of interest while saving the original positions of these cropped images to use them later in identifying object's 3D position.

IV. Edge Detection after Eliminating Shadow Effects using Image Shifting

With the current lighting condition (regular room lighting), the Canny edge detection technique worked with 10 out of 11 images with a threshold value of 0.5 (if one value is specified as the threshold, the lower threshold is taken as 40% of this value) and standard deviation of the Gaussian filter of 1.5, Figure 3.7 and Figure 3.8.

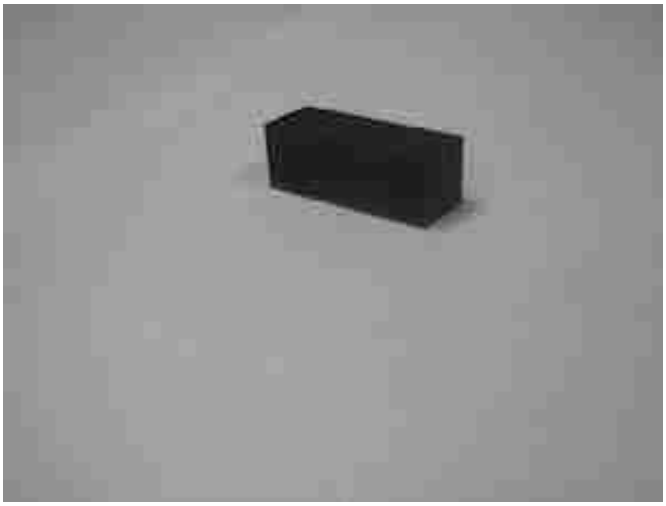


Figure 3.7 Image of the prismatic object, prism01.jpg

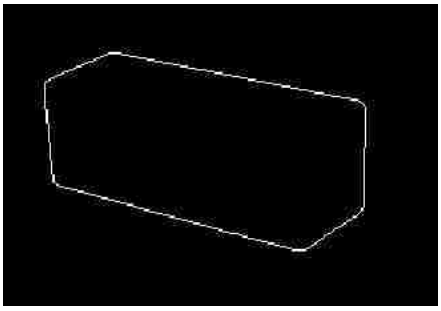


Figure 3.8 Results of Image cropping and Canny edge detection, prism01.jpg

The failed image was producing a proper edge map accompanied by an extra line due to shadow, Figure 3.9 and Figure 3.10.

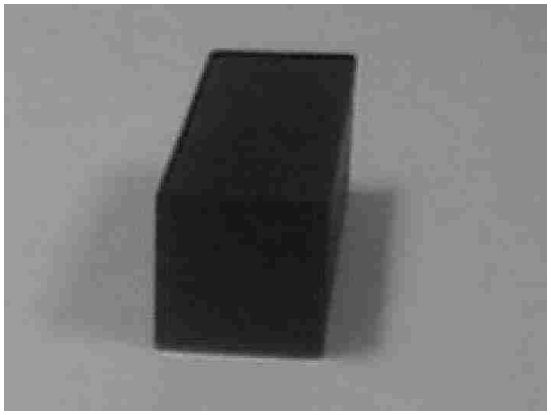


Figure 3.9 Image of the prismatic object, Prism08.jpg

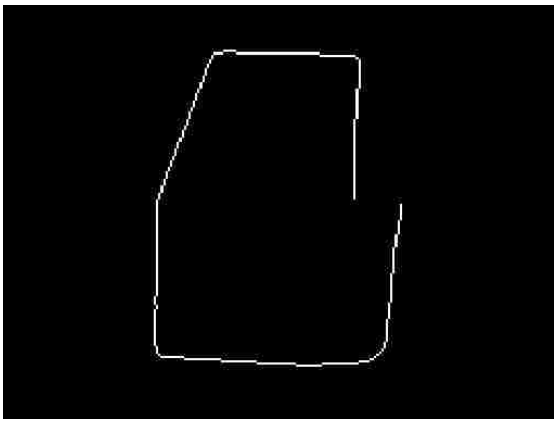


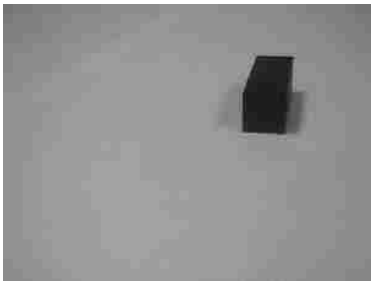
Figure 3.10 Results of Canny edge detection, Prism08.jpg

As stated in chapter 2, several shadow detection and removal techniques were proposed in the literature and each one of them has its points of strength along with points of weakness. None of these techniques could remove shadows without negatively affecting the image quality. For simple applications such as pick and place and localization of an object where the backgrounds are plain colored, the image quality in the shadow area is not of a great significance. Simple histogram equalization (dynamic range transformation) can reduce the intensity of shadow and the possibility of producing false edges. In general, this is not the perfect solution to eliminate the effect of shadows. A proposed image shifting technique following the image histogram equalization is introduced and proven to be eliminating the detection of false edges.

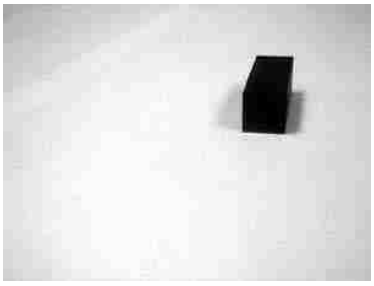
In the proposed image shifting technique, the pixel values are shifted and multiplied by a scaling factor. This can be represented by the following equation:

$$y_{shifted} = \propto y + \theta \tag{3.1}$$

where y is the original pixel value, α is the scaling factor, θ is the shifting value. This technique was applied to the images that failed to produce a proper edge maps using the standard Canny detector. The original image, the image after applying the histogram equalization, the equalized image after applying the image shifting, and the equalized shifted image after applying the histogram equalization one more time are presented in Figure 3.11.



(a) Original image, Prism08.jpg



(b) Equalized image

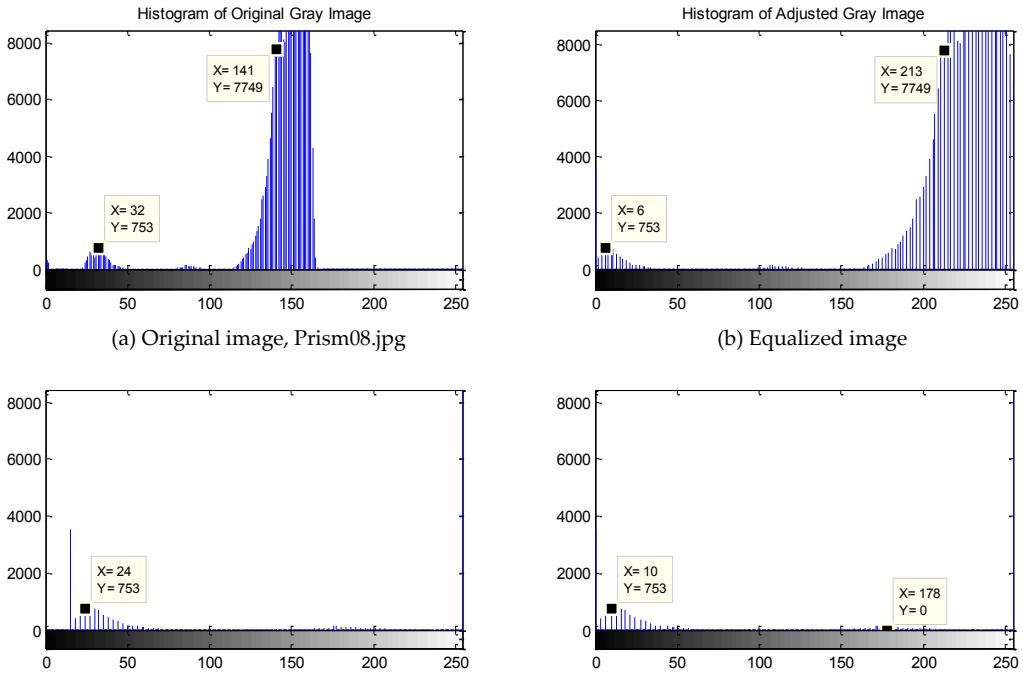
(c) Equalized then Shifted image

(d) Equalized, Shifted then Equalized image

Figure 3.11 Image with applying histogram equalization and shifting techniques

The histogram of each of these images appears in Figure 3.12. Applying the histogram equalization then the image shifting then the histogram equalization one more time resulted in shifting 294362 pixels to the maximum value of 255. The value of 255 or (2^8-1) corresponds to a full white color. The amount of background pixels (294362) that were turned to white with respect to the total

number of pixels (640 x 480=307200 pixels) on this image is 95.82 %. Similar results were obtained with images with the same problem.



(a) Original image, Prism08.jpg

(b) Equalized image

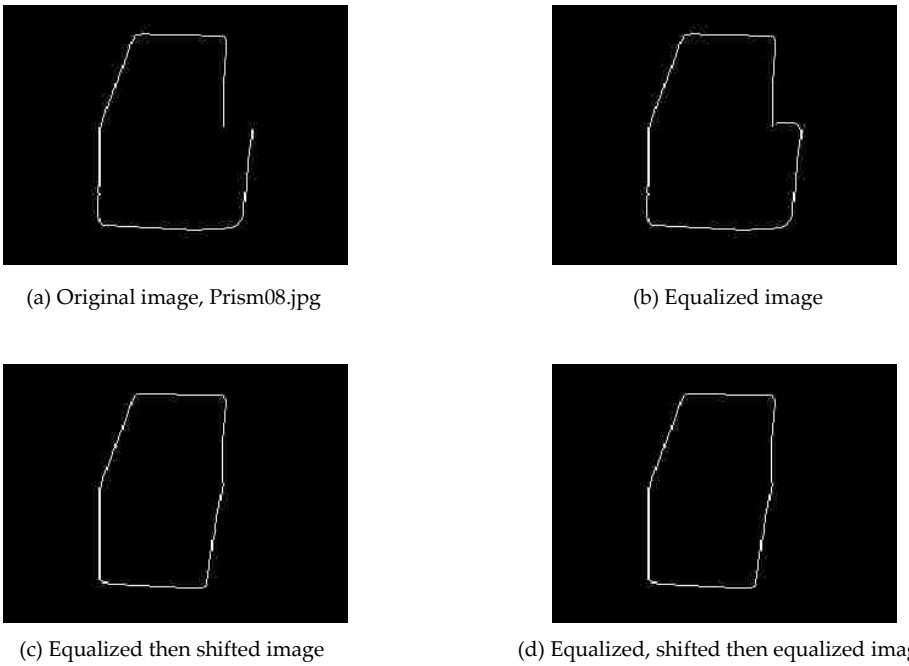
(c) Equalized then shifted image

(d) Equalized, shifted then equalized image

Figure 3.12 Histogram of image Prism08.jpg with equalization and shifting techniques

The effect of these successive processes on the detected edge map is presented in Figure 3.13. It is clear that the last two cases produce a proper edge maps with the last case having extra image histogram equalization step. The last histogram equalization process might not be necessary in case of this image but in some cases it improves the edges more than the third case. This procedure is proven to eliminate the effects of shadow on the detection of edges using Canny's technique. A possible disadvantage of this might be the reduction of the accuracy

of the detected edges especially around the corners. This problem is addressed during the process of reconstruction of lines using the Hough transform.



(a) Original image, Prism08.jpg

(b) Equalized image

(c) Equalized then shifted image

(d) Equalized, shifted then equalized image

Figure 3.13 Edge maps of image Prism08.jpg with equalization and shifting techniques

V. Line Reconstruction using Hough transform

As described in chapter 2, the Hough transform is used to extract line segments from the detected edges. The resulting lines are geometrically identified in terms of the coordinates of starting and ending points. Due to lighting, image quality, and the performed processes to reduce shadows, there is always inconsistency in the slopes of detected edges around the corners and weakness of edges around corners. In some cases, these inconsistency and weakness in edges causes the Hough transform to result in incomplete and fragmented lines. Figure 3.14 shows the extracted lines in colors and the edges that failed to be translated as line in white. Figure 3.14 shows 6 line segments on 5 sides of the edge map of the prismatic object. Only three of these lines are in the full

length. Two lines are lying in the same edge side while one edge does not show any extracted line. Adjusting the sensitivity of the Hough transform results in either the extraction of all the six lines, but not always in full length, lying in the corresponding six edges, or the extraction of more than six lines with at least one segment in each side and two or more segments in some sides (fragmented line).

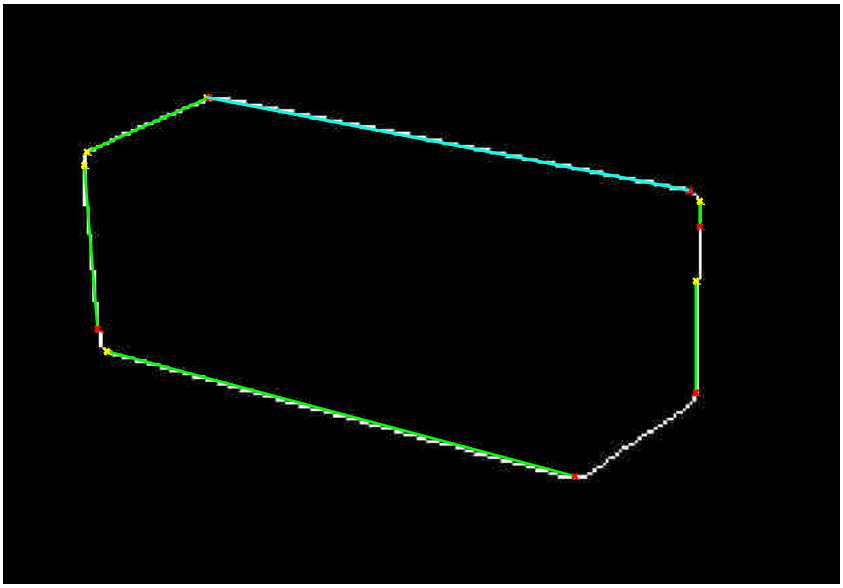


Figure 3.14 Extracted lines (6 segments in 5 sides), prism01.jpg

VI. Corner extraction and Line joining

To address these problems, the extracted lines were first arranged in either clockwise or counter clockwise order. An algorithm was developed to extrapolate the incomplete lines and extract the corner points. Another algorithm was developed to join the fragmented line segments, if any, and then apply the corner extraction algorithm. An example of each case is given below:

A. Extracting Corners from incomplete line segments

The Hough transform algorithm returns the extracted lines in a random order, Figure 3.17 (a), (b). Often times, these lines are missing small portion from each side. In this algorithm, the starting step is to put the extracted line segments in order. The starting point of the first extracted line is chosen as a starting point. All other terminal points are tested for the possibility of being on a line that should be intersecting with the current line to form the right polygon out of these lines. The criterion for testing these points is the mutual distance between points which was sufficient for all cases tested. So, the closest point to the ending point of current line marks the starting point of the next line in the order and so on.

Table 3.2 displays the extracted coordinates of starting and ending points of the lines resulting from applying the Hough transform to the detected edges of the image prism06.jpg.

Table 3.2 Coordinates of starting and ending points of extracted lines, Prism06.jpg

Line	Starting point		Ending point	
	x_s	y_s	x_e	y_e
1	45	68	180	36
2	117	148	231	117
3	42	70	42	114
4	240	108	244	63
5	45	135	75	157
6	204	32	240	47

The MATLAB function `line_pairs2intersect` uses the array `lines1` containing the coordinates of the starting points and ending points of the extracted lines as its first input argument. The second input argument, `all_short1_sorted`, is an array containing all possible distances from each point of any of the extracted lines to any other points on other lines sorted by original order of the lines then by the shortest distance. After eliminating pairs of lines that do not belong to adjacent sides, the result of this function is an array that gives the pairs of lines that should be intersected to analytically form the polygon that was represented by the edge map, Figure 3.15.

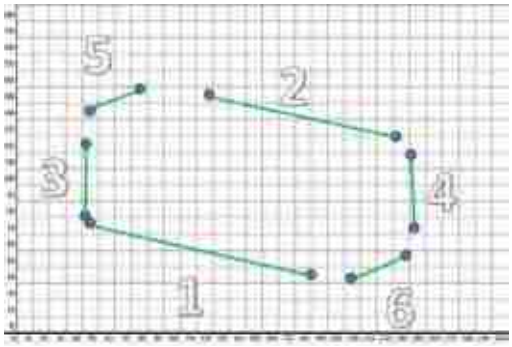
```
function line_pairs2intersect = lines2intersect (lines1, all_short1_sorted)
use1=zeros(size(all_short1_sorted,1),1);
use1(1:2)=1;
for i=size(lines1,1):size(all_short1_sorted,1)
    if all_short1_sorted(i,1)~=all_short1_sorted(i-1,1)
        use1(i)=1
        for j=1:size(all_short1_sorted,1)
            if all_short1_sorted(i,1)==all_short1_sorted(j,2) &&
                        all_short1_sorted(i,2)==all_short1_sorted(j,1)
                if use1(j)==1;
                    use1(i)=0
                end
            end
            if all_short1_sorted(i+1,1)==all_short1_sorted(j,2) &&
                  all_short1_sorted(i+1,2)==all_short1_sorted(j,1)
                if use1(j)~=1;
                    use1(i+1)=1
                end
             end
         end
    end
end
% Display the pairs of intersecting lines:
line_pairs2intersect=all_short1_sorted(use1==1,:)
```

Figure 3.15 MATLAB function for determining the pairs of intersecting lines

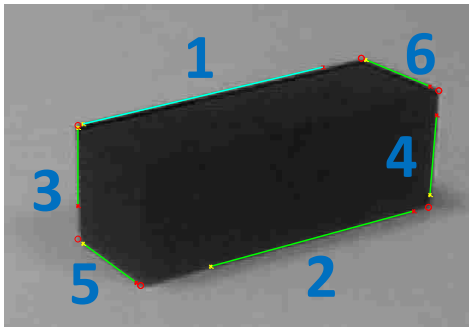
```
function intersec_pt=polygon_corners(lines1,line_pairs2intersect)
for i=1:size(line_pairs2intersect,1)
      intersec_pt(i,:)=lineIntersect(lines1(line_pairs2intersect(i,1),:),
      lines1(line_pairs2intersect(i,2),:))
      plot(intersec_pt(i,1),intersec_pt(i,2),'or')
end
```

Figure 3.16 MATLAB function for determining intersection point of extracted lines

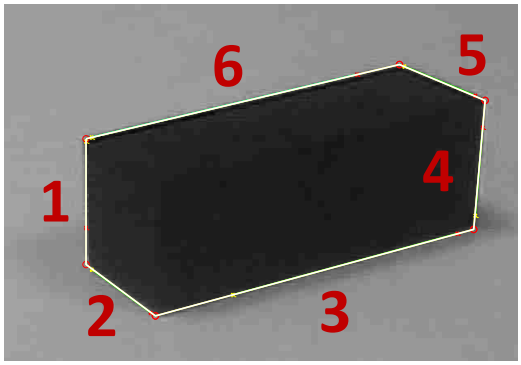
Once the pairs of line segments to be intersected are determined, the algorithm applies a line intersection subroutine to get the intersection of each pair. The function `intersec_pt` determines the all vertices of the edge polygon by applying the line intersection sub-function, `lineIntersect`, to each of the line pairs found by the `line_pairs2intersect` function, Figure 3.16. Finally these intersection points are arranged in counterclockwise order to get the corners of the contour polygon of the target object. Figure 3.17 (c) shows the complete polygon formed by the extracted lines in the counterclockwise order, marked by the red numerals, calculated by the proposed algorithm, along with the extracted corners, marked with red circles. Notice that the order of lines was shifted by one after using the first intersection point to mark the beginning of line 1. Table 3.3 displays the calculated coordinates of the corner points of the contour polygon in the counterclockwise order with respect to the image space.



