

Air Force Institute of Technology

**AFIT Scholar**

---

Theses and Dissertations

Student Graduate Works

---

3-2000

## Realtime Color Stereovision Processing

Byron P. Formwalt

Follow this and additional works at: <https://scholar.afit.edu/etd>



Part of the [Electrical and Computer Engineering Commons](#)

---

### Recommended Citation

Formwalt, Byron P., "Realtime Color Stereovision Processing" (2000). *Theses and Dissertations*. 4786.  
<https://scholar.afit.edu/etd/4786>

This Thesis is brought to you for free and open access by the Student Graduate Works at AFIT Scholar. It has been accepted for inclusion in Theses and Dissertations by an authorized administrator of AFIT Scholar. For more information, please contact [richard.mansfield@afit.edu](mailto:richard.mansfield@afit.edu).



**REALTIME COLOR STEREOVISION  
PROCESSING**

THESIS

Byron P. Formwalt, First Lieutenant, USAF

AFIT/GE/ENG/00M-08

**DEPARTMENT OF THE AIR FORCE  
AIR UNIVERSITY**

**AIR FORCE INSTITUTE OF TECHNOLOGY**

---

**Wright-Patterson Air Force Base, Ohio**

APPROVED FOR PUBLIC RELEASE; DISTRIBUTION UNLIMITED.

20000815 169



**REALTIME COLOR STEREOVISION  
PROCESSING**

**THESIS**

Byron P. Formwalt, First Lieutenant, USAF

AFIT/GE/ENG/00M-08

**DEPARTMENT OF THE AIR FORCE  
AIR UNIVERSITY**

**AIR FORCE INSTITUTE OF TECHNOLOGY**

---

---

**Wright-Patterson Air Force Base, Ohio**

APPROVED FOR PUBLIC RELEASE; DISTRIBUTION UNLIMITED.

**ERIC QUALITY INSPECTED 4**

The views expressed in this thesis are those of the author and do not reflect the official policy or position of the United States Air Force, Department of Defense or the U. S. Government.

AFIT/GE/ENG/00M-08

# REALTIME COLOR STEREOVISION PROCESSING

THESIS

Presented to the Faculty

Department of Electrical Engineering

Graduate School of Engineering and Management

Air Force Institute of Technology

Air University

Air Education and Training Command

In Partial Fulfillment of the Requirements for the Degree of

Master of Science in Electrical Engineering

Byron P. Formwalt, B.S.

First Lieutenant, USAF

March 2000

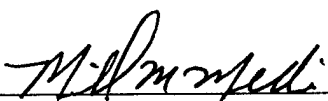
APPROVED FOR PUBLIC RELEASE; DISTRIBUTION UNLIMITED.

REALTIME COLOR STEREOVISION PROCESSING

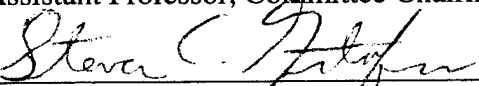
Byron P. Formwalt, B.S.E.E.

First Lieutenant, USAF


Approved:

  
\_\_\_\_\_  
Mikel M. Miller, PhD, Lt Col, USAF  
Assistant Professor, Committee Chairman

14 March 00  
Date

  
\_\_\_\_\_  
Dr. Steven C. Gustafson  
Professor, Committee Member

14 March 00  
Date

  
\_\_\_\_\_  
John F. Raquet, PhD, Capt, USAF  
Assistant Professor, Committee Member

14 March 00  
Date

## ***ACKNOWLEDGEMENTS***

I thank God for all He has blessed me with. I also appreciate the guidance of my thesis committee, which includes Lt. Col. Mikel Miller, Capt. John F. Raquet, and Dr. Steve Gustafson. Thank you, Phil Corbell, for facilitating some of my thought processes and, Brian Bracy, for assisting me with the intricacies of GPS data. I thank my parents for guiding me to this point in my life, and I acknowledge all the fine teachers who inspired me along the way. I especially want to acknowledge Mr. Jim Hurley for first introducing me to the field of machine vision when I was only seventeen years old. Finally, I thank my close friend, Jenn, who lifted me up with her loving prayers and encouragement during the home stretch.