(a) Retrieved lines - Cartesian coordinate



(b) Retrieved lines - Image space



(c) Complete Reconstructed Polygon with corners identified

Figure 3.17 Extracted line segments, using Hough transform, Prism06.jpg

Table 3.3 Coordinates of the corner points of the edge polygon

Point i	xi	yi
1	42	68.711
2	42	132.8
3	77.408	158.77
4	239.4	114.71
5	245.23	49.179
6	201.42	30.924
1	42	68.711

B. Joining fragmented line segments

Due to the weakness of some edges and the inconsistency of the slopes of the chains of points forming that edge, the Hough transform might recognize one line as two or more fragments. These two fragments will appear as almost collinear with minimum or no lateral gap. An algorithm is developed to fix this

problem. The algorithm start by checking for line fragments which are collinear or almost collinear. The criteria for determining whether two line fragments are collinear or not are mainly the difference in slope and the lateral distance. If the two line fragments has a difference in angle less than 5° and the lateral distance between them is 2 pixels or less (or if the distance between the two farthest points of the two fragments is less than or equal to the combined length of both lines plus the length of the gap between them), then the two fragment will be considered collinear, Figure 3.18. In this case a one line joining the two farthest points will replace these two fragments. The previous line extrapolation and corner extraction algorithms are then applied to get the contour polygon as before.

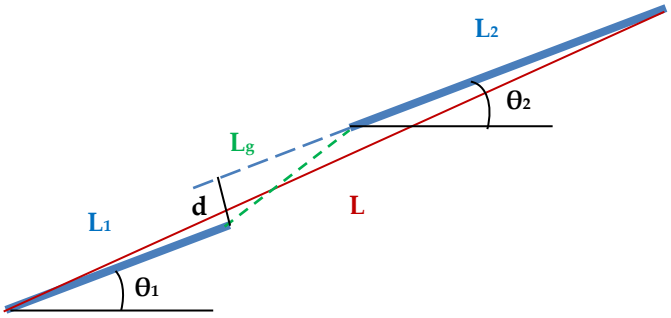


Figure 3.18 Two fragments of a line segment

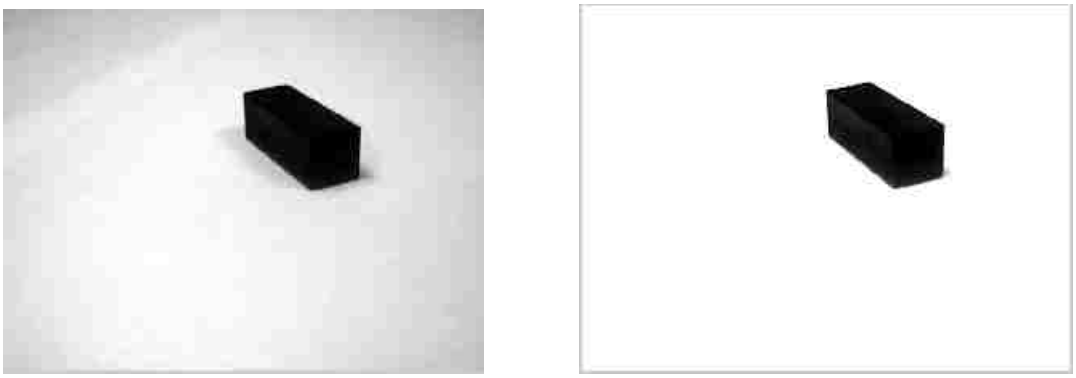
If two line fragments are collinear, then the following relation should be satisfied:

$$|\theta_1 - \theta_2| \leq 5^\circ \tag{3.2}$$

and

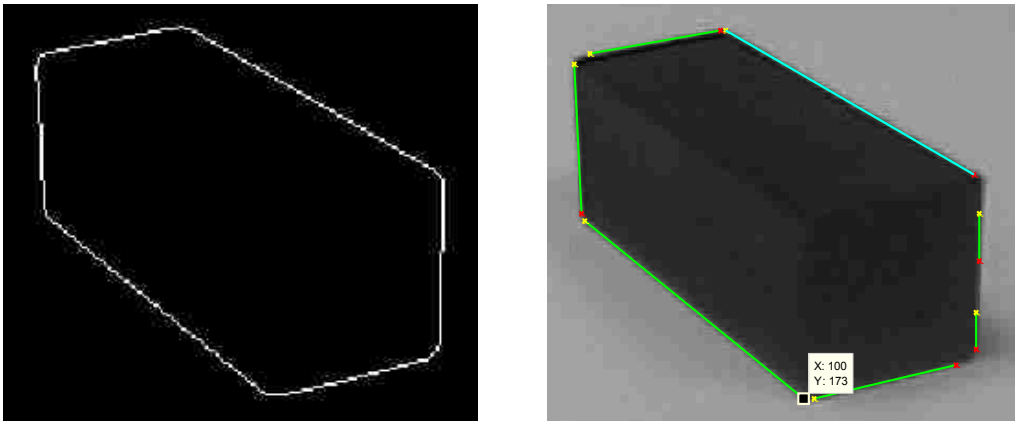
$$d \leq 2\ pixels \tag{3.3}$$

An example image for this case is represented in Figure 3.19. Figure 3.20 shows the detected edges and the extracted lines with 2 fragments in one side.



(a) Original image, Prism03.jpg (b) Equalized and shifted image, Prism03.jpg

Figure 3.19 Original and equalized-shifted images, Prism03.jpg



(a) Detected edges, Prism03.jpg (b) Extracted lines, Prism03.jpg

Figure 3.20 Detected edges and extracted lines, Prism03.jpg

The fragmented line are marked by numerals 3 and 4 in Figure 3.21. The developed algorithm detected these two fragments, joined them and rearranged all extracted lines by deleting these two segments and adding the joined line to the end of the list, (number 6 in Figure 3.22). The resulting lines appear with the new order marked with red numerals, Figure 3.22.

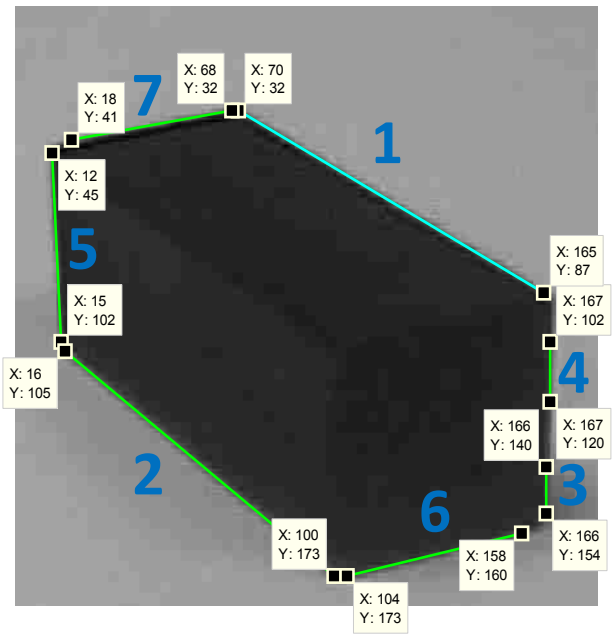


Figure 3.21 Extracted lines with coordinates, Prism03.jpg

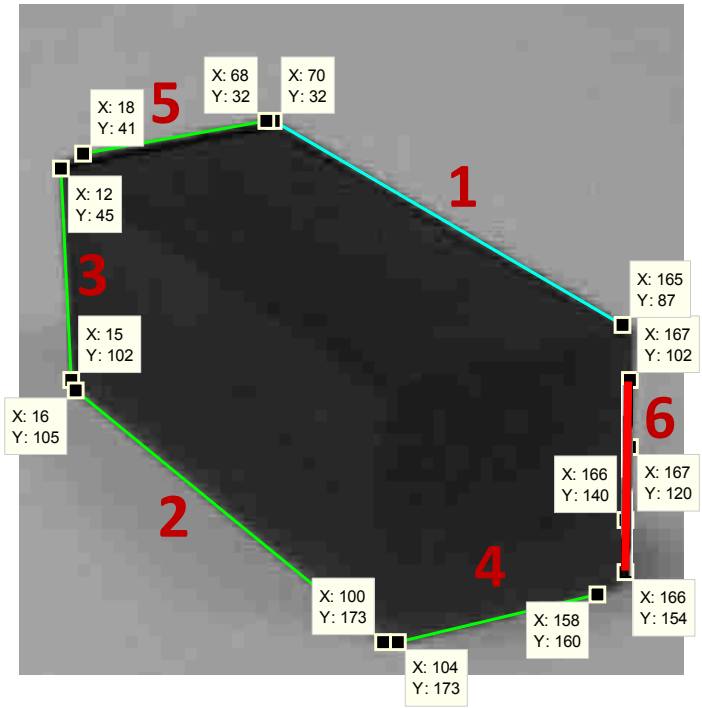


Figure 3.22 Extracted lines after joining two collinear fragments, Prism03.jpg

```
function [lines2, line_pairs2intersect,
all_short1_sorted]=chkParlzmAndElim(lines1, line_pairs2intersect,
all_short1, all_short1_sorted)
i=0;
lines2=[];
while i~=size(lines1,1) && size(lines2,1)~=size(lines1,1)
    lines2=lines1;
    for i=1:size(lines1,1)
        L1=lines1(line_pairs2intersect(i,1),:);
        L2=lines1(line_pairs2intersect(i,2),:);
        len1=linelength(L1)
        len2=linelength(L2)
        short_long=shortNlong(L1, L2)
        len_g=short_long(1,size(short_long,2))
        Len=short_long(size(short_long,1),size(short_long,2))
        max_lat_pts=2   % Maximum allowed lateral points between 2 lines
        if Len<=len1+len2+len_g && abs(angle2lines(L1,L2))<5
            new_line_xs_co=L1(2*short_long(size(short_long,1),1)-1)
            new_line_ys_co=L1(2*short_long(size(short_long,1),1))
            new_line_xe_co=L2(2*short_long(size(short_long,1),2)-1)
            new_line_ye_co=L2(2*short_long(size(short_long,1),2))
            new_line=[new_line_xs_co new_line_ys_co new_line_xe_co
new_line_ye_co]
            lines2(size(lines2,1)+1,:)=new_line
            lines2([line_pairs2intersect(i,1)
line_pairs2intersect(i,2)],:)=[]
            linemerge=1;
            break
        else
            linemerge=0;
        end
    end
    lines1=lines2;
    all_short1=all_short_Dist(lines1)
    all_short1_sorted=sortrows(all_short1,[1 size(all_short1,2)])
    line_pairs2intersect=lines2intersect(lines1,all_short1_sorted)
end
```

Figure 3.23 MATLAB function for determining intersection point of extracted lines

The function `chkParlzmAndElim` checks if any pair of lines that were determined by `line_pairs2intersect` function is collinear or almost collinear based on the criteria described above. If any of the pairs meet these criteria, the input argument, `lines1`, will be processed to produce a new array, `lines2`, after merging this pair of lines into one line, Figure 3.23.

VII. Conclusion:

For specific work environment with fixed lighting condition, optimized edge detection parameters can be chosen and applied for all images taken in this environment. Applying image histogram equalization along with the proposed image shifting technique results in enough elimination of shadows, and hence, edge detection that is satisfactory for engineering environment. Using Hough transform, lines can be extracted with some deficiencies. These deficiencies include missing parts of these lines, fragmentation of some lines. Algorithms are developed to address these problems.

In stereo vision system and using epipolar geometry, a set of two corresponding images with corner points extracted can be used to find the disparity matrix and identify some corresponding points [43]. This is a classical problem and to get the corresponding points, it will need to be solved using either an extensive searching algorithm or defining a geometric relation between the two images using the relative location and orientation of both cameras, see for example [2], [44]. This is computationally expensive and slow process relative to the theory introduced in the following chapter.

CHAPTER 4

IMAGE SEGMENTATION, OBJECT DETECTION, AND DISTANCE ESTIMATION (*) [45]

Abstract

In order for a robot to work on an object, first it has to recognize the object. The light conditions often times are such that the gray-scale or red, green, and blue values have a relatively small dynamic ranges. As a result of the small dynamic range it is very difficult for the existing edge detection algorithms to produce unbroken edges, and therefore properly segment the image. In order to overcome the impact of the light effects on the image first we introduce a transformation on the (r,g,b) values of the pixels, then we perform edge detection and segmentation, and finally we compute the distance of the object from the origin of a global Euclidean coordinate system. For the distance resolution we use two or more cameras. Our algorithm is based on our method that involves proper camera calibration, super triangulation, and least squares. Our algorithm is very fast compared to the epipolar method which is our competitor, and gives better accuracy than the epipolar method.

* - This chapter was submitted for publication in the International Journal of Advanced Robotic Systems (IJARS) [45].

I. Introduction

Automation and robotics is emerging as a very large area, and it is only going to get larger with time. The applications of automation and robotics are immense and they touch every area one can imagine, including manufacturing of goods, customer order fulfillment, space exploration, mining, underwater exploration, medical applications, including surgery, diagnostics, and home care. Also Housework, driving a car, flying an airplane, military applications, firefighting applications, police applications, instructions, and many other applications. The mechanical part of the robot although important is no longer as challenging, considering the advancements and know how we already have in this area. The robot vision and the intelligence of the robot is the part that will enable the robot to be aware of its environment and perform the tasks that is programmed to perform. Let's consider a real world example. Automation in customer orders over the internet for example. The customer activates the merchant's web site and orders a pair of shoes, sun glasses, a belt, five pairs of shocks, and a notebook. The customer pays with a credit card and the system sends the customer an electronic receipt with an order number and estimated day of arrival. The customer's order is transmitted to the robot server. The server identifies which robot is available and transmits the order to that robot. If all the robots are busy then the server waits until the first robot becomes available.

When a robot takes the customer's order, for each item in the list identifies the bin and its location coordinates in the warehouse. The location coordinates are with respect to a global warehouse coordinate system. Then the robot computes an optimal path that will minimize the total travel distance and therefore the total travel time. Based on the size of each object and the type of object (if it is fragile or not) the robot provides the proper packaging material and the proper size. It visits every bin, picks up the item, packages it appropriately and places all the items into a large enough box. When all the items are placed on the box, picks up the proper address sticker from the printer, places it on the box and places the box on the pickup bay to be loaded to the truck. While the robot is traveling across the aisles going from one bin to another, its path intersects the path of other traveling robots. In order for the robots to avoid collision they have to communicate and coordinate their travels. This is an example of automation implemented today by a number of e-commerce companies. One robot does the job of many employees. In a robotics environment today there are many components that have to be incorporated. Some of the important ones are robot vision, pattern recognition, artificial intelligence, wireless communication, networking and compression, network security, which includes channel security, authentication, server security, and robot security. In this chapter, the focus will be on robot vision, object detection, and distance resolution. Often times the light

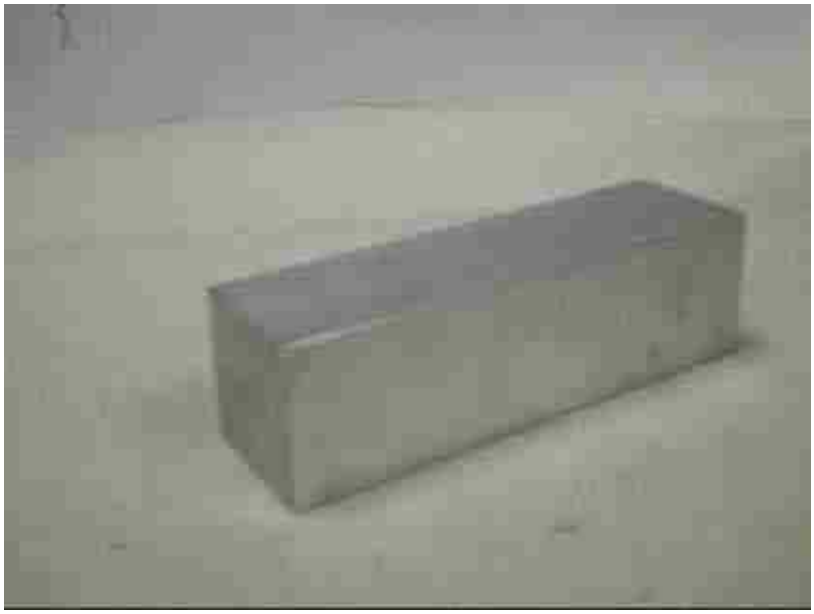
conditions are not right and the image obtained by the cameras have limited dynamic range. This limited dynamic range becomes an inhibitor for the edge detection algorithms which they produce broken edges often times very difficult to repair in post processing. A remedy to this problem as we show below is first to make a pixel transformation in the (r,g,b) space and increase the dynamic range from 0 to the maximum color value which is usually 255, the new image has higher contrast, it is usually sharper, and easier to detect the edges. Once the edges are detected then we can segment the image so that the object of interest is included in the segment. The segment is also called the video object plane. Using two or more cameras, we calibrate first the cameras, and then we trace every pixel of the edges by the two or more cameras finding the (x,y,z) coordinates of its point in the three dimensional space. The global Euclidean coordinate system is fixed and decided a priori. If the object has a known geometric shape this process enables us to reconstruct it in the 3-D space and find its dimensions, volume, and total area. If the 3-D object is irregularly shaped then we use a non-invasive laser along with the cameras to project a regular grid of squares onto the irregularly shaped object. Every square projection onto the object produces quadrilaterals irregularly shaped. Following the corners of its quadrilateral we are able to compute the exact grid coordinates. From these grid coordinates we are able to compute the volume of the object, as well as the total surface area of

the object. Our algorithm has many applications in robot vision, computer vision, automation, optimal loading of airplanes, ships, and trucks, as well as many other areas. Image processing, pattern recognition, camera calibration, distance resolution methods, motion detection, motion estimation, motion compensation, are very important for robot vision and robot intelligence. However these two areas are not the only important areas for robotics. Wireless communications, robot networking, robot server configuration, channel security, robot security, and server security are also as important.

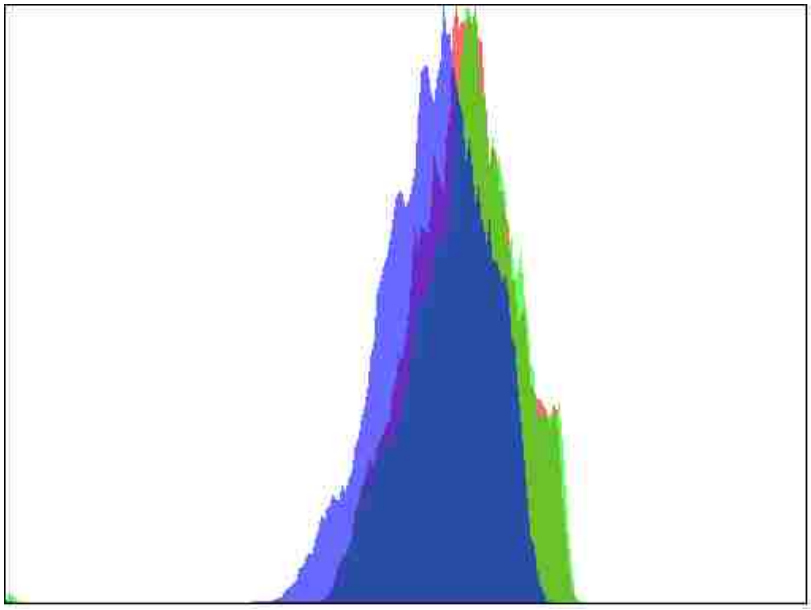
II. Multi Camera Vision System with Two-Mode Non Invasive Laser

Robot vision includes several steps. The first one is to understand where the object is and be able to outline in the image space the boundaries of the object. This image segmentation defines the segment in the image space that includes the object. That segment is also called the video object plane corresponding to that object. If more cameras are used it is very important to perform camera calibration first, and then use this knowledge and the segmentation to get the coordinates of the object when the object has one of the known geometric shapes. The theory that follows solves the problem we just stated. Every pixel in the image has three values, one for the red channel, one for the blue, and one for the green. We refer to this triad of pixel values as (r,g,b),

and except for the medical images $0 \leq r, g, b \leq 255$. If we consider all the red values for all the pixels of a certain image then there is a minimum red value r_{min}, which often times is greater than zero, and a maximum red value over the pixels of the image space which we will denote as r_{max} which is less than 255. For every pixel therefore its red value *r* is greater than or equal to r_{min} , and less than or equal to r_{max}. The dynamic range of the red component is represented in the literature as the difference $r_{max} - r_{min}$, or often times as the interval $[r_{min}, r_{max}]$. In a similar way we define the dynamic range for the green, and blue components. The image intensity, or luminance, or luma value for a pixel with (R, G, B) values for the red, green and blue components is denoted by Y, and is defined as Y=0.299*R+0.587*G+0.114*B. Notice that the sum of the coefficients is equal to 1, and the weight of the green is more than the sum of the weights of the red and blue. The reason for this is that the green is in the middle of the visible light color spectrum and is very stable. Also notice that the coefficient for the blue light is the smallest. This is because the blue light has the highest frequency of the three of them and is subject to atmospheric scattering and is very unstable. When the (R, G, B) channels have a small dynamic range then the Luma also has a small dynamic range. Figure 4.1 shows an image of a prismatic object and its histogram of the Red, Green, and Blue channels that experience a small dynamic range of 103 to 170.



(a) Image of a prismatic object



(b) Probability density function of the R,G,B channels of the prismatic object

Figure 4.1 Histogram of a prismatic object showing a small dynamic range, form 103 to 170.

Although on special occasions we apply the edge detection algorithms on one or two color channels, especially if these one or two channels are the only ones present in the image, usually we apply it on the luma Y. Thus if the dynamic range of the luma is very limited, then first we apply a dynamic range

transformation, then we reconstruct the image, and finally we apply the edge detection. The major transformations applied are either

$$Y_{ij} = \frac{y_{ij} - y_{min}}{y_{max} - y_{min}} 255 \tag{4.1}$$

where y_{ij} is the luma value at pixel coordinates (i,j), in the image space, and Y_{ij} is the transformed luma value. This produces a new grey scale image with dynamic range between [0, 255], as opposed to the dynamic range $[y_{min}, y_{max}]$ of the original image. The histograms of the original image and the new image look very similar, except the new image histogram stretches from 0 to 255. Another method used is the histogram equalization first and then the edge detection. The idea behind the algorithm is that the cumulative distribution function of any random phenomenon is a random variable with uniform distribution between [0,1]. Thus if for any intensity y the estimated cumulative distribution is F(y), and if the maximum intensity is 255 then the transformed intensity

$$i' = (m - 1)F(i) \tag{4.2}$$

If m=256 then m-1=255 and

$$Y = 255 * F(y) \tag{4.3}$$

performs the histogram equalization. If the dynamic range of the grey scale is between [a, b], a>0, b<255. Since F(a) is close to zero, 255*F(a) will be close to zero, and since F(b)=1, 255*F(b)=255. Histogram equalization therefore

increases the dynamic range from [a,b] to [close to zero, 255]. Histogram equalization can be applied to the luma, Y, and/or the R, G, B values. After drawing the boundaries of the objects in the image space, it is very important to be able to find the 3-D coordinates of each object in the 3-D space with respect to a coordinate system. In order to do that, we need two or more cameras. In the following we explain very briefly the camera calibration. More details can be found in the literature [24], [46], [47], [48], [49], [50]. The most commonly used mathematical model for camera representation is the pinhole camera model, and it is based on the principals of perspective projection. Using this model given a point P(X, Y, Z), in the three dimensional space, the projection $\hat{P}(x, y)$, is obtained by intersecting the ray joining the point P(X,Y,Z) and the center of projection L, which represents the center of the camera lens. Figure 4.2 shows the perspective projection.

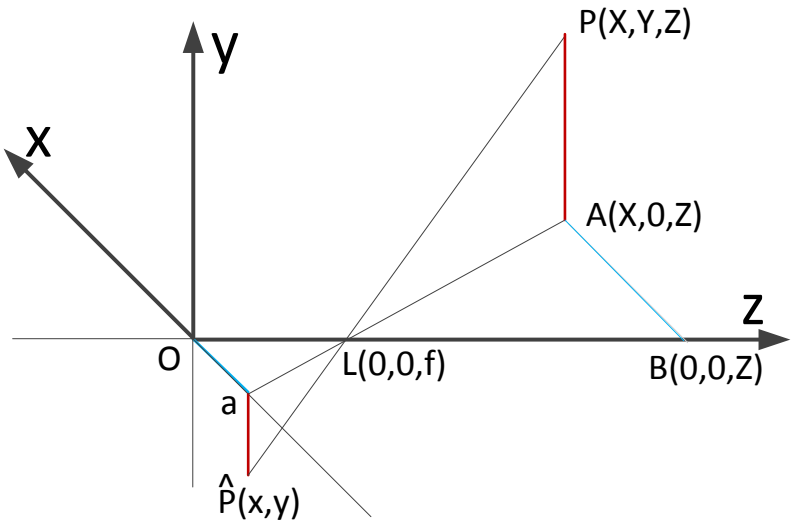


Figure 4.2 Perspective projection

The point L(0,0,f) is the center of projection and f is the Z-coordinate of the center point of the lens. The triangle BAL is a right triangle at A and is similar to the triangle aOL which is a right triangle with the right angle at O. From these two triangles we have $\frac{aO}{Ol} = -\frac{AB}{BL}$ or $\frac{x}{f} = -\frac{X}{Z-f}$ which implies

$$x = \frac{fX}{f-Z} \tag{4.4}$$

Similarly from the triangles $\hat{paL}$ and the triangle PAL which are right triangles at a, and A respectively and similar, we obtain

$$\frac{a\hat{p}}{AP} = \frac{aL}{AL} = \frac{OL}{LB} \quad \text{or} \quad \frac{y}{Y} = -\frac{f}{Z-f} \quad \text{or} \quad y = \frac{fY}{f-Z} \tag{4.5}$$

Depending on the camera model, when the world coordinate system is not the same as the camera model coordinate system then an affine transformation which is represented as a 4x4 matrix A would enable the two coordinate systems to coincide and therefore go from the world coordinate system to the camera coordinate system. This affine transformation for a pan and tilt camera consists of translations, and rotation about the Z-axis, and rotation about the X-axis. This is very well documented in the literature [50], [51], [52]. In stereo imaging one of the challenges is recognizing a common point in each one of the image planes of each camera. This is one of the reasons that our vision system has a cross configuration, Figure 4.3, and in the middle we have a non-

invasive laser, that has two modes. In the one mode it emits a dot, in the other the light goes thru a square grid which is projected on the object.

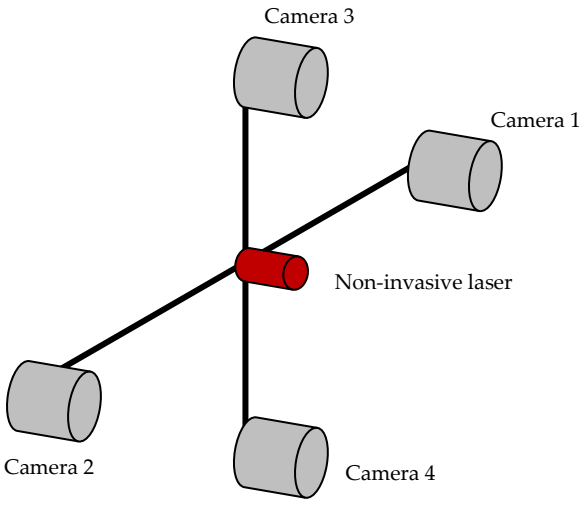


Figure 4.3 Cross configuration of the vision system

Once we recognize the object in the image space of every camera then as a first approximation we obtain the dimension of the object by recognizing common corner points and computing their world coordinates. Often times due to the light conditions the histogram of the luma has very small dynamic range, also the light effect is different for every camera depending on its position relative to the light. This makes it challenging to identify a common point in every image space of every camera.

One of the purposes of the laser is to project a point on the object which can easily be identified in each of the camera image spaces, and creates an undisputed common point. The other purpose of the laser is to project a grid of

squares on the object especially if it does not have a regular shape in order to estimate its total surface and volume. The camera system can pan and tilt and each one of the cameras can also pan and tilt. In our initial analysis we assume that all cameras are exactly the same with same type of lenses, and the distance of the center of the lens from the center of the image space is *f*, for each one of the four cameras. We also assume that the pan and tilt for each camera and the system of four cameras are zero so that all cameras have the same orientation and their lenses are parallel.

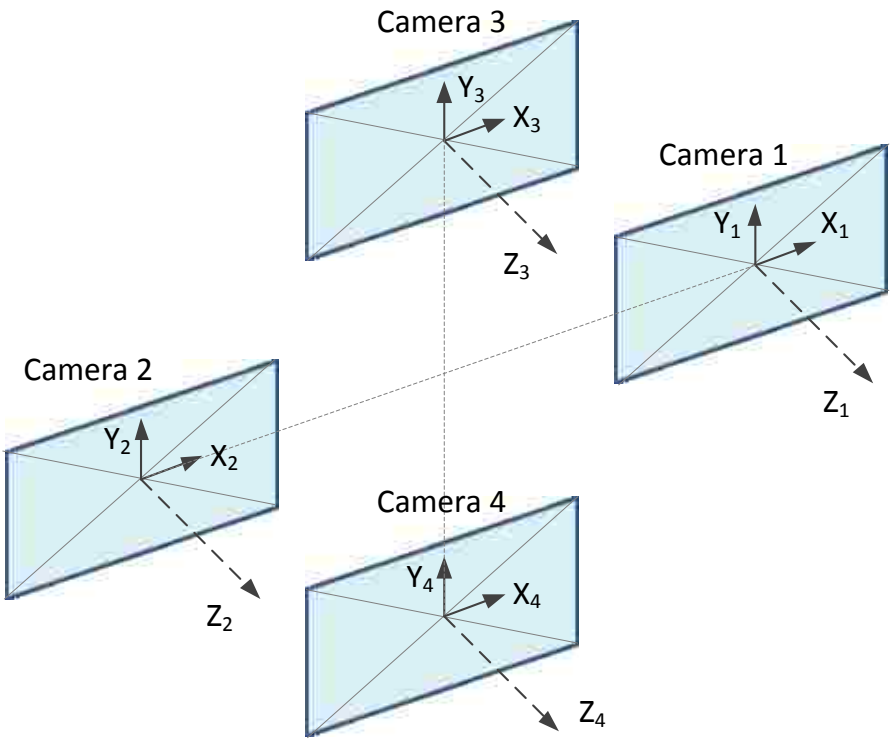


Figure 4.4 Parallel arrangement of the four cameras

Each camera has its own coordinate system with center in the center of the camera image plane, the Z –coordinate passing thru the center of the image plane

and the center of the lens. Also we assume that the coordinate systems of all four cameras are parallel as shown in Figure 4.4.

At first we consider the two cameras on the horizontal axis of the cross configuration, Figure 4.5.

Let A_x be the distance between the centers of the two lenses. Let P, be a point in the 3-D space. Let (x_1, y_1), be the image space coordinates of the point P on the camera 1 coordinate system, and $P(X_1, Y_1, Z_1)$ be the coordinates of the point P, with respect to the camera 1 coordinate system. Then

$$X_1 = \frac{x_1}{f}(f - Z_1) \tag{4.6}$$

$$Y_1 = \frac{y_1}{f}(f - Z_1) \tag{4.7}$$

The same point P has coordinates (X_2, Y_2, Z_2) with respect to camera 2 coordinate system. With

$$X_2 = \frac{x_2}{f}(f - Z_2) \tag{4.8}$$

Since in both camera coordinate systems

$$Z_1 = Z_2 = Z \tag{4.9}$$

and also since

$$X_2 = X_1 + A_x \tag{4.10}$$

then

$$X_1 = \frac{x_1}{f}(f - Z) \tag{4.11}$$

and

$$X_1 + A_x = \frac{x_2}{f}(f - Z) \tag{4.12}$$

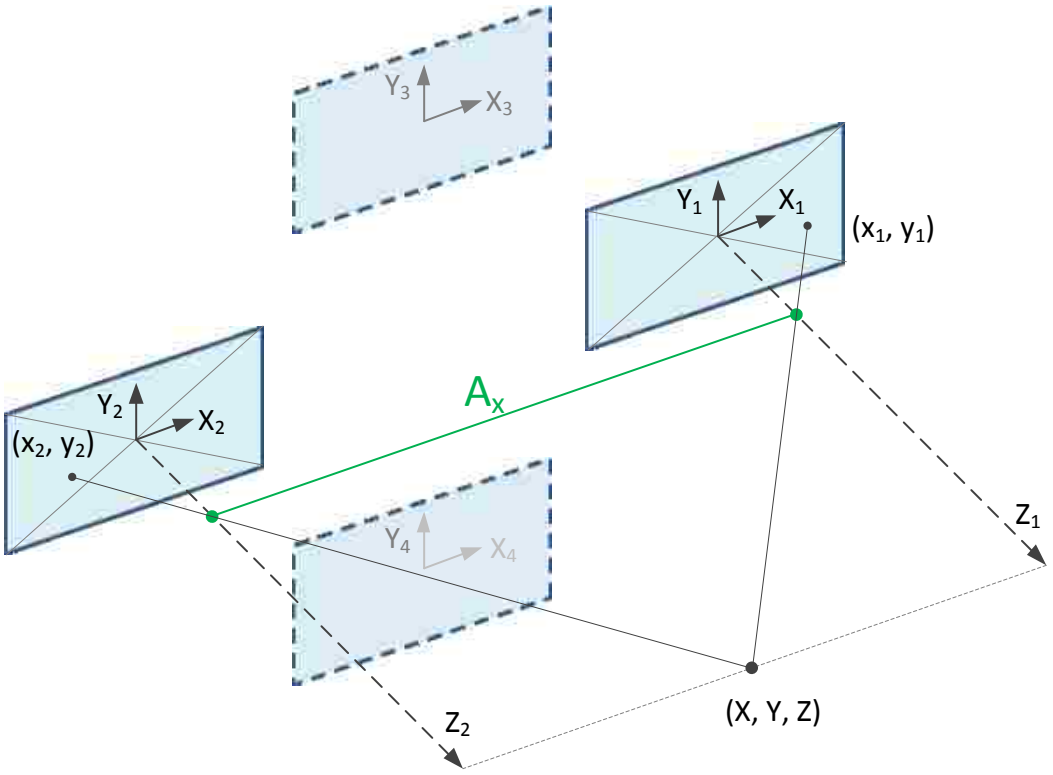


Figure 4.5 Perspective projection of camera 1 and camera 2

From (4.12)-(4.11) we obtain

$$A_x = \frac{x_2 - x_1}{f}(f - Z) \tag{4.13}$$

or

$$Z = f(1 - \frac{A_x}{x_2 - x_1}) \tag{4.14}$$

Once we compute Z we substitute it to

$$X_1 = \frac{x_1}{f}(f - Z) \tag{4.15}$$

and

$$Y_1 = \frac{y_1}{f}(f - Z) \tag{4.16}$$

and we compute the coordinates $P(X_1, Y_1, Z)$ with respect to camera 1 coordinate system.

$$X_2 = X_1 + A_x = \frac{x_2}{f}(f - Z) \tag{4.17}$$

and

$$Y_2 = \frac{y_2}{f}(f - Z) \tag{4.18}$$

computes the coordinates of point P with respect to camera 2 coordinate system.

Now we consider the two cameras along the vertical directions, which are identical to the other two cameras, and the distance between the centers of their lenses is A_y, as shown in Figure 4.6.

First we consider camera 3, then the coordinates of P with respect to the coordinate system of camera 3 is $P(X_3, Y_3, Z)$, where the Z is the same as before but we will get a new estimate of the Z as follows:

$$Y_3 = \frac{y_3}{f}(f - Z) \tag{4.19}$$

and

$$X_3 = \frac{x_3}{f}(f - Z) \tag{4.20}$$

Next we consider the point $P(X_4, Y_4, Z)$ with respect to the camera 4 coordinate system. Then

$$X_4 = \frac{x_4}{f}(f - Z) \tag{4.21}$$

and

$$Y_3 + A_y = Y_4 = \frac{y_4}{f}(f - Z) \tag{4.22}$$

Thus from (4.22)-(4.19) we obtain

$$A_y = \frac{y_4 - y_3}{f}(f - Z) \tag{4.23}$$

This gives us a new estimate of Z which is

$$Z = f(1 - \frac{A_y}{y_4 - y_3}) \tag{4.24}$$

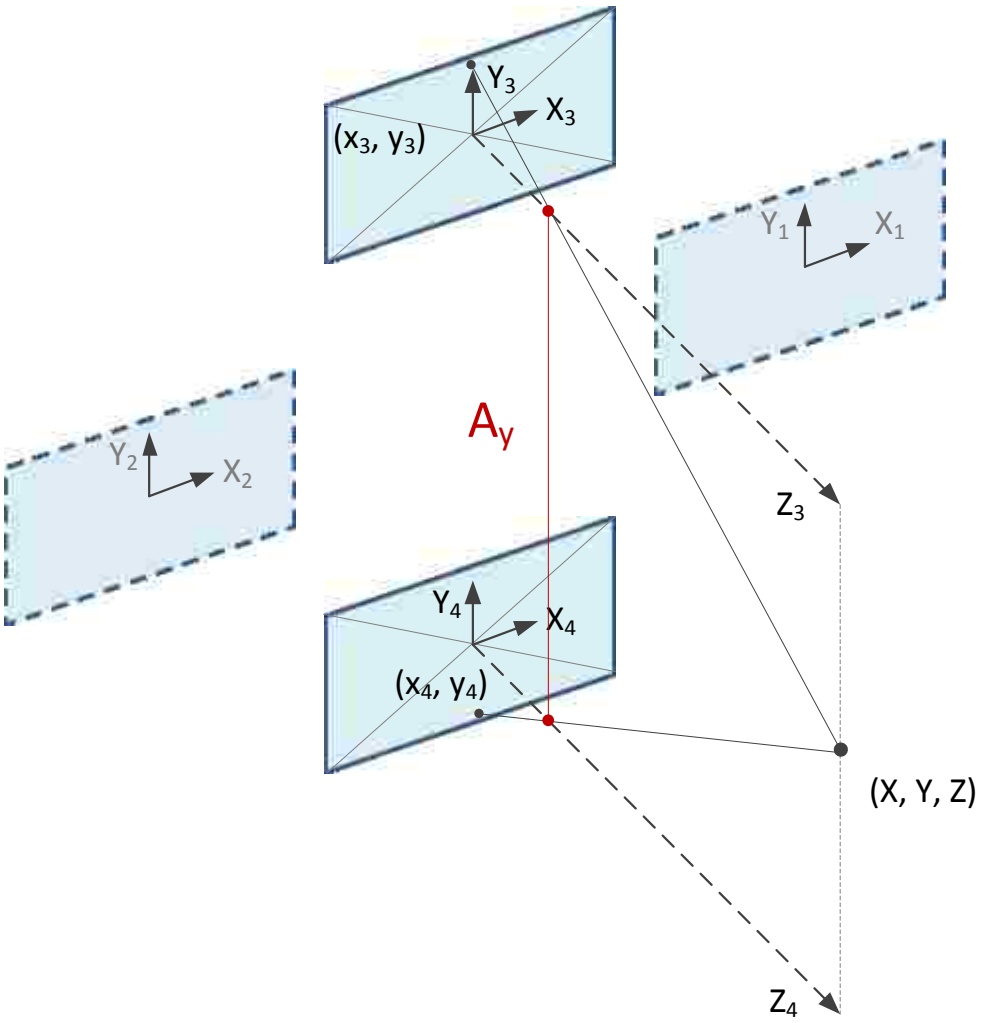


Figure 4.6 Perspective projection of camera 3 and camera 4

Now consider cameras 1 and 3, Figure 4.7.

The horizontal distance of their lenses is $\frac{A_x}{2}$ and the vertical distance is $\frac{A_y}{2}$.

Then we have

$$X_3 = X_1 + \frac{A_x}{2} \tag{4.25}$$

and

$$Y_1 = Y_3 + \frac{A_y}{2} \tag{4.26}$$

From (4.25) and (4.20)-(4.11) we obtain

$$X_3 - X_1 = \frac{A_x}{2} = \frac{x_3 - x_1}{f}(f - Z) \tag{4.27}$$

this implies

$$Z = f\left(1 - \frac{A_x}{2(x_3 - x_1)}\right) \tag{4.28}$$

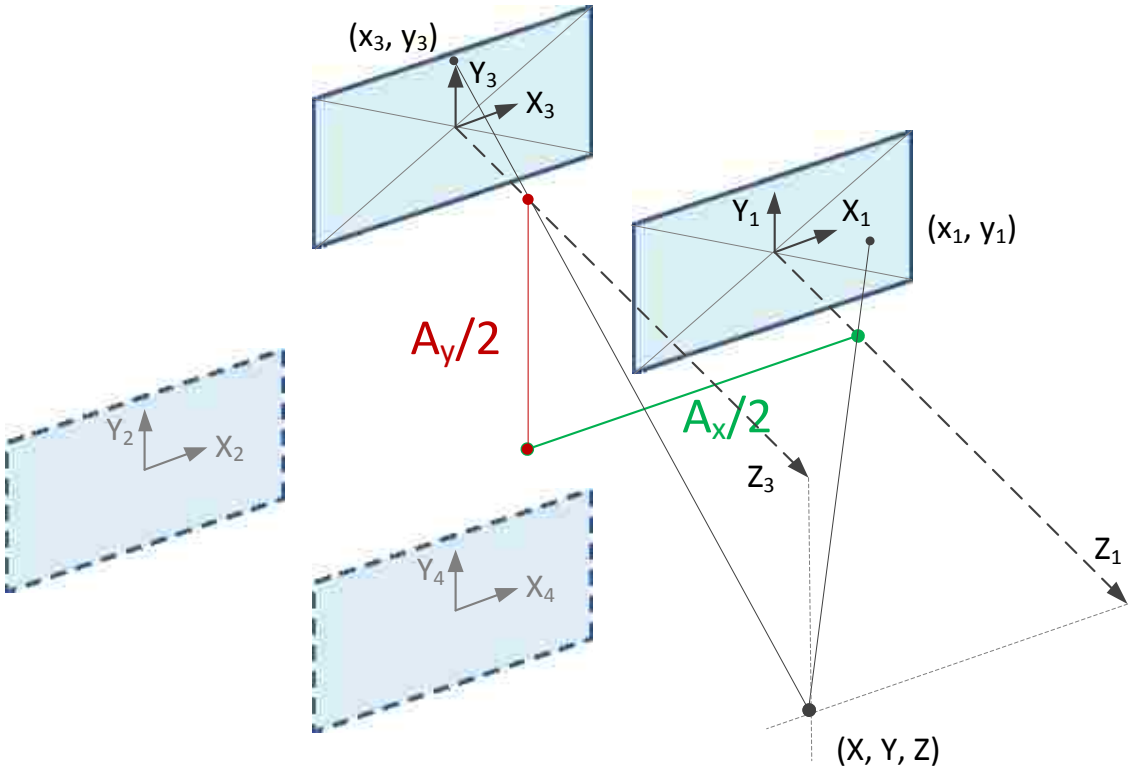


Figure 4.7 Perspective projection of camera 1 and camera 3

From equations (4.26) and (4.16)-(4.19) we obtain another estimate of Z

$$Z = f(1 - \frac{A_y}{2(y_1 - y_3)}) \tag{4.29}$$

Now if we consider cameras 2 and 3, Figure 4.8, then

$$X_2 = X_3 + \frac{A_x}{2} \tag{4.30}$$

and

$$Y_2 = Y_3 + \frac{A_y}{2} \tag{4.31}$$

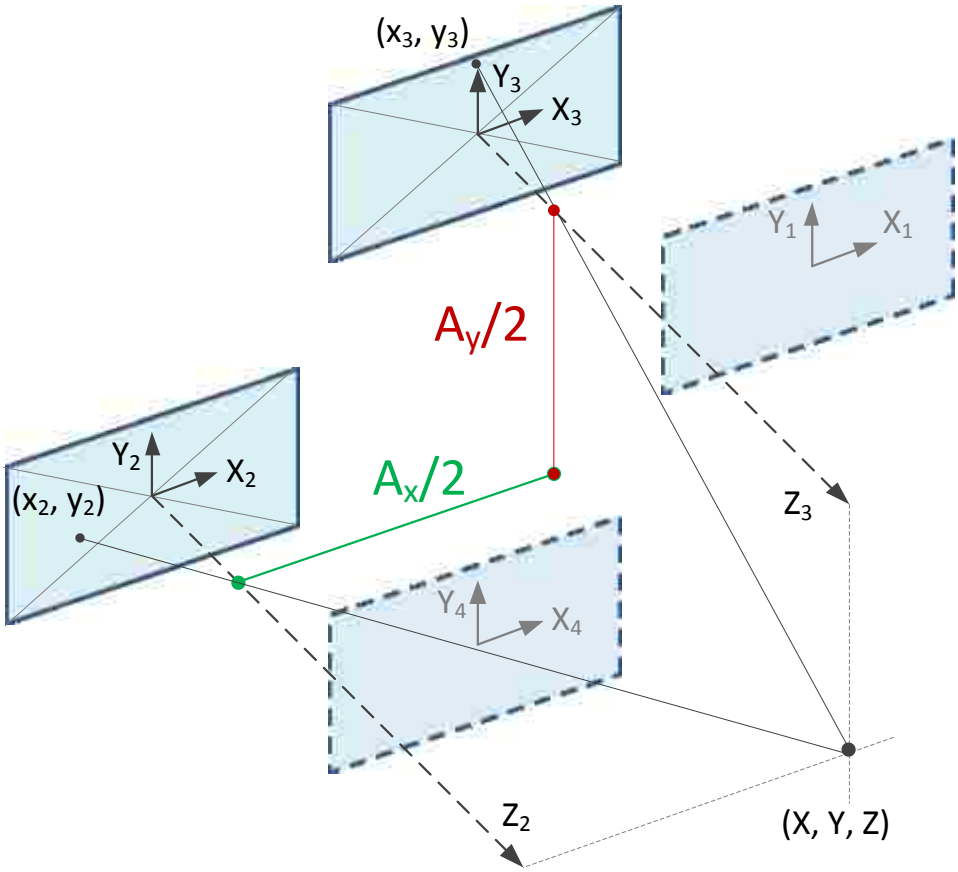


Figure 4.8 Perspective projection of camera 2 and camera 3

From these equations we obtain another two estimates of Z. Thus from (4.30) and (4.17)-(4.20) we obtain:

$$Z = f\left(1 - \frac{A_x}{2(x_2 - x_3)}\right) \tag{4.32}$$

and from (4.31) and (4.18)-(4.19) we obtain

$$Z = f\left(1 - \frac{A_y}{2(y_2 - y_3)}\right) \tag{4.33}$$

Similarly if we consider cameras 2 and 4, Figure 4.9, then we obtain

$$X_2 = X_4 + \frac{A_x}{2} \tag{4.34}$$

and

$$Y_4 = Y_2 + \frac{A_y}{2} \tag{4.35}$$

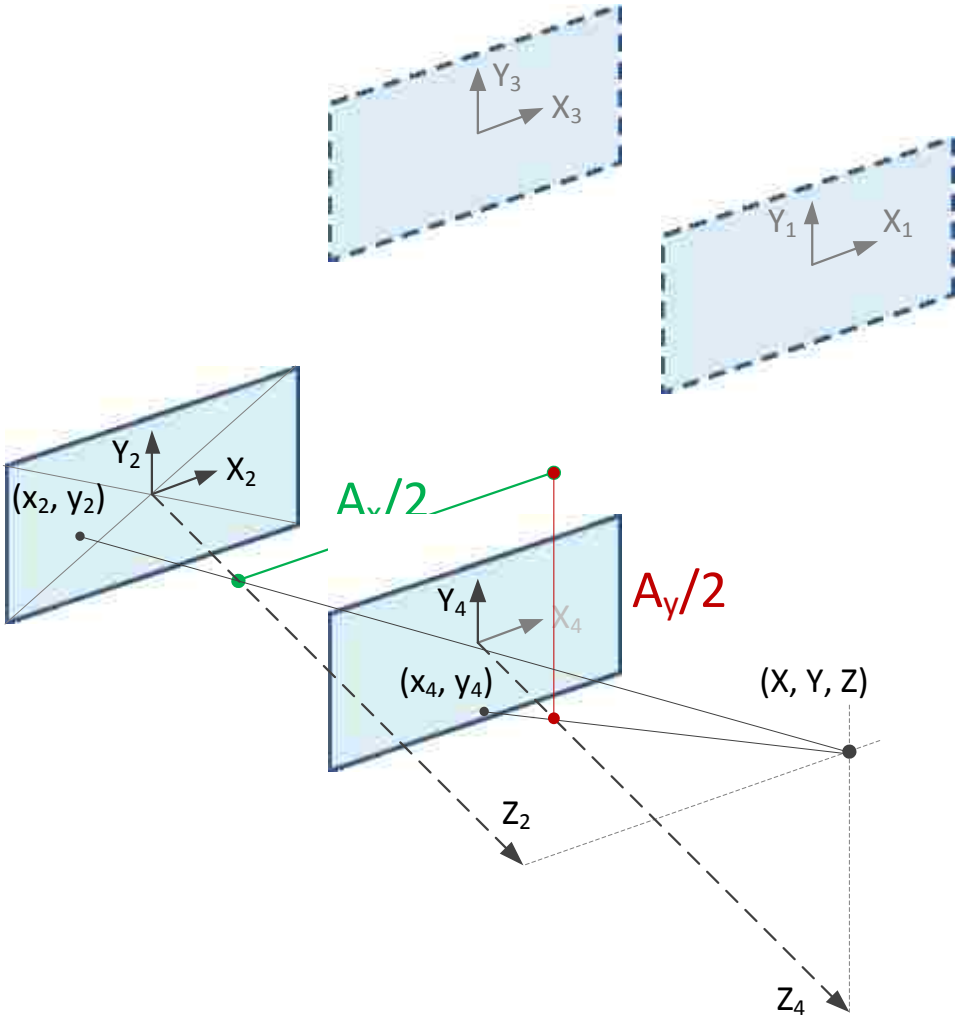


Figure 4.9 Perspective projection of camera 2 and camera 4

From equations (4.34) and (4.17)-(4.21) we obtain

$$Z = f(1 - \frac{A_x}{2(x_2 - x_4)}) \tag{4.36}$$

and from equations (4.35) and (4.22)-(4.18) we obtain

$$Z = f(1 - \frac{A_y}{2(y_4 - y_2)}) \tag{4.37}$$

Finally if we consider cameras 1, and 4, Figure 4.10, then we obtain

$$X_4 = X_1 + \frac{A_x}{2} \tag{4.38}$$

and

$$Y_4 = Y_1 + \frac{A_y}{2} \tag{4.39}$$

From equations (4.38) and (4.21)-(4.11) we obtain

$$Z = f(1 - \frac{A_x}{2(x_4 - x_1)}) \tag{4.40}$$

and from equations (4.39) and (4.22)-(4.16) we obtain

$$Z = f(1 - \frac{A_y}{2(y_4 - y_1)}) \tag{4.41}$$

So far we obtained 10 estimates of the coordinate Z. Z is an important coordinate because every other coordinate is a function of Z. Each one of these 10 estimates of Z is different due to the noise in the system. In the subsequent text we will denote these 10 estimates of Z by $\widehat{Z_1}, \widehat{Z_2}, \widehat{Z_3}, \ldots \widehat{Z_{10}}$. Thus

$$\hat{Z}_i = Z + \varepsilon_i \tag{4.42}$$

where ε_i represents the distance between the estimated value and the true value Z. In the subsequent we use mean square error to find an optimal estimate of Z.

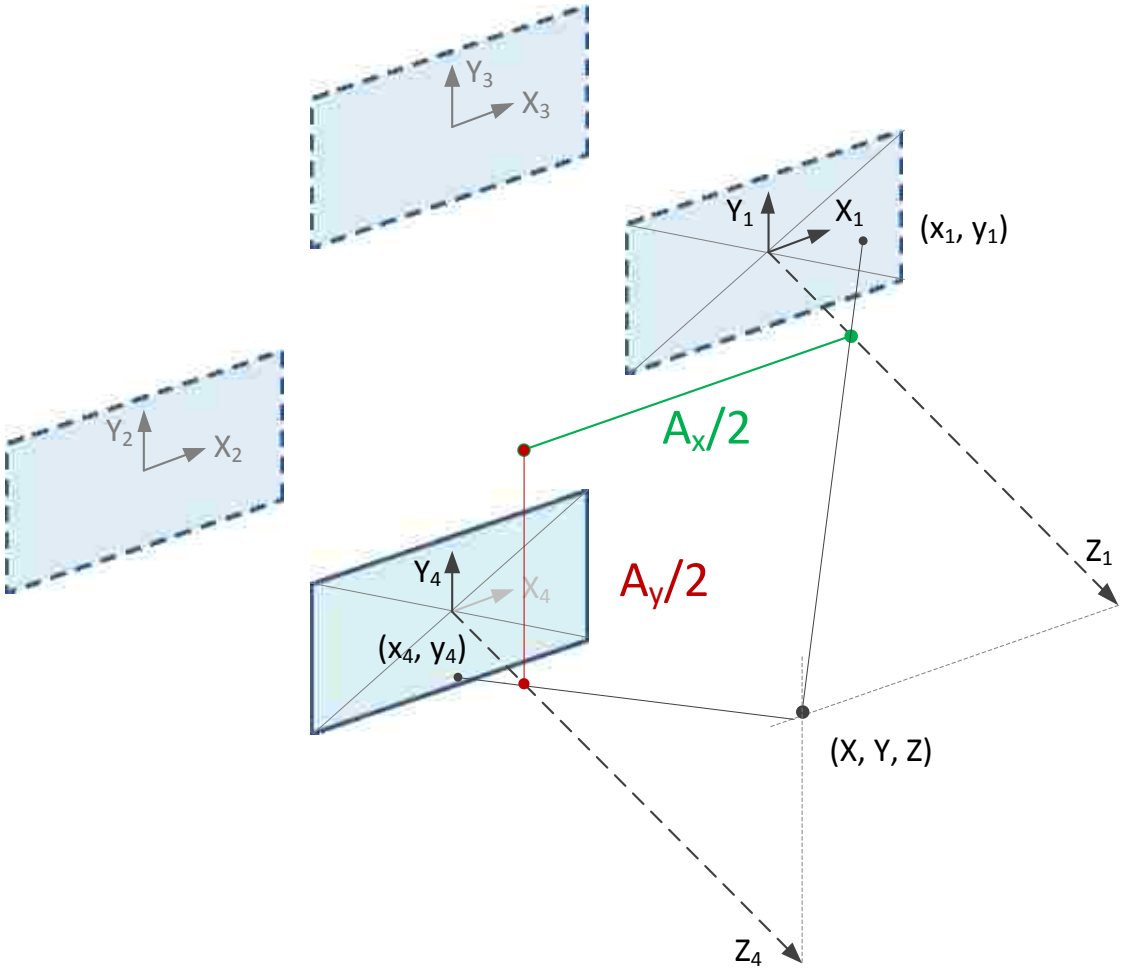


Figure 4.10 Perspective projection of camera 1 and camera 4

From equation (4.42) we obtain

$$\varepsilon_i = \hat{Z}_i - Z \tag{4.43}$$

and therefore

$$f(Z) = \sum_{i=1}^{10} \varepsilon_i^2 = \sum_{i=1}^{10} (\hat{Z}_i - Z)^2 \tag{4.44}$$

the idea here is to find the Z that minimizes the square error.

$$\frac{\partial f(Z)}{\partial Z} = -2 \sum_{i=1}^{10} (\hat{Z}_i - Z) = 0 \tag{4.45}$$

This implies

$$\sum_{i=1}^{10} \hat{Z}_i = 10\, Z \tag{4.46}$$

therefore

$$Z = \frac{\sum_{i=1}^{10} \hat{z}_i}{10} \tag{4.47}$$

This is the best estimate of Z to be used in computing the X, Y coordinates for each one of the camera systems. The next thing is to convert each point in the global coordinate to the individual camera coordinate system then performs the perspective projection. For this we have to follow the following steps. 1st) Translate the origin to the cross gimbal, by (X_g, Y_g, Z_g), which represents the position of the system gimbal with respect to the global coordinate system. 2nd) Rotate about the Z-axis by θ, and about the X-axis by an angle ϕ. 3rd) Translate the new coordinate system by $(X_{cig}, Y_{cig}, Z_{cig})$, so that its new position is in camera i, i=1,2,3,4, gimbal. 4th) Rotate about the new Z-axis by θ_{ci}, and about the X-axis by an angle φ_{ci}. 5th) Translate the new coordinate system from the camera i gimbal to the center of the image plane of camera i by (X_{ci}, Y_{ci}, Z_{ci}). 6th) Perform the perspective projection of the point used as center of projection, the point (0,0,f). We can go back to the global coordinate system by using the inverse transformations in reverse. The matrix associated with the first transformation is:

$$G = \begin{bmatrix} 1 & 0 & 0 & -X_g \\ 0 & 1 & 0 & -Y_g \\ 0 & 0 & 1 & -Z_g \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{4.48}$$

This transformation places the origin at the point with coordinates (X_g, Y_g, Z_g). Thus the new coordinates of the point P(X,Y,Z) are G*P.

Step 2 combines rotation by θ about the Z axis, and by ϕ about the X axis. The transformation matrix is:

$$R = \begin{bmatrix} cos\theta & sin\,\theta & 0 & 0 \\ -sin\theta cos\varphi & cos\theta cos\varphi & sin\varphi & 0 \\ sin\theta sin\varphi & -cos\theta sin\varphi & cos\varphi & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{4.49}$$

The third step is materialized by using the matrix

$$G_{ci} = \begin{bmatrix} 1 & 0 & 0 & -X_{cig} \\ 0 & 1 & 0 & -Y_{cig} \\ 0 & 0 & 1 & -Z_{cig} \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{4.50}$$

The fourth step is given by the matrix:

$$R_{ci} = \begin{bmatrix} cos\theta_{ci} & sin\theta_{ci} & 0 & 0 \\ -sin\theta_{ci} cos\varphi_{ci} & cos\theta_{ci} cos\varphi_{ci} & sin\varphi_{ci} & 0 \\ sin\theta_{ci} sin\varphi_{ci} & -cos\theta_{ci} sin\varphi_{ci} & cos\varphi_{ci} & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{4.51}$$

The fifth step is given by the matrix:

$$C_{ci} = \begin{bmatrix} 1 & 0 & 0 & -X_{ci} \\ 0 & 1 & 0 & -Y_{ci} \\ 0 & 0 & 1 & -Z_{ci} \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{4.52}$$

The last step which is the perspective projection matrix for homogeneous coordinates is:

$$T = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & -\frac{1}{f} & 1 \end{bmatrix} \tag{4.53}$$

In order to represent the perspective projection in matrix form it is best if we use homogenous coordinates for the world coordinates instead of Cartesian

coordinates. The homogenous coordinates of the point P(X, Y, Z) with constant k=1 are:

$$P_h = \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} \tag{4.54}$$

Thus after performing the above six steps we obtain:

$$C_{hci} = \begin{bmatrix} x_{hci} \\ y_{hci} \\ z_{hci} \\ c_{kci} \end{bmatrix} = TC_{ci}R_{ci}G_{ci}RGP_h \tag{4.55}$$

The image plane x, and y, Cartesian coordinates for camera- i are:

$$x_{ci} = \frac{x_{hci}}{c_{kci}} \tag{4.56}$$

and

$$y_{ci} = \frac{y_{hci}}{c_{kci}} \tag{4.57}$$

III. Applications

The theory presented here can be applied to any object with regular or irregular geometry. Any point on the object can be easily located on the four cameras by using the dot mode of the non-invasive laser pointer to confirm the correspondence and solving using the math presented above. Unlike the epipolar geometry approach, no computational effort is needed to determine the corresponding points. In case of objects with regular geometry, reference points can be found in the final extracted polygon, e.g. corners on Figure 3.17 (c). These

3D coordinates of these points can be compared to a library of all possible regular geometries to determine the part geometry, orientation along with its dimensions. In case of irregular objects, the grid of squares mode of the non-invasive laser pointer will be used to find the 3-D coordinates of all points where the grid is projected at. By scanning more points of the surface of this irregular object using the projection of grid of squares, the total surface and volume can be estimated. We have applied similar theory and robot vision system in our NASA robot vision system [53], [54], [55], [56], [57]. We are in the process of applying it to measure the dimensions and volume of regularly and irregularly shaped luggage in order to place them in containers to be loaded on airplanes as cargo and optimize the weight distribution with respect to the center of gravity of the airplane. Thus as luggage are placed on the scale, the proposed system, with the help of the laser square grid and the four cameras, divides the luggage into small sections and computes the volume of each small section. After summing the small volumes up it computes the luggage volume. Having the volume and the weight, an optimal placement algorithm that places each luggage to a container will be applied. The containers are numbered and placement algorithm optimizes the weight distribution by placing the containers appropriately in the cargo area. Thus the luggage is given a number, which is placed in the data base, along with the passengers name, the container number, the weight and volume.

In case of an emergency, the luggage can be located very fast and removed from the airplane if needed. The other application that is being under experiment at the present time with this robot vision system is telemedicine. In the state of Nevada, there are many rural areas with small hospitals where specialists visit once every one week or once in a while when there is demand. Yet the specialist needs to follow up with the progress of the patient. With the help of the nurse, the robotic system calculates the progress once or twice a day, by computing the volume of the area affected by the surgery and also the amount of reduction or increase since the last time. The robot transmits this information wirelessly to the server in the hospital and from there it's transmitted either in real time or in a store and forward fashion to the physician.

IV. Conclusions

A robot vision system consisting of four identical cameras and a non-invasive laser operating in two different modes was described. In one mode, the laser transmits a dot, and in the other mode it transmits a grid of squares. The cross is supported by a mechanism that allows the system of four cameras to pan and tilt. Each camera is supported by its own pan and tilt mechanism. We present the Mathematics and analytics of our computer vision system in this research as well as some of the applications.

CHAPTER 5

A CAMERA SYSTEM FOR DETECTING DUST AND OTHER DEPOSITS ON SOLAR PANELS (*) [58]

Abstract

Solar panels over time, due to winds, sandstorms, bird droppings, suffer from dust, and other deposits. As a result of these deposits the sunlight is refracted, and only part of the sunlight reaches the chips inside the glass cage that are generating the electricity. In a remote area with thousands of solar panels, it is both expensive and cumbersome to send maintenance people to inspect each panel and clean it if needed. A system of smart cameras with R,G,B, and infrared for night vision, that take the picture of each panel continuously is considered. The picture becomes input to the proposed classification algorithm that decides in the real time if the panel needs cleaning or not. The classification algorithm consists of: the classification vector, the metric used, the training of the classifier, the testing of the classifier, and the classifier put into play for everyday use. At the present time a commercial camera transmitting JPEG frames wireless to the server where the classification and storage takes place is used. But in the near future, the classification algorithm will reside on a flash memory which will

* - This chapter was submitted to the 7th ACM/IEEE International Conference on Distributed Smart Cameras (ICDSC13) [58].

be part of a circuit board that is being currently designed. The algorithm operates on the incoming data and will be executed by an ARM processor which will also be on the board. The circuit board also will include a CCD and Infrared camera. The hardware and software on this electronic board will be designed and programmed by the author. Once the proposed intelligent system detects that the panel needs cleaning, it will automatically trigger a mechanism which will clean the panel.

I. Introduction

In a large scale solar energy array of panels built by a variety of solar panel manufacturers, each panel having a different area, and based on different type of chip architectures, the energy output is not uniform and there could be a considerable panel to panel variation. The energy output of each panel depends on the time of the day, day of the month, month of the year, year in the solar cycle, and of course the climatological conditions during the time of the day that energy measurements are obtained. In arid climates it rains very rarely so the ground is dry and therefore when the wind blows the loose ground is carried by the wind and dust storms are very common. As the dust settles on the glass of the various panels the energy output falls because the dust prevents part of the light incident to the glass to enter and reach the chips generating the energy.

Thus, part of the light is refracted and less light goes through the glass and into chip structure designed to convert the solar energy to electricity. The loss of light energy depends on the amount of dust, the size of particulates, and the chemical composition of dust [59], [60]. Loss of solar energy is not just due to dust but also due to bird droppings and other factors that are yet to be explained. In arid climates there are not as many trees as in climates with normal amount of rain. As the birds migrate from north to warmer climates during the fall, or from warm to cooler climates during the spring, solar farms are used by the migrating birds as rest areas, and therefore get many bird dropping that reduce the electric energy output. Also during the regular season solar panels get a fair amount of bird droppings. In arid climates that often times the summers are too hot, and the winters nights are cold, solar panels provide shade in the summer and protection from the cold and the wind in the winter. Thus many small animals find rescue from the difficult climate conditions and make home under the panels. Often times some of these animals try to eat the cables attached to the panels and in the process damage the cable to the point that stops the flow of electricity from the panel to the destination. In addition to the animals and birds, the protections the panels provide from the harsh weather conditions promote the growth of various desert plants that grow faster in the shaded ground underneath the panels. Often times these plants and trees grow enough to crack

the base of the panel, or lift the panel so that the orientation of the panel with respect to the sun is not what it should be, therefore the amount of light going through the glass and into the chip structure designed to convert the solar energy to electricity is less than the optimal amount. Often times these plants as they grow into trees they lift the panel and severely damage it as well as other neighboring panels that this panel falls on the top of. In addition to the above possibilities for damaging the panels and reducing the electric energy output there are others, such as vandalism, earthquakes, theft, and a number of natural catastrophes. For now the only purpose of our smart camera is to detect if a panel needs cleaning by applying the proposed classifier and then if it does need cleaning to trigger a mechanism to clean the panel. Inspecting a number of cleaning mechanisms including a container with water and two sponges one to wash the panel with the water and the other to dry, or a container with a cleaning fluid and two sponges, or a mechanism consisting of small spike-like needles to crate static electricity on the dirt of the panel and attract the dirt into a vacuum, it is concluded that the later is the best method and has many advantages over the previous mentioned methods. Some of the advantages of the static electricity mechanism with the vacuum are its very low maintenance. The mechanism with the water and the sponges need enormous amount of maintenance in an arid climate. The sponges dry out relatively fast in the

extreme climate conditions of the summer months that the temperature could rise well over 110 degrees Fahrenheit and could need to be replaced very often adding to the expense of maintenance of the solar panel site. Furthermore water is very scarce, and evaporates very fast as it is poured on the top of the panels. As the mechanism cleans the panel with water is very possible that it would create uneven cleaning and/or leave a film of material that acts as inhibitors for the panel to utilize the solar energy in an optimal way. Replacing the water in the containers of each solar panel could be expensive and challenging. It might have an initial expense to bring the water pipe into the site and then a continuous expense to maintain the containers and their sensors that allow the proper release of the water. If a cleaning fluid is used the challenges are similar to those of using the water, but the expense could be much higher, and that includes cleaning fluid replacement, sponge and/or wiper replacement, and man power needed for the proper maintenance. The static electricity mechanism to remove the dust is very light, gets a small amount of energy from the panel itself, is relatively simple to build, therefore could be very reliable, and needs very little maintenance [61]. In this research, the focus will be only on building a classifier to detect if the panel needs to be cleaned or not. The steps for building this classifier includes sampling, deciding which parameters the classification vector should include, train the classifier, compute the probability of misclassification

which includes the probability of the classifier deciding that the panel is dirty although it is not, or the panel is clean although it is not. After testing the classifier it is to be put into play on a real solar panel site. For now the classifier is computer software on the server where the camera transmits the video of the solar panels. In the near future the classifier will be embedded into a hardware board, which will include the visual camera chip, the infrared camera, and the alarm events that will be transmitted wirelessly and alarming the care takers of the site about broken glass, a malfunction in one of the panels, cut cables, or vandalisms. The classifier is based on the multivariate probability distribution function of the mode of the red channel, the mode of the green channel, and the mode of the blue channel. It also includes the marginal distribution function of the modes of the red, green, and blue channels. In clean panels the three dimensional vector of the mean vector of the modes has relatively low values, while as the panel gets dirty by dust particulates and bird droppings, the means of the mode vectors of the red, green, and blue channels increase. A Hotellings T-square test shows that the mean vectors of the modes between clean and dirty panels are significantly different. Also the Mahalanobis distance is capable of classifying correctly with high probability that a panel belongs to the clean or dirty category of panels. This chapter is structured as follows: First is the

Abstract, section I is the Introduction, section II is the Background Information, Section III is the Classification Algorithm, Section IV is the Conclusions.

II. Background Information

A. The color of the Sky

The sky color variations from shades of blue to shades of gray indicate higher levels of concentration of particulates larger than the clean air. The clean air is composed approximately of 78.084% Nitrogen, 20.94% Oxygen, 0.934% Argon, and 0.0350% other gases [62]. The color of the sky is explained by the Mie solution to Maxwell's equation, and Rayleigh's scatter approximation theory. According to Rayleigh's scatter theory, the small molecule size of Nitrogen and Oxygen molecules scatter the components of the light with shorter wavelength range and thus higher frequency. These light components in the visible spectrum are the blue and violet colors. The scattering of the blue colors therefore by Nitrogen and Oxygen explains the blue color of the sky. Larger molecules scatter color components with longer wavelength. In the visible light, the red color component of the white light has the larger wavelength, then the green color component has lower wavelength than the red, and the blue color component has lower wavelength than the green. When the concentration of larger molecules, such as pollutants, increases, the scattering of colors with longer

wavelength increases. The mix of the longer wavelength color components in the atmosphere with the blue, change the color of the sky, making it more grey, or light grey. Thus when there is high concentration of water vapor molecules that cling together in that part of the sky make it look white or gray. This is the reason that clouds look white or gray. When pollution levels increase, since their molecule size is usually relatively large, that triggers the scattering of higher wavelength color components, [63]. The color of the sky is a good indicator of how clean the air is. Lighter grey colors signify the presence of water vapor, sulfur, aerosols, and soot in the atmosphere. In any given direction, skylight is light scattered by all the molecules and particles in its path from the sun, which is the source of daylight in the Earth. Since the atmosphere on the Earth is composed mostly of Nitrogen and Oxygen molecules, then these are the most frequent molecules that light will encounter in its trajectory. Electromagnetic waves with shorter wavelengths are affected by particles much smaller than their wavelength, scattering these wave in different directions. Waves with larger wavelength are not being affected by these small particles. As a result the blue and violet waves of the white light are scattered while traveling in the clean atmosphere turning the sky blue [64], [65], [66]. The lower intensity of the violet color compared to other colors in the light coming from the sun, combined with the lower sensitivity of this color by the human eyes is the main reason why the

sky is blue and not violet even though the violet wavelength is smaller than the blue and scatters even more. So there is very little scattering of the red color due to its relatively large wavelength. The red color travels straight thru the atmosphere. Similarly there is very little scattering for the green color because although its wavelength is smaller than that of red is still relatively large compared to the size of Nitrogen and Oxygen. So the green light travels straight thru a clean atmosphere.

B. Light and Electric Energy

The infrared and red wavelengths having the larger wavelengths compared to the other components of the white light incident to the glass of the solar panel penetrates the glass and produces the most amount of electricity. The blue and violet wavelengths suffer the most absorption by the glass of the panel and contribute very little in the production of electric energy. As it was pointed out in the previous section when the sky is cloudy due to water vapor, the red and green components of the light suffer a great deal of scattering and do not go straight thru the atmosphere. The infrared wavelengths are the ones that produce more energy following by the red, green, and blue. The energy produced by the red and green wavelengths is substantially reduced during cloudy days. Also when dust particulates, or other pollutants are in the atmosphere they cause the

scattering of the larger wavelength components of light thus the energy produced by the solar panels is reduced. Dust storms are very common in arid climates, especially during the summer months when the ground is very dry and the top soil is sandy. Sandstorms are responsible for depositing dust and sand on the solar panels. As the sunlight passes thru these particulates, the larger wavelengths that normally are the larger energy producers, scatter, and as a result of this scattering less light energy is reaching the chips producing the energy and therefore less energy is produced. Birds flying over the solar panel area, or stopping on solar panels for a rest or out of curiosity, often times they deposit their droppings. Bird droppings usually form a thick coating on an area of the panel preventing the sun light in that area from going thru. It is very difficult for one to change the weather or the pollution in the air. It is not difficult to detect the dust and sand on the glass and clean it. As we mentioned before, our approach is not to use water or other liquid for cleaning, but to use a small vacuum cleaner flying over the panel in a methodical way that goes thru in a raster type of way and visits every point of the panel from top to button twice. During the first pass it vacuums all the gravel, dust, and droppings it can. During The second pass it activates the needles that have static current, thus charging the dust and droppings on the glass with opposite static current. Thus our system visits every point on the glass in a raster scanning fashion, the

needles attract the remaining deposits on the glass, and those deposits go thru the vacuum and get deposited thru a pipe system on the ground by the base of panel. This is a relatively simple system, draws the energy from the panel itself, does a very good job in cleaning, does not leave any film on the glass, has a relatively small manufacturing and installation cost, it needs very little maintenance. It is very easy to train a technician to repair or replace the system.

III. Classification Algorithm

As we examine an image of a clean panel, Figure 5.1, and compute the probability density functions of the red channel, the green channel, and the blue channel of the pixels of the image, Figure 5.3, we observe that each one of the marginal probability density functions for each one of the channels is skewed to the left with a maximum (mode) to the left of the 127.5 value. Here we assume that the scale of the red channel of the pixels is between 0-255, similarly the scales of the green and blue channels of the pixels are also between 0-255. Thus in the graphs of the probability density function of the color channels the scale on the x-axis is between 0-255. Figure 5.3(a) gives the univariate probability distribution functions of the red, green, and blue, channels, for a clean panel. The mode for the red channel is 56, for the green is 60, and for the blue is 84. The x-axis for each one of the channels is from 0-255. Figure 5.3(b) gives the univariate probability

distribution functions of the red, green, and blue channels, for a dusted panel, Figure 5.2. The mode for the red channel is 79, for the green is 82, and for the blue is 105. Based on a sample of 50 pictures of dirty panels and fifty of clean ones, the gamma distribution provides enough flexibility to model these probability functions. The gamma density function is of the form [67], [68]:

$$f(x) = \frac{x^{\alpha-1}e^{-\frac{x}{\beta}}}{\Gamma(\alpha)\beta^{\alpha}}, \; x > 0, \; \alpha, \beta > 0 \tag{5.1}$$

Theorem 1, gives an estimate of the global mode of the gamma density function in terms of its parameters α, β, and also estimates the parameters from the data.

Theorem 1. The global mode of the gamma density function (the point x that maximizes the density function f(x) is $x_m = (\alpha - 1)\beta$, if $\hat{x}$ is the global mode based on the empirical density function computed from the data, and $\bar{x}$ is the average then an estimate of the parameters α, β, from the data is: $\hat{\alpha} = \frac{\bar{x}}{\bar{x}-\hat{x}}$ and $\hat{\beta} = \bar{x} - \hat{x}$

Proof

The global mode x_m of the gamma probability function is the point that maximizes the density function. Namely the point for which the first derivative of the function f(x) is zero and the second derivative is negative.

$$f'(x) = e^{-\frac{x}{\beta}} \; \frac{(\alpha-1)x^{\alpha-2} - \frac{x^{\alpha-1}}{\beta}}{\Gamma(\alpha)\beta^{\alpha}} = 0 \tag{5.2}$$

The solution to the above equation is

$$x_m = (\alpha - 1)\beta \tag{5.3}$$

$$f''(x) = -\frac{1}{\beta}e^{-\frac{x}{\beta}}\frac{(\alpha-1)x^{\alpha-2} - \frac{x^{\alpha-1}}{\beta}}{\Gamma(\alpha)\beta^{\alpha}} + e^{-\frac{x}{\beta}}\frac{(\alpha-1)(\alpha-2)x^{\alpha-3} - \frac{(\alpha-1)x^{\alpha-2}}{\beta}}{\Gamma(\alpha)\beta^{\alpha}} \tag{5.4}$$

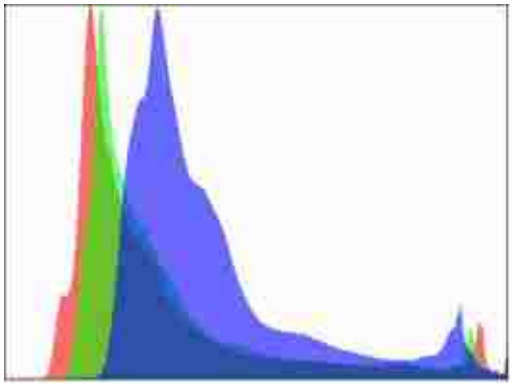
For the type of gamma probability function that fits this data $\alpha > 2$ always.

Thus

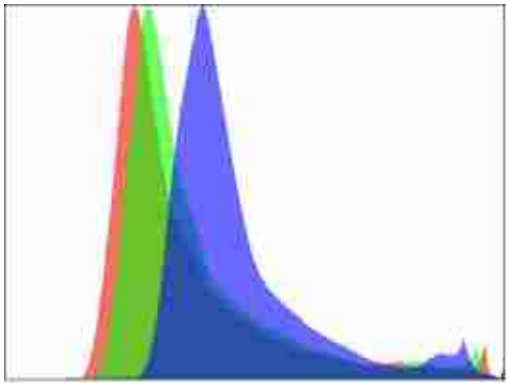
$$f''(x_m) = -e^{-(\alpha-1)}\frac{\left((\alpha-1)\beta\right)^{\alpha-3}}{\Gamma(\alpha)\beta^{\alpha}} < 0 \tag{5.5}$$

Figure 5.1 Image of a clean panel

Figure 5.2 Image of a dusty panel

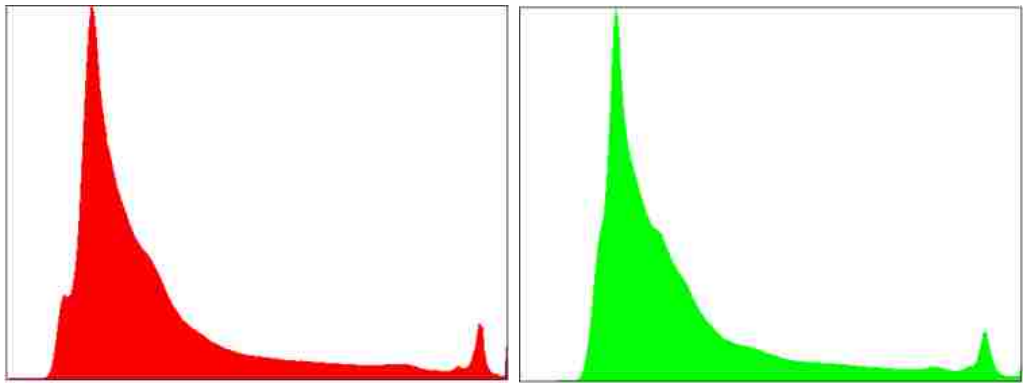


(a) Clean Panel



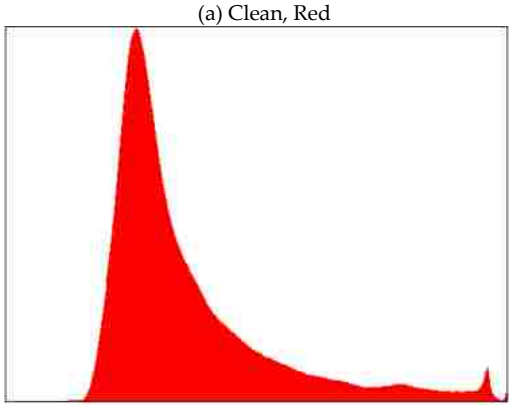
(b) Dusty Panel

Figure 5.3 Probability density function for the R, G, B channels, Observe the shift of the red, green, and blue modes

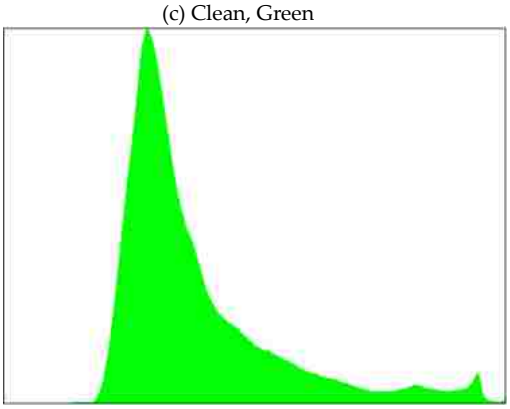


(a) Clean, Red

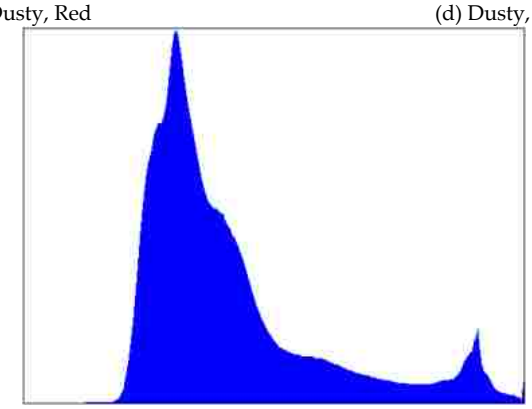
(c) Clean, Green



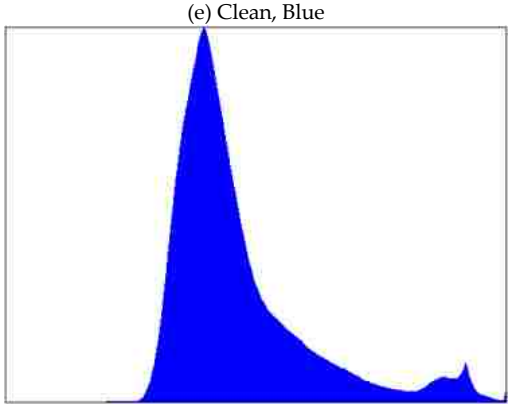
(b) Dusty, Red



(d) Dusty, Green



(e) Clean, Blue



(f) Dusty, Blue

Figure 5.4 Individual probability density functions for R, G, B channels of both clean and dusty images

The point $x_m = (\alpha - 1)\beta$ is the global maximum for f(x), thus it is the mode of the density function. It is very easy to prove that the mean or expected value of a gamma is $\mu=\alpha\beta$. Given a panel image we can use the pixels to extract the values of the red, green, and blue channels. For each one of these channels we can estimate the mean, and the maximum mode. The mean is the average of all the values for that channel, and the maximum mode of course is the point between 0 and 255 for which most of the values of that channel for the picture that the channel belongs to, are equal to. Let the sample mean for the arbitrary channel be denoted by $\bar{x}$ and let the estimate of the maximum mode be denoted by $\hat{x}$. Then the parameters α, β, of the gamma can be estimated by setting:

$$\bar{x} = \alpha\beta \tag{5.6}$$

and

$$\hat{x} = (\alpha-1)\beta \tag{5.7}$$

From the above two equations we obtain

$$\bar{x} - \beta = \hat{x} \tag{5.8}$$

which implies that:

$$\hat{\beta} = \bar{x} - \hat{x} \tag{5.9}$$

From equation 5.6 therefore we obtain that

$$\hat{\alpha} = \frac{\bar{x}}{\bar{x}-\hat{x}} \tag{5.10}$$

QED.

Formulas 5.9 and 5.10 give us estimates of the parameters α and β which they define the gamma density function for the channel based on the channel data. We took the picture of each one of the fifty panels when they were dusty, then we cleaned them and we took their picture again. Thus we obtained 50 clean panel pictures and fifty dusty under the same conditions. For each one of the pictures we computed the probability density function of the red, green, and blue, channels. As we graphed each one of the probability density functions the x-axis denotes all the possible values a color-channel can have. Thus the values for each one of the three channels vary between 0-255. The value in the x-axis for which the experimental density function obtains its global maximum is the mode. For the clean pictures we obtained 50 mode data for the red channel, 50 mode data for the green, and 50 mode data for the blue. In each one of the pictures the mode for the blue is over 20 units higher than the mode for the green and the mode for the green is about 4 units higher than the mode for red. As the panels get dusty then the modes are shifted to higher values, and the electric energy of the panel is reduced. There is a tradeoff between the energy needed for the vacuum to clean the panel and the energy gained. There is a cutoff point beyond which not cleaning the panel significant energy is lost and therefore we need to detect that the panel has reached that cutoff point and trigger the mechanism automatically to clean the panel. The classification theory that

enables us to make this decision is based on the following statistics. Consider the 50 clean panels. Let $(r_{m1}, r_{m2}, \dots, r_{m50})$, $(g_{m1}, g_{m2}, \dots, g_{m50})$ and $(b_{m1}, b_{m2}, \dots, b_{m50})$, be the 50 computed mode values for each one of the clean channels. Let $\bar{r}_m, \bar{g}_m, \bar{b}_m$, denote the average of the red-modes, green-modes, and blue-modes respectively. Then according to the central limit theorem the probability distribution function, of the vector $\bar{r}_m, \bar{g}_m, \bar{b}_m$ approaches to a trivariate normal. The random vectors (r_{m1}, g_{m1}, b_{m1}), (r_{m2}, g_{m2}, b_{m2}), ..., $(r_{m50}, g_{m50}, b_{m50})$, are independent since each panel is autonomous and independent of the other panels, and identically distributed, with a mean vector $\underline{\mu}' = (\mu_{rm}, \mu_{gm}, \mu_{bm})$, where the $\underline{\mu}'$ signifies the transpose vector, and variance- covariance matrix

$$\Sigma = \begin{bmatrix} \sigma_{rm}^2 & \sigma_{rmgm} & \sigma_{rmbm} \\ \sigma_{rmgm} & \sigma_{gm}^2 & \sigma_{gmbm} \\ \sigma_{rmbm} & \sigma_{gmbm} & \sigma_{bm}^2 \end{bmatrix} \tag{5.11}$$

The expected value $E(\bar{r}_m, \bar{g}_m, \bar{b}_m) = \underline{\mu}' = (\mu_{rm}, \mu_{gm}, \mu_{bm})$, and the variance covariance matrix of the vector $\bar{r}_m, \bar{g}_m, \bar{b}_m$ is $\frac{\Sigma}{50}$. In general if n-pictures are used to compute the vector $\bar{r}_m, \bar{g}_m, \bar{b}_m$, then the variance-covariance vector of $\bar{r}_m, \bar{g}_m, \bar{b}_m$, would be $\frac{\Sigma}{n}$. The vector $\underline{\mu}' = (\mu_{rm}, \mu_{gm}, \mu_{bm})$ is estimated from the data. Thus if a sample of n clean panels is used and we denote the trivare vector of the mode of the ith panel as $x_{mi} = (r_{mi}, g_{mi}, b_{mi})$ then

$$\bar{x}_m = \frac{x_{m1} + x_{m2} \dots + x_{mn}}{n} \tag{5.12}$$

And the variance covariance matrix Σ can be estimated from the data as follows:

$$\hat{\Sigma} = \frac{1}{n-1}\sum_{i=1}^{n}(x_{mi} - \bar{x}_m)(x_{mi} - \bar{x}_m)' \quad (5.13)$$

Theorem 2 creates the criterion for classifying a panel as being clean or needing cleaning.

Theorem 2. Let x denote the vector of r,g,b, modes of a new panel and $t^2 = \frac{n}{n+1}(x - \bar{x}_m)'\hat{\Sigma}^{-1}(x - \bar{x}_m)$ then $\frac{n-3}{3(n-1)}t^2 \sim F_{3,n-3}$ is an F distribution with parameters 3, n-3. If $x'\hat{\Sigma}^{-1}x < \bar{x}_m{}'\hat{\Sigma}^{-1}\bar{x}_m + \frac{3(n^2-1)}{n(n-3)}F_{3,n-3,\varsigma}$, then the panel is classified as clean, otherwise is needed to be cleaned. $F_{3,n-3,\varsigma}$ in the classifier formula is the F-distribution value with parameters 3 and n-3, and level of significance ς.

Proof

Since x is the vector of r, g, b modes of a new panel, is independent of the n clean panels already used to compute $\bar{x}_m$. The

$$E(x - \bar{x}_m) = E\big((x - \mu) - (\bar{x}_m - \mu)\big) = E(x - \mu) - E(\bar{x}_m - \mu) = 0 \quad (5.14)$$

and the variance covariance matrix of the vector $x - \bar{x}_m$ is

$$E(x - \bar{x}_m)(x - \bar{x}_m)' = E[(x - \mu) - (\bar{x}_m - \mu)][(x - \mu) - (\bar{x}_m - \mu)]' = E(x - \mu)(x - \mu)' + E(\bar{x}_m - \mu)(\bar{x}_m - \mu)' - E(x - \mu)(\bar{x}_m - \mu)' - E(x - \mu)'(\bar{x}_m - \mu) =$$

$$\Sigma + \frac{1}{n}\Sigma - 0 - 0 = (1 + \frac{1}{n})\Sigma \quad (5.15)$$

Notice that $(x - \mu),\ and\ (\bar{x}_m - \mu)'$ are statistically independent therefore

$$E(x - \mu)(\bar{x}_m - \mu)' = 0 \tag{5.16}$$

For the same reason

$$E(x - \mu)'(\bar{x}_m - \mu) = 0 \tag{5.17}$$

an estimate of Σ is

$$\hat{\Sigma} = \frac{1}{n-1}\sum_{i=1}^{n}(x_{mi} - \bar{x}_m)(x_{mi} - \bar{x}_m)' \tag{5.18}$$

from this we infer that

$$t^2 = \frac{n}{n+1}(x - \bar{x}_m)'\hat{\Sigma}^{-1}(x - \bar{x}_m) \sim T^2_{3,n-1} \tag{5.19}$$

and therefore

$$\frac{n-3}{3(n-1)}t^2 \sim F_{3,n-3} \tag{5.20}$$

If ς is the level of significance chosen then the probability that

$$\frac{n-3}{3(n-1)}\frac{n}{n+1}(x - \bar{x}_m)'\hat{\Sigma}^{-1}(x - \bar{x}_m) < F_{3,n-3,\varsigma} \tag{5.21}$$

is (1-ς) , where ς is the level of significance and is selected to be less than or equal to 0.05.

From the above equation we obtain

$$(x - \bar{x}_m)'\hat{\Sigma}^{-1}(x - \bar{x}_m) < \frac{3(n^2-1)}{n(n-3)}F_{3,n-3,\varsigma} \tag{5.22}$$

or

$$x'\hat{\Sigma}^{-1}x < \bar{x}_m'\hat{\Sigma}^{-1}\bar{x}_m + \frac{3(n^2-1)}{n(n-3)}F_{3,n-3,\varsigma} \tag{5.23}$$

The right hand side of the above inequality is computed from the clean data after the sampling and formulation of the classifier. Every time we need to classify a new panel then we compute its vector x of r,g,b, modes and after computing the left hand side $x'\hat{\Sigma}^{-1}x$ of the above equation we compare it with the right hand side. If it satisfies the above inequality then we classify it as clean otherwise we trigger the cleaning mechanism to clean it.

IV. Conclusion

Solar panels in the arid climates produce a great deal of energy because the sun shines either throughout the year, or close to 350 days of the year. Sandstorms, small animals, and birds, depositing their droppings are responsible for the reduction of the energy produced by the panels. Here we introduced an algorithm that takes as input the panel picture and decides if the panel is clean or needs cleaning. This project is relatively new and the algorithm we described in this chapter is still being tested. So far our classification algorithm seems to be working. As the panels get dusty or subjected to bird droppings but they do not have enough deposits to pass our cut off point we see that the energy drop is within the normal fluctuation energy of clean panels. As they get dustier to the point that the inequality 5.21 is not being satisfied then we see that there is enough drop in energy that by investing some energy to clean the panel will

increase the energy output enough so that we can recover the energy invested by the panel within a few hours and have an energy gain until the next cleaning is required. The mean time between cleaning could be several months or less than a day depending on the weather conditions, the day of the month, the month of the year, and even the sun spot periodicity.

CHAPTER 6

POLLUTION DETECTION IN URBAN AREAS

USING THE EXISTING CAMERA NETWORKS (*) [69]

Abstract

Many cities throughout the world have surveillance cameras to monitor traffic events or for security reasons. In this chapter we develop a classification algorithm to detect the degree of pollution, and to report it on a large screen bellow the camera, so if the pollution becomes a health liability with people with respiratory problems then these people can be forewarned and stay indoors. The camera video becomes input to our classification algorithm that decides real time the degree of pollution and uses a quantization algorithm based on prior calibration and prints the number on the screen along with a qualitative score. The qualitative score is: "Today's Air Quality is Very Good", "Today's Air Quality is Good", "Today's Air Quality is Fair", "Today's Air Quality is Poor", "Today's Air Quality is Very Poor". Our classification algorithm consists of: our classification vector, the metric used, the training of the classifier, the testing of the classifier, and the classifier put into play for everyday use. At the present time we use existing commercial cameras transmitting MPEG-4 or H.264 frames

* - This chapter is accepted for publication in the International Journal of Multimedia Technology and is currently IN PRESS [69].

connected to our circuit board where the classification takes place. The algorithm resides on flash memory on our board. From there it is loaded into an ARM (Advanced RISC Machine) processor on the board, and operates on in coming data. Our intelligent system based on our algorithm classifies the air using a numeric system from 1 to 10. We have a transformation from the numeric quantization to the qualitative scale.

I. Introduction

Cameras are part of every landscape of every city in the world. As the population of the earth increases the crime rate increases. Economic instabilities, along with conflicting political views, are part of the reasons in the crime rise and violence. Also traffic violations are responsible for many fatalities and those are some of the reasons for the proliferation of the surveillance cameras. In this chapter, we take advantage of all these cameras and we use them to assess the air quality in a city. At every camera location we estimate the degree of air pollution based on our algorithm which quantizes the air pollution in the scale from 1 to 10 and then assigns a qualitative measure to air pollution. All this spatial data obtained are used to interpolate the pollution throughout the city. Thus we create isopleths showing where in the city the pollution has high levels, and where the air is cleaner. If the air pollution reaches unhealthy levels then we

issue warnings and inform the public living in the areas with unhealthy level of pollution, if they suffer from sinus infections, or asthma, or other respiratory problems to stay indoors. The majority of camera systems used for surveillance today produce an analog NTSC video signal with resolution 640x480 pixels at 29.97 frames per second. Or they produce a PAL-SECAM video at 25 frames per second and slightly higher resolution than the NTSC. The video producing chip is usually a Charge Couple Device (CCD), which produces 3 channels per pixel one is the Luma, and the other two are the Chroma channels. At night the resolution of the video is 320x480 and it is only Luma (grey scale). Unlike the Complementary Metal Oxide Semiconductor (CMOS) chips that require relatively small electric power to produce digital video, but they produce relatively low quality video, the CCD chips require more power than the CMOS to operate, and produce higher quality video than the CMOS chips. However the CCD produces analog video. This analog video becomes input to our circuit board consisting of an analog to digital converter chip, that converts each frame to a digital frame, with each pixel having an (R,G,B) vector. Thus a frame can be viewed as a 640x480 array of (R,G,B), vectors with each one of the three elements of the vector varying between 0 and 255.The (R, G, B) pixel composition, is also referred to as the additive color system and it is used in the computer monitors. Other systems such as the (C,M,Y), or (C,M,Y,K) which is also referred to as the

subtractive color system, are used in the printers. The (R,G,B) pixel values are highly auto correlated, and cross correlated. Thus here we consider each (R,G,B) vector as a random variable having a trivariate distribution function with a mean vector μ, which will be denoted as a column vector with three elements, and a variance covariance matrix Σ which is a 3x3 matrix with the diagonals being the variances of the red, green, and blue components, and the off diagonals being their covariances. This matrix Σ is a positive definite symmetric, with positive eigenvalues. We use all this information to create our classification vector which becomes input to our classifier. Based on the data collected we train the classifier to be able to classify a set of video frames to the proper degree of pollution. Once this is accomplished then the classifier becomes part of the software of our RISC chip which does all the intelligence related to classification. Once the video is digitized it is split into two identical signals, one is treated by the chip performing the classification, and the other by a RISC chip performing the compression real time. The classification chip outputs the findings on the screen bellow the camera to inform the public. It also transmits the information to our file server. This information is used to study air pollution quality trends for the city. The compressed video is also transmitted and stored in our file server. This chapter is structured as follows: First is the abstract, section I is the Introduction,

section II is the background information, section III is the classification algorithm, section IV is the conclusions.

II. Background information

A. The color of the Sky

The sky color variations from shades of blue to shades of gray indicate higher levels of concentration of particulates larger than the clean air. The clean air is composed approximately of 78.084% Nitrogen, 20.94% Oxygen, 0.934% Argon, and 0.0350% other gases [62], [63]. The color of the sky is explained by the Mie solution to Maxwell's equation, and Rayleigh's scatter approximation theory. According to Rayleigh's scatter theory the small molecule size of Nitrogen and Oxygen molecules scatter the components of the light with shorter wavelength range and thus higher frequency. These light components in the visible spectrum are the blue and violet colors. The scattering of the blue colors therefore by Nitrogen and Oxygen explains the blue color of the sky. Larger molecules scatter color components with longer wavelength. In the visible light the red color component of the white light has the larger wavelength, then the green color component has lower wavelength than the red, and the blue color component has lower wavelength than the green. When the concentration of larger molecules such as pollutants increase, the scattering of colors with longer

wavelength increases. The mix of the longer wavelength color components in the atmosphere with the blue, change the color of the sky, making it more gray or light gray. Thus when there is high concentration of water vapor molecules that cling together in that part of the sky make it look white or gray. This is the reason that clouds look white or gray. When pollution levels increase, since their molecule size is usually relatively large, that triggers the scattering of higher wavelength color components. Lighter grey colors signify the presence of water vapor, sulfur, aerosols, and soot in the atmosphere. In any given direction skylight is light scattered by all the molecules and particles in its path from the sun which is the source of daylight in the Earth. Since the atmosphere on the Earth is composed mostly of Nitrogen and Oxygen molecules, then these are the most frequent molecules that light will encounter in its trajectory. Electromagnetic waves with shorter wavelengths are affected by particles much smaller than their wavelength, scattering these waves in different directions. Waves with larger wavelength are not being affected by these small particles. As a result the blue and violet waves of the white light are scattered while traveling in the clean atmosphere turning the sky blue [64],[65],[66]. The lower intensity of the violet color compared to other colors in the light coming from the sun, combined with the lower sensitivity of this color by the human eyes is the main reason why the sky is blue and not violet even though the violet wavelength is

smaller than the blue and scatters even more. So there is very little scattering of the red color due to its relatively large wavelength. The red color travels straight thru the atmosphere. Similarly there is very little scattering for the green color because although its wavelength is smaller than that of red is still relatively large compared to the size of Nitrogen and Oxygen. So the green light travels straight thru a clean atmosphere.

B. Light and Air Pollution

The infrared and red wavelengths having the larger wavelengths compared to the other components of the white light, incident to various air pollution particulates move thru because of the relatively large wavelength of the red color . The blue and violet wavelengths suffer the most absorption by the air pollution particulates. As we pointed out in our previous section when the sky is cloudy due to water vapor, the red and green components of the light suffer a great deal of scattering and do not go straight thru the atmosphere. The infrared wavelengths are the ones that go straight thru most of the pollution particulates in the atmosphere, following by the red, and green. The energy produced by the red and green wavelengths is substantially reduced during cloudy days. Also when dust particulates, or other pollutants are in the atmosphere they cause the scattering of the larger wavelength components of light. Dust storms are very

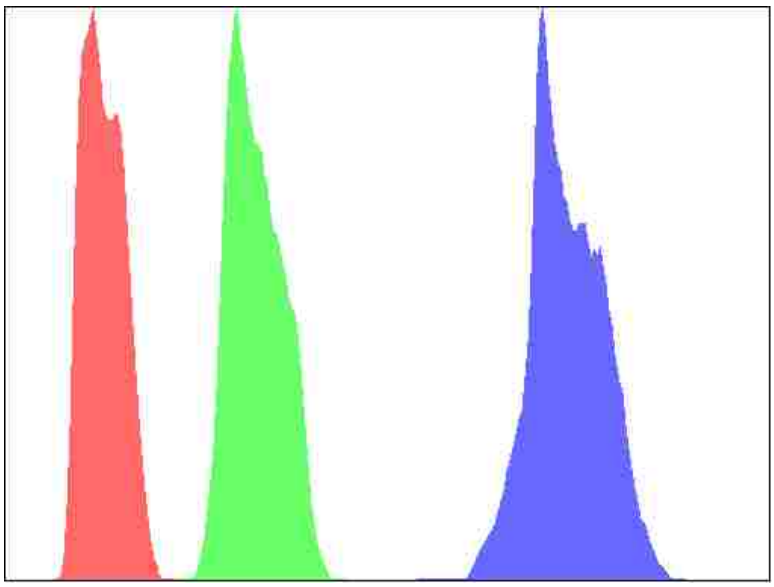
common in arid climates, especially during the summer months when the ground is very dry and the top soil is sandy. The larger dust particulates in the air scatter the larger light wavelengths.

III. Classification Algorithm

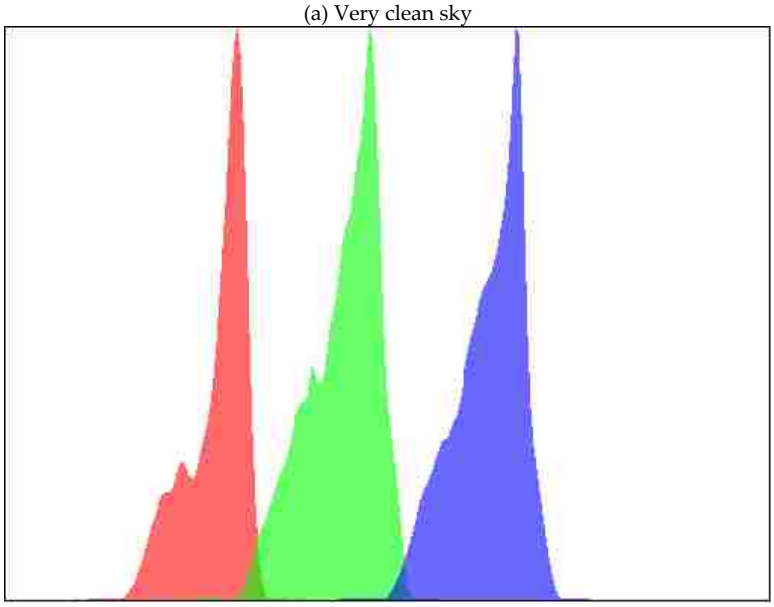
As we examine an image of a clean air which we characterize as very good quality, Figure 6.1, and compute the probability density functions of the red channel, the green channel, and the blue channel of the pixels of the image, Figure 6.3, we observe that each one of the marginal probability density functions for each one of the channels is skewed with a maximum value (mode) which for the red and green is to the left of the 127.5 value. Here we assume that the scale of the red channel of the pixels is between 0-255, similarly the scales of the green and blue channels of the pixels are also between 0-255. Thus in the graphs of the probability density function of the color channels the scale on the x-axis is between 0-255. Figure 6.3(a) gives the univariate probability distribution functions of the red, green, and blue, channels, for a very clean air. The mode for the red channel is 30, for the green is 78, and for the blue is 180. The x-axis for each one of the channels is from 0-255. Figure 6.3(b) gives the univariate probability distribution functions of the red, green, and blue channels, for a good air quality, Figure 6.2.

Figure 6.1 Image of a very clean sky

Figure 6.2 Image of a clean sky



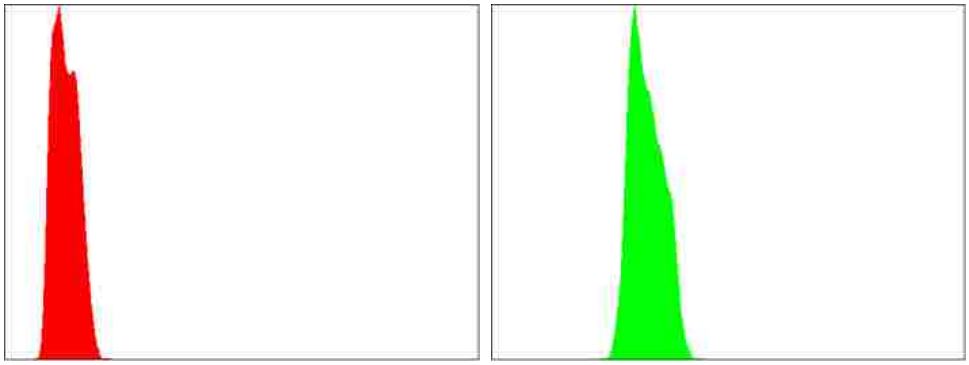
(a) Very clean sky



(b) Clean sky

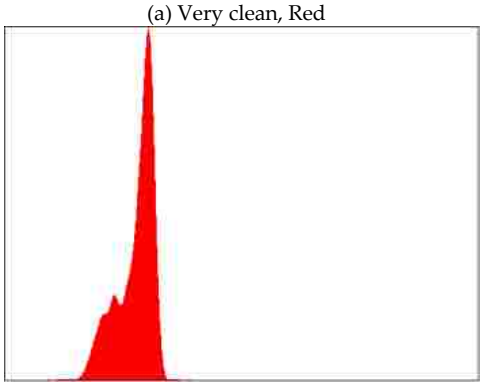
Figure 6.3 Probability density function for the R, G, B channels, Observe the shift of the red, green, and blue modes

The mode for the red channel is 78, for the green is 122, and for the blue is 171. Notice that as the air quality decreased the mode of the red component increased, also of the green, but the mode of the blue decrease.

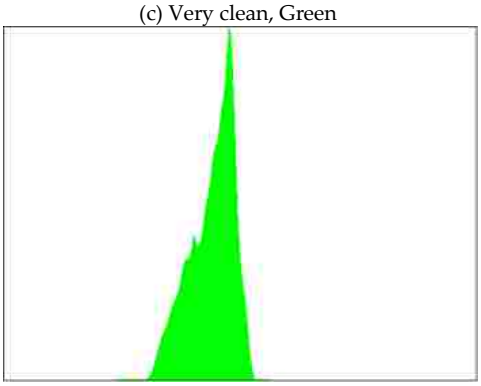


(a) Very clean, Red

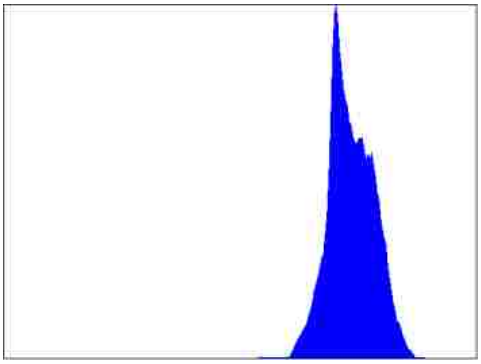
(c) Very clean, Green



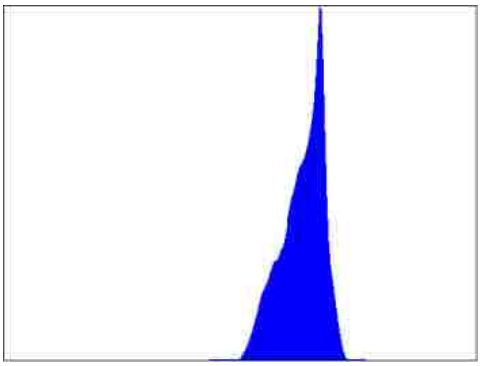
(b) Clean, Red



(d) Clean, Green



(e) Very clean, Blue



(f) Clean, Blue

Figure 6.4 Individual probability density functions for R, G, B channels of both the very clean and the clean sky images

Based on a sample of 50 pictures of very clean air quality, and fifty of clean ones, the gamma distribution provides enough flexibility to model these probability functions. The gamma density function is of the form [67], [68]:

$$f(x) = \frac{x^{\alpha-1}e^{-\frac{x}{\beta}}}{\Gamma(\alpha)\beta^{\alpha}},\ x > 0,\ \alpha, \beta > 0 \tag{6.1}$$

Theorem 1, gives an estimate of the global mode of the gamma density function in terms of its parameters α, β, and also estimates the parameters from the data.

Theorem 1. The global mode of the gamma density function (the point x that maximizes the density function f(x)) is $x_m = (\alpha - 1)\beta$, if $\hat{x}$ is the global mode based on the empirical density function computed from the data, and $\bar{x}$ is the average then an estimate of the parameters α, β, from the data is: $\hat{\alpha} = \frac{\bar{x}}{\bar{x}-\hat{x}}$ and $\hat{\beta} = \bar{x} - \hat{x}$

Proof

The global mode x_m of the gamma probability function is the point that maximizes the density function. Namely the point for which the first derivative of the function f(x) is zero and the second derivative is negative.

$$f'(x) = e^{-\frac{x}{\beta}}\ \frac{(\alpha-1)x^{\alpha-2} - \frac{x^{\alpha-1}}{\beta}}{\Gamma(\alpha)\beta^{\alpha}} = 0 \tag{6.2}$$

The solution to the above equation is

$$x_m = (\alpha - 1)\beta \tag{6.3}$$

$$f''(x) = -\frac{1}{\beta}e^{-\frac{x}{\beta}}\frac{(\alpha-1)x^{\alpha-2} - \frac{x^{\alpha-1}}{\beta}}{\Gamma(\alpha)\beta^{\alpha}} + e^{-\frac{x}{\beta}}\frac{(\alpha-1)(\alpha-2)x^{\alpha-3} - \frac{(\alpha-1)x^{\alpha-2}}{\beta}}{\Gamma(\alpha)\beta^{\alpha}}$$

(6.4)

For the type of gamma probability function that fits this data $\alpha > 2$ always. Thus

$$f''(x_m) = -e^{-(\alpha-1)}\frac{\left((\alpha-1)\beta\right)^{\alpha-3}}{\Gamma(\alpha)\beta^{\alpha}} < 0 \qquad (6.5)$$

The point $x_m = (\alpha-1)\beta$ is the global maximum for f(x), thus it is the mode of the density function. It is very easy to prove that the mean or expected value of a gamma is $\mu=\alpha\beta$. Given a panel image we can use the pixels to extract the values of the red, green, and blue channels. For each one of these channels we can estimate the mean, and the maximum mode. The mean is the average of all the values for that channel, and the maximum mode of course is the point between 0 and 255 for which most of the values of that channel for the picture that the channel belongs to, are equal to.

Let the sample mean for the arbitrary channel be denoted by $\bar{x}$ and let the estimate of the maximum mode be denoted by $\hat{x}$. Then the parameters α, β, of the gamma can be estimated by setting:

$$\bar{x} = \alpha\beta \qquad (6.6)$$

and

$$\hat{x} = (\alpha - 1)\beta \tag{6.7}$$

From the above two equations we obtain

$$\bar{x} - \beta = \hat{x} \tag{6.8}$$

which implies that

$$\hat{\beta} = \bar{x} - \hat{x} \tag{6.9}$$

From equation 6.6 therefore we obtain that

$$\hat{\alpha} = \frac{\bar{x}}{\bar{x} - \hat{x}} \tag{6.10}$$

QED.

Formulas 6.9 and 6.10 give us estimates of the parameters α and β which they define the gamma density function for the channel, based on the channel data. The picture of each one of the obtained 50 very clean air quality pictures has relatively small variation. The fifty clean air pictures also have relatively small variation between one another. For each one of the pictures we computed the probability density function of the red, green, and blue, channels. As we graphed each one of the probability density functions, the x-axis denotes all the possible values a color-channel can have. Thus the values for each one of the three channels vary between 0-255. The value in the x-axis for which the experimental density function obtains its global maximum is the mode. For the very clean air pictures we obtained 50 mode data for the red channel, 50 mode data for the green, and 50 mode data for the blue. In each one of the pictures the

mode for the blue is over 20 units higher than the mode for the green and the mode for the green is higher than the mode for red. As the air quality shifts from very good air quality to good, then the modes for red and green are shifting to higher values, while the mode for the blue shifts to lower values. The classification theory that enables us to make the decision between very clean air and clean air is based on the following statistics. Consider the 50 very clean air quality. Let $(r_{m1}, r_{m2}, \dots, r_{m50})$, $(g_{m1}, g_{m2}, \dots, g_{m50})$ and $(b_{m1}, b_{m2}, \dots, b_{m50})$, be the 50 computed mode values for each one of the very clean channels. Let $\bar{r}_m, \bar{g}_m, \bar{b}_m$ denote the average of the red-modes, green-modes, and blue-modes respectively. Then according to the central limit theorem the probability distribution function, of the vector $\bar{r}_m, \bar{g}_m, \bar{b}_m$ approaches to a trivariate normal. The random vectors $(r_{m1}, g_{m1}, b_{m1}), (r_{m2}, g_{m2}, b_{m2}), \dots, (r_{m50}, g_{m50}, b_{m50})$, are independent since each picture is autonomous and independent of the other pictures, and identically distributed, with a mean vector $\underline{\mu}' = (\mu_{rm}, \mu_{gm}, \mu_{bm})$, where the $\underline{\mu}'$ signifies the transpose vector of the mean, and variance- covariance matrix is:

$$\Sigma = \begin{bmatrix} \sigma^2_{rm} & \sigma_{rmgm} & \sigma_{rmbm} \\ \sigma_{rmgm} & \sigma^2_{gm} & \sigma_{gmbm} \\ \sigma_{rmbm} & \sigma_{gmbm} & \sigma^2_{bm} \end{bmatrix} \tag{6.11}$$

The expected value $E(\bar{r}_m, \bar{g}_m, \bar{b}_m)$ $\underline{\mu}' = (\mu_{rm}, \mu_{gm}, \mu_{bm})$, and the variance covariance matrix of the vector $\bar{r}_m, \bar{g}_m, \bar{b}_m$ is $\frac{\Sigma}{50}$. In general if n-pictures are used to compute the vector $\bar{r}_m, \bar{g}_m, \bar{b}_m$, then the variance-covariance vector

of $\bar{r}_m, \bar{g}_m, \bar{b}_m$, would be $\frac{\Sigma}{n}$. The vector $\underline{\mu}' = (\mu_{rm}, \mu_{gm}, \mu_{bm})$ is estimated from the data. Thus if a sample of n clean panels is used and we denote the trivare vector of the mode of the i[th] panel as $x_{mi} = (r_{mi}, g_{mi}, b_{mi})$ then

$$\bar{x}_m = \frac{x_{m1} + x_{m2} \ldots + x_{mn}}{n} \tag{6.12}$$

And the variance covariance matrix Σ can be estimated from the data as follows:

$$\hat{\Sigma} = \frac{1}{n-1} \sum_{i=1}^{n} (x_{mi} - \bar{x}_m)(x_{mi} - \bar{x}_m)' \tag{6.13}$$

Theorem 2 creates the criterion for classifying a panel as being clean or needing cleaning.

Theorem 2. Let x denote the vector of r,g,b, modes of a new panel and $t^2 = \frac{n}{n+1}(x - \bar{x}_m)'\hat{\Sigma}^{-1}(x - \bar{x}_m)$ then $\frac{n-3}{3(n-1)} t^2 \sim F_{3,n-3}$ is an F distribution with parameters 3, n-3. If $x'\hat{\Sigma}^{-1}x < \bar{x}_m{}'\hat{\Sigma}^{-1}\bar{x}_m + \frac{3(n^2-1)}{n(n-3)} F_{3,n-3,\varsigma}$, then the panel is classified as clean, otherwise as needed to be cleaned. $F_{3,n-3,\varsigma}$ in the classifier formula is the F-distribution value with parameters 3 and n-3, and level of significance ς.

Proof

Since x is the vector of r, g, b modes of a new panel, is independent of the n clean panels already used to compute $\bar{x}_m$. The

$$E(x - \bar{x}_m) = E\big((x - \mu) - (\bar{x}_m - \mu)\big) = E(x - \mu) - E(\bar{x}_m - \mu) = 0 \tag{6.14}$$

and the variance covariance matrix of the vector $x - \bar{x}_m$ is

$$E(x - \bar{x}_m)(x - \bar{x}_m)' = E[(x - \mu) - (\bar{x}_m - \mu)][(x - \mu) - (\bar{x}_m - \mu)]' = E(x - \mu)(x - \mu)' + E(\bar{x}_m - \mu)(\bar{x}_m - \mu)' - E(x - \mu)(\bar{x}_m - \mu)' - E(x - \mu)'(\bar{x}_m - \mu) = \Sigma + \frac{1}{n}\Sigma - 0 - 0 = (1 + \frac{1}{n})\Sigma \quad (6.15)$$

Notice that $(x - \mu),\ and\ (\bar{x}_m - \mu)'$ are statistically independent therefore

$$E(x - \mu)(\bar{x}_m - \mu)' = 0 \quad (6.16)$$

For the same reason

$$E(x - \mu)'(\bar{x}_m - \mu) = 0 \quad (6.17)$$

an estimate of Σ is

$$\hat{\Sigma} = \frac{1}{n-1}\sum_{i=1}^{n}(x_{mi} - \bar{x}_m)(x_{mi} - \bar{x}_m)' \quad (6.18)$$

From this we infer that

$$t^2 = \frac{n}{n+1}(x - \bar{x}_m)'\hat{\Sigma}^{-1}(x - \bar{x}_m) \sim T^2_{3,n-1} \quad (6.19)$$

and therefore

$$\frac{n-3}{3(n-1)}t^2 \sim F_{3,n-3} \quad (6.20)$$

If ς is the level of significance chosen then the probability that

$$\frac{n-3}{3(n-1)}\frac{n}{n+1}(x - \bar{x}_m)'\hat{\Sigma}^{-1}(x - \bar{x}_m) < F_{3,n-3,\varsigma} \quad (6.21)$$

is (1-ς) , where ς is the level of significance and is selected to be less than or equal to 0.05.

From the above equation we obtain

$$(x - \bar{x}_m)'\hat{\Sigma}^{-1}(x - \bar{x}_m) < \frac{3(n^2-1)}{n(n-3)}F_{3,n-3,\varsigma} \quad (6.22)$$

or

$$x'\hat{\Sigma}^{-1}x < \bar{x}_m'\hat{\Sigma}^{-1}\bar{x}_m + \frac{3(n^2-1)}{n(n-3)}F_{3,n-3,\varsigma} \tag{6.23}$$

The right hand side of the above inequality is computed from the very clean air data after the sampling, formulation, and testing of the classifier. Every time we need to classify a new picture that we know it is either the picture of very clean air, or clean air, then we compute its vector x of r,g,b, modes and after computing the left hand side $x'\hat{\Sigma}^{-1}x$ of the above equation we compare it with the right hand side. If it satisfies the above inequality then we classify it as very clean air quality otherwise we classify it as clean air quality. The initial classification of the samples was based on traditional analysis of air samples in the laboratory, followed by our analysis to help us calibrate our method with respect to the traditional well accepted laboratory methods of testing air quality. Our quantization was based on a scale of 1 to 10, with 9 and 10 classified as very good air quality, 7,8 as good, 5,6 as fair, 3,4 as poor, and 1,2 as very poor. The boundaries for the classification between clean, fair, poor, and very poor were made in a similar way as described above between very clean and clean air. Thus a new picture of the sky after being processed the way described above it will fall in one of the five clusters which will enable us to classify in one of the five air quality categories.

IV. Conclusion

In our day and time surveillance cameras are very common in every city. Here we described a method that uses these cameras to characterize the air quality. Based on the result of air quality in each one of the camera locations we can interpolate and estimate the air pollution throughout the city. Furthermore we can report the air quality real time at the camera location by using a display panel. This could be a useful warning in case of unhealthy air quality conditions. The air quality can be transmitted to a file server where can be stored, and used to understand long term air quality patterns.

CHAPTER 7

CONCLUSION AND FUTURE WORK

Through this research, computer vision based on cameras is investigated. The use of cameras is a cost and energy effective alternative especially when the use of other sensors is limited. Images taken by cameras provide more information than sensors do but need some effort to retrieve them. Lighting condition and image quality has great effect on the ability of retrieving useful information from images. Improper lighting condition and image quality can cause problems such as the existence of shadows, and the relatively small dynamic range of the gray-scale or red, green, and blue values of the image. In the edge detection process, the presence of shadow results in false classification of shadows as objects and detection of broken edges or extra false edges. Small dynamic range also results in broken or week edges. For fixed lighting condition, Canny's edge detection parameters can be optimized and applied for all images taken in this environment. Applying the proposed dynamic range transformation and the image histogram equalization along with the proposed image shifting technique results in better distribution of the image histogram and enough elimination of shadows, and hence, edge detection that leads to proper segmentation. Using the detected edge maps, the Hough transform leads to the

extraction of lines, and other geometries, with some deficiencies. These deficiencies include missing parts of lines, and fragmentation of some lines. Algorithms to solve these problems are introduced.

A stereoscopic computer vision system, consisting of four identical cameras in a cross configuration and a non-invasive laser pointer, is described. The laser pointer has two modes; in one mode, it transmits a dot, and in the other mode it transmits a grid of squares. The system of the four cameras is supported by a mechanism that allows it to pan and tilt. Each camera is supported by its own pan and tilt mechanism. The mathematics and analytics of this computer vision system are presented. This stereoscopic vision system allows for faster and more accurate estimation of the 3-D positions of points in the working space, in comparison to the estimation done in the epipolar geometry. The mathematical relations between the 2-D information in each image plane and the actual 3-D position is theoretically formulated. This formulation leads to compute the minimum geometrical description of the target object that allows for reliable robotic handling. Utilizing the dot mode of the non-invasive laser, which marks a non-disputable corresponding point, objects of regular geometry can be identified using the images produced by the four cameras. These four images have to be processed using the proposed transformations first to get the proper edge maps. The reference points, e.g. corner points, have to be extracted from the

edge maps using Hough transform and then the proposed correction algorithms. In case of irregular objects, the laser square grid projects on the object as irregularly shaped quadrilateral. The 3-D coordinates of the corners of this quadrilateral can be computed using the developed mathematical relations, and hence, geometrical aspects of the objects can be determined.

As a computer vision application, the detection of cleanness condition of remote solar panels using image analysis is investigated. Since the dust and other deposits on the solar panels affect the produced energy, a classifier that detect the presence of dust and other deposits is developed. The classifier is based on analysis of the trivariate distribution function of the red, green, and blue channels of the solar panel images. As the panels get dusty but they do not have enough deposits to pass a cut off point, the energy drop is considered within the normal fluctuation energy of clean panels. As they get dustier and the cut off point is reached, the drop in energy is enough to consider investing some energy to clean the panel and restore the produced energy to an optimal level.

Another computer vision application is based on using existing camera networks to characterize the air quality. A classification algorithm is developed to detect the degree of air quality (or degree of pollution). The video streamed by existing cameras becomes input to the classification algorithm that decides real

time the degree of pollution and uses a quantization algorithm based on prior calibration and prints the number on the screen along with a qualitative score.

Future Work

The theory of stereo vision presented in this dissertation is planned to be applied to an actual robot in a real working environment to perform handling and assembly tasks. The theory will also be applied to measure the dimensions and volume of regularly and irregularly shaped luggage in order to place them in containers to be loaded on airplanes as cargo and optimize the weight distribution with respect to the center of gravity of the airplane. Another future application is the telemedicine where the vision system can be used to allow a physician to follow up with patient in rural or remote places. With the help of the nurse and the robotic vision system calculates the progress once or twice a day, by computing the volume of the area affected by the surgery and also the amount of reduction or increase since the last time. This information can be stored to the hospital file server and hence can be transmitted in real time to the physician.

The smart cleaning robot and vision system for solar panel dust detection is being built and will be tested in real time.

BIBLIOGRAPHY

1 - Raval, Harshil N., Shree Prakash Tiwari, Ramesh R. Navan, and V. Ramgopal Rao. "Determining ionizing radiation using sensors based on organic semiconducting material." *Applied Physics Letters* 94, no. 12 (Mar 2009): 123304-123304-3.

2 - Kotthäuser, Tobias, and Georg F. Mauer. "Vision-Based Autonomous Robot Control For Pick And Place Operations." *2009 IEEE/ASME International Conference on Advanced Intelligent Mechatronics.* Singapore, 2009. 1851 - 1855.

3 - Peynot, Thierry, James Underwood, and Abdallah Kassir. "Sensor Data Consistency Monitoring for the Prevention of Perceptual Failures in Outdoor Robotics." *Proceedings of the 7th IARP Workshop on Technical Challenges for Dependable Robots in Human Environments.* Toulous, France, 2010. 145-152.

4 - IFM effector, INC. *IFM effector, INC.* http://www.ifm.com/ifmgb/web/pinfo1_50_25_40.htm (accessed Oct 2011).

5 - Davison, Andrew John. "Mobile Robot Navigation Using Active Vision." *PhD Dissertation, Engineering Science.* University of Oxford, UK, 1998.

6 - *Wikipedia.* http://en.wikipedia.org/wiki/Robot (accessed Oct 2011).

7 - Abrahama, Steffen, and Wolfgang Forstner. "Fish-eye-stereo calibration and epipolar rectification." *Journal of Photogrammetry & Remote Sensing* 59 (2005): 278– 288.

8 - Jeong, Yongwoo. "Bio-Inspired Hardware/Software Motion Sensor System." *PhD Dissertation, Electrical Engineering.* The State University of New York at Buffalo, May 2011.

9 - Kunkel, Anthony . "Obstacle Avoidance and Target Tracking Utilizing Fly Eye Sensors." *MS thesis, Electrical Engineering.* University of Wyoming, Dec 2010.

10 - Sriram, Vinay, David Cox, Kuen Hung Tsoi, and Wayne Luk. "Towards an Embedded Biologically-Inspired Machine Vision Processor." *International Conference on Field-Programmable Technology (FPT).* 2010. 273-278.

11 - DiCarlo, James J., and David D. Cox. "Untangling invariant object recognition." *Trends in Cognitive Sciences* 11, no. 8 (2007): 333-341.

12 - Pinto, Nicolas, David D. Cox, and James J. DiCarlo. "Why is Real-World Visual Object Recognition Hard?" *PLoS Computational Biology* 4, no. 1 E27 (2008): 0151-0156.

13 - Szeliski, Richard, and Bing Kang. "Direct Methods for Visual Scene Reconstruction." *IEEE Workshop on Representations of Visual Scenes.* Cambridge, MA: IEEE, 1995. 26-33.

14 - Gurram, Prudhvi K. "Automated 3D Object Modeling from Aerial Video Imagery." *PhD Dissertation, Electrical Engineering.* Rochester Institute of Technology, Sep 2009.

15 - Kim, Taeone, Yongduek Seo, and Ki-Sang Hong. "Physics-based 3D Position Analysis of a Soccer Ball from Monocular Image Sequences." *Proceedings of the 6th International Conference on Computer Vision.* 1998. 721-726.

16 - Criminisi, A., I. Reid, and A. Zisserman. "Single View Metrology." *Proceedings of the Seventh IEEE International Conference on Computer Vision.* Kerkyra, Greece, 1999. 434 - 441.

17 - Kotthäuser, Tobias. "Reliable Reconstruction and Manipulation of Cylindrical Objects within Interior Environment – Conception and Implementation." *MS Thesis, Mechanical Engineering.* University of Nevada Las Vegas, Dec 2007.

18 - Heikkilä, Janne, and Olli Silvén. "A four-step camera calibration procedure with implicit image correction." *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'97).* Washington, DC, 1997. 1106-1112.

19 - Tsai, Roger Y. "An Efficient and Accurate Camera Calibration Technique for 3D Machine Vision." *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition.* Miami Beach, FL, 1986. 364-374.

20 - Tsai, Roger Y. "A versatile Camera Calibration Technique for High-Accuracy 3D Machine Vision Metrology Using Off-the-Shelf TV Cameras and Lenses." *IEEE Journal of Robotics and Automation* 3, no. 4 (1987): 323-344.

21 - Clarke, T. A., and J. F. Fryer. "The development of camera calibration methods and models." *Photogrammetric Record* 16, no. 91 (1998): 51-66.

22 - Heikkilä, J. "Geometric Camera Calibration Using Circular Control Points." *IEEE Transactions on Pattern Analysis and Machine Intelligence* 22, no. 10 (2000): 1066-1077.

23 - Brown, D. C. "Decentering distortion of lenses." *Photogrammetric Engineering* 32, no. 3 (1966): 444-462.

24 - Heikkila, J., and O. Silven. "Calibration Procedure for Short Focal Length off-the Shelf CCD Cameras." *Proceedings of the 13th International Conference on Pattern Recognition.* 1996. 166-170.

25 - Al-Najdawi, Nijad, Helmut E. Bez, Jyoti Singhai, and Eran. A. Edirisinghe. "A survey of cast shadow detection algorithms." *Pattern Recognition Letters* (Elsevier Science Inc.) 33, no. 6 (April 2012): 752–764.

26 - Sanin, Andres, Conrad Sanderson, and Brian C. Lovell. "Shadow detection: A survey and comparative evaluation of recent methods." *Pattern Recognition* 45, no. 4 (2012): 1684–1695.

27 - Horprasert, Thanarat, David Harwood, and Larry S. Davis. "A Statistical Approach for Real-time Robust Background Subtraction and Shadow Detection." *Proceedings IEEE ICCV.* 1999. 1-19.

28 - Finlayson, Graham D., and Steven D. Hordley. "Color constancy at a pixel." *Optical Society of America A* 18, no. 2 (Feb 2001): 253-264. Also, UK Patent application no. 0000682.5. Under review, British Patent Office.

29 - Prati, Andrea, Ivana Mikic, Mohan M. Trivedi, and Rita Cucchiara. "Detecting Moving Shadows: Algorithms and Evaluation." *IEEE Transactions on Pattern Analysis and Machine Intelligence* 25, no. 7 (2003): 918–923.

30 - Miyazaki, Daisuke, Yasuyuki Matsushita, and Katsushi Ikeuchi. "Interactive Shadow Removal from a Single Image Using Hierarchical Graph Cut." *Proceedings Of the 9th Asian Conf on Computer Vision, ACCV'09.* Xi'an: Springer-Verlag, 2010. 234-245.

31 - Lalonde, Jean-Francois, Alexei A. Efros, and Srinivasa G. Narasimhan. "Detecting ground shadows in outdoor consumer photographs." Edited by Kostas Daniilidis, Petros Maragos and Nikos Paragios. *Proceedings of the 11th European conference on Computer vision: Part II (ECCV'10).* Heraklion, Crete, Greece, 2010. 322-335.

32 - Ullah, Habib, Mohib Ullah, Muhammad Uzair, and Fasih Rehman. "Comparative Study: The Evaluation of Shadow Detection Methods, International Journal of Video & Image Processing and Network Security IJVIPNS-IJENS." *International Journal of Video & Image Processing and Network Security IJVIPNS-IJENS Vol:10 No:02* 10, no. 2 (2010).

33 - Kumar, Sanjeev, and Anureet Kaur. "Shadow Detection and Removal in Color Images using Matlab." *International Journal of Engineering Science and Technology* 2, no. 9 (2010): 4482-4486.

34 - Canny, John. "A Computational Approach to Edge Detection." *IEEE Transactions on Pattern Analysis and Machine Intelligence* PAMI-8, no. 6 (1986): 679,698.

35 - Deriche, Rachid. "Using Canny's Criteria to Derive a Recursively Implemented Optimal Edge Detector." *International Journal of Computer Vision* (Kluwer Academic Publishers, Boston) 1, no. 2 (1987): 167-187.

36 - Russ, John C. *The Image Processing Handbook.* Fifth Edition. London: CRC Press, 2006.

37 - Ravikumar, S., and A. Shanmugam. "Performance on Image Segmentation Resulting In Canny and MoG." *IOSR Journal of Computer Engineering (IOSRJCE)* 6, no. 2 (2012): 01-08.

38 - Duda, Richard O., and Peter E. Hart. "Use of the Hough Transformation to Detect Lines and Curves in Pictures." *Communications of the ACM* 15, no. 1 (1972): 11-15.

39 - Hough, Paul V. C. Method and means for recognizing complex patterns. Patent US 3069654. Dec 18, 1962.

40 - Rosenfeld, Azriel. *Picture Processing by Computer.* New York: Academic Press, 1969.

41 - Griffith, A. K. "Computer recognition of prismatic solids." *Ph.D. Dissertation, Department of Mathematics.* MIT, 1970.

42 - Bouguet, Jean-Yves. *Camera Calibration Toolbox for Matlab.* July 9, 2010. http://www.vision.caltech.edu/bouguetj/calib_doc/htmls/ref.html (accessed Aug 2013).
43 - Ogale, Abhijit S., and Yiannis Aloimonos. "Shape and the Stereo Correspondence Problem." *International Journal of Computer Vision* 65, no. 3 (2005): 147-162.
44 - Guerra-Filho, Gutemberg. "An optimal time-space algorithm for dense stereo matching." *Journal of Real-Time Image Processing* 7, no. 2 (2012): 69-86.
45 - Fayed, A. "Image Segmentation, Object Detection, Robot Vision, and Distance Estimation." *International Journal of Advanced Robotic Systems*, SUBMITTED July 19, 2013: IJARS-D-13-00070.
46 - Faugeras, O. D., and G. Toscani. "The Calibration Problem for Stereo." *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'86).* Miami Beach, FL, 1986. 15-20.
47 - Faugeras, O. D., and G. Toscani. "Camera Calibration for 3D Computer Vision." *Proceedings of the International Workshop on Industrial Application of Machine Vision and Machine Intelligence.* 1987. 240-247.
48 - Weng, J., P. Cohen, and M. Herniou. "Camera Calibration with Distortion Models and Accuracy Evaluation." *IEEE Transaction on Pattern Analysis and Machine Intelligence* 14, no. 10 (1992): 965-980.
49 - Trucco, E., and A. Verri. *Introductory Techniques for 3-D Computer Vision.* Prentice Hall, 1998.
50 - Salvi, J., X. Armangue, and J. Batlle. "A Comprehensive Review of Camera Calibrating Methods with Accuracy Evaluation." *Pattern Recognition* 36, no. 7 (2002): 1617-1635.
51 - Huang, Xinyu, Jizhou Gao, and Ruigang Yang. "Calibrating Pan-Tilt Cameras with Telephoto Lenses." *Proceedings of the 8th Asian conference on Computer vision - Volume Part I (ACCV'07).* Tokyo: Springer-Verlag, Berlin, Heidelberg, 2007. 127-137.
52 - Quan, Long. "Self-calibration of an affine camera from multiple views." *International Journal of Computer Vision* 19, no. 1 (1996): 93-105.

53 - Teruel, M. B., O. Kubushyna, R. El-Khater, E. A. Yfantis, and R. Boyle. "On Estimation of 3D Hand Position Using Stereo Vision." *Proceedings of the ISCA 19th International Conference on Computers and their Applications.* 2004. 47-50.

54 - El-Khater, Ramzi W. "Camera calibration for the NASA Virtual GloveBox project." *M.S. Thesis, Computer Science Department.* University of Nevada Las Vegas, 2004. 1-54.

55 - Teruel, M. B., K. L. Sauer, E. A. Yfantis, and R. W. El-Khater. "A Novel Camera Calibration Algorithm as Part of an HCI System: Experimental Procedure and Results." *Proceedings of the 9th World Multi-Conference on Systemics, Cybernetics and Informatics.* 2005. 212-217.

56 - Teruel, M. B., K. L. Sauer, E. A. Yfantis, and R. W. El-Khater. "Hand Pose Detection Cartesian Coordinate Reconstruction." *Proceedings of the International Conference on Human-Computer Interaction.* 2005. 33-39.

57 - Teruel, M. B. "Color Detection and 3D Reconstruction For Hand Pose Estimation As part Of An HCI System." *M.S. Thesis, Computer Science Department.* University of Nevaada Las Vegas, 2006. 1-83.

58 - Fayed, A. "A Camera System for Detecting Dust and Other Deposits on Solar Panels." *7th ACM/IEEE International Conference on Distributed Smart Cameras (ICDSC13).* Palm Springs, CA, Oct 29 – Nov 1, 2013. SUBMITTED.

59 - Ryan, C. P., F. Vignola, and D. K. McDaniels. "Solar cell arrays: Degradation due to dirt." *Proceedings of the Annual Meeting, American Solar Energy Society, Inc.* 1989. 234 -237.

60 - Mazumder, M. K., A. S. Biris, C. U. Yurteri, R. A. Sims, C. E. Johnson, R. Sharma, K. Pruessner, C. I. Calle, S. Trigwell, C. R. Buhler, and J. S. Clements. "Solar Panel Obscuration by Dust and its Mitigation in the Martian Atmosphere." Edited by K. L. Mittal. *Proceedings of the 9th International Symposium on Particles on Surfaces: Detection, Adhesion and Removal.* Philadelphia, PA: VSP Publications, 2006. 1-29.

61 - Mazumder, M. K., R. Sharma, A. S. Biris, J. Zhang, C. Calle, and M. Zahn. "Self-Cleaning Transparent Dust Shields for Protecting Solar Panels and Other Devices." *Particulate Science and Technology* 25, no. 1 (2007): 5-20.

62 - Bohren, C. E., and A. B. Fraser. "Colors of the Sky." *The Physics Teacher* 5 (1985): 269-272.

63 - van de Hulst, H. C. *Light Scattering by Small Particles.* New York: John Wiley and Sons Inc., 1957.

64 - Haralabidis, P. E., and C. Pilinis. "Skylight Color Shifts due to Variations of Urban-Industrial Aerosol Properties: Observer Color Difference Sensitivity Compared to a Digital Camera." *Aerosol Science and Technology* 8, no. 42 (2008): 658-673.

65 - Ramos, E. Zamora. "Using Image Processing Techniques to Estimate the Air Quality." *Journal of Mcnair Scholars Institute, 6th Edition*, 2012: 189-194.

66 - Tsay, Si-Chee, Graeme L. Stephens, and Thomas J. Greenwald. "An Investigation of Aerosol Microstructure on Visual Air Quality." *Atmospheric Environment A* 25, no. 5-6 (1991): 1039-1053.

67 - Hogg, R. V., and A. T. Craig. *Introduction to mathematical Statistics.* Fourth Edition. Macmillan Publishing Co., 1978.

68 - Theodoridis, S., and K. Koutroumbas. *Pattern recognition.* Fourth Edition. Academic Press, 2008.

69 - Fayed, A. "Pollution Detection in Urban Areas Using the Existing Camera Networks." *International Journal of Multimedia Technology (IJMT)*, IN PRESS.

VITA

Graduate College
University of Nevada, Las Vegas

Ahmad Fayed

Degrees:

Bachelor of Science, Mechanical Engineering, 1997
Al-Azhar University, Cairo, Egypt

Master of Science, Mechanical Engineering, 2007
King Saud University, Riyadh, KSA

Special Honors and Awards:

Outstanding Teaching Assistant, Mechanical Engineering Dept, UNLV 2012/2013
Outstanding Teaching Assistant, Mechanical Engineering Dept, UNLV 2011/2012
Outstanding Teaching Assistant, Mechanical Engineering Dept, UNLV 2010/2011
Summer Session Scholarship, Graduate College, UNLV, Summer 2011
Graduate Access Grant, Graduate College, UNLV 2008 – 2013

Publications:

Fayed, A. "Image Segmentation, Object Detection, Robot Vision, and Distance Estimation." *International Journal of Advanced Robotic Systems*, SUBMITTED July 19, 2013: IJARS-D-13-00070.

Fayed, A. "A Camera System for Detecting Dust and Other Deposits on Solar Panels." *7th ACM/IEEE International Conference on Distributed Smart Cameras (ICDSC13).* Palm Springs, CA, Oct 29 – Nov 1, 2013. SUBMITTED.

Fayed, A. "Pollution Detection in Urban Areas Using the Existing Camera Networks." *International Journal of Multimedia Technology (IJMT)*, IN PRESS.

Fayed, A. "Authentication and Secure Robot Communication." *International Journal of Advanced Robotic Systems*, SUBMITTED June 24, 2013: IJARS-D-13-00063.

ElMadany, M. M., B. A. Al Bassam, and A. A. Fayed. "Preview Control of Slow-active Suspension Systems." *Journal of Vibration and Control* 17, no. 2 (2011): 245-258.

Fayed, Ahmad A., Mohamed B. Trabia, and Mohamed M. ElMadany. "Optimization of Active Suspension Design of a Vehicle." *Proceedings of 2008 ASME International Mechanical Engineering Congress and Exposition (IMECE2008).* Boston, Massachusetts, 2008. 893-901.

AlBassam, Bassam A., Ahmad A. Fayed, and Mohamed M. ElMadany. "Optimal Linear Preview Control of Slow-Active Suspension Systems." *Proceedings of the 7th Saudi Engineering Conference (SEC7).* Riyadh, Saudi Arabia: KSU, 2007. 125-140.

Dissertation Title:

COMPUTER-BASED STEREOSCOPIC PARTS RECOGNITION FOR ROBOTIC APPLICATIONS

Dissertation Examination Committee:

Chairperson, Georg Mauer, Ph.D.
Committee Member, Evangelos Yfantis, Ph. D.
Committee Member, Woosoon Yim, Ph. D.
Committee Member, Brendan O'Toole, Ph. D.
Graduate Faculty Representative, Moses Karakouzian, Ph. D.