## Table of Contents

	Page
Acknowledgements .....	ii
List of Figures.....	v
List of Tables.....	vii
Abstract .....	viii
1. Introduction.....	1-1
1.1 Background .....	1-1
1.2 Problem Statement .....	1-3
1.3 Summary of Current Knowledge.....	1-6
1.3.1 Applied Knowledge.....	1-6
1.3.2 Theoretical Knowledge .....	1-8
1.4 Assumptions .....	1-14
1.5 Scope .....	1-14
1.6 Approach .....	1-15
1.7 Materials and Equipment.....	1-15
1.8 Thesis Organization.....	1-16
2. Background Theory.....	2-1
2.1 Chapter Overview.....	2-1
2.2 Image Characteristics .....	2-1
2.3 Feature Extraction .....	2-3
2.4 Stereo Imaging.....	2-4
2.5 Feature Analysis .....	2-7
2.6 Chapter Summary.....	2-11
3. Methodology.....	3-1
3.1 Chapter Overview.....	3-1
3.2 System Overview.....	3-1
3.3 Hardware Configuration.....	3-2
3.4 Preprocessing.....	3-5
3.5 Feature Clustering .....	3-9
3.6 Feature Group Correspondence.....	3-10
3.7 Depth Map.....	3-11
3.8 Object Recognition.....	3-18
3.9 Chapter Summary.....	3-22



4. Results and Analysis.....	4-1
4.1 Chapter Overview.....	4-1
4.2 Preprocessing.....	4-1
4.2.1 Future Recommendations.....	4-2
4.3 Feature Clustering.....	4-2
4.3.1 Feature Extraction.....	4-3
4.3.2 Histogram Zooming.....	4-5
4.3.3 Future Recommendations.....	4-7
4.4 Feature Group Correspondence.....	4-8
4.4.1 Future Recommendations.....	4-9
4.5 Depth Mapping.....	4-12
4.5.1 Future Recommendations.....	4-20
4.6 Object Recognition.....	4-26
4.6.1 Future Recommendations.....	4-30
4.7 Conclusions.....	4-30
5. Conclusions.....	5-1
5.1 Summary.....	5-1
5.2 Future Recommendations.....	5-2
Appendix A Emulating Stop&Go.....	A-1
A.1 Overview.....	A-1
A.2 Pixel Classifier.....	A-1
A.3 Correspondence.....	A-6
A.4 Data Validation.....	A-8
A.5 Data Clustering.....	A-10
A.6 Summary.....	A-13
Glossary of Terms.....	GLO-1
Bibliography.....	BIB-1
Vita.....	VIT-1

## *List of Figures*

Figure	Page
1.1 Raven Test Bed and Left-Hand Local Coordinate Frame .....	1-4
1.2 Generic Procedure for Stereo Machine Vision Processing .....	1-5
2.1 Pixel Reference System.....	2-2
2.2 Color Image Representation .....	2-2
2.3 Image Data Structure .....	2-3
2.4 Region Growing .....	2-4
2.5 Contrast Detection—An Image Point Isolation Technique .....	2-5
2.6 Epipolar Constraint .....	2-6
2.7 Ordering Constraint .....	2-6
2.8 Disparity .....	2-8
2.9 Depth Map.....	2-8
2.10 Depth-Assisted Object Detection .....	2-9
2.11 Template Matching.....	2-10
2.12 Template Filtering .....	2-10
3.1 Raven Stereovision Process.....	3-2
3.2 Hardware Configuration.....	3-3
3.3 Stereo Capture Process .....	3-4
3.4 Preprocessing Method .....	3-6
3.5 Vertical Color Band.....	3-6
3.6 Horizontal Color Band .....	3-7
3.7 Color Cross Optimization.....	3-8
3.8 Depth Map Transformation .....	3-11
3.9 Center Row Anatomy .....	3-13
3.10 Top View of Focal Point Projection.....	3-14
3.11 Cross-View Diagram Displaying Vertical Viewing Angles.....	3-15
3.12 Dual Image Plane Diagram.....	3-17
3.13 Multiscale Nominal Frequency-Based Recognition Flow Diagram.....	3-20
3.14 Multiscale Nominal Frequency-Based Recognition—Detail Pruning .....	3-21
4.1 Futuristic Rendition of Miniature Precalibrated Stereo Camera Rig .....	4-3
4.2 Contrast Point Feature Extraction Schemes .....	4-4
4.3 Histogram Zooming Results Overlaid on Right SUV Target Image Frame .....	4-6
4.4 Complete Results of Histogram Zooming for Image Frame of Figure 4.3 .....	4-6
4.5 Blocking Effects on Close Range Targets .....	4-7
4.6 Proper Feature Group Correspondence on SUV Target with One Pixel Error .....	4-8
4.7 Confidence in Target Detection for the First Stop Sign Target.....	4-10
4.8 Confidence in Detection for SUV Target .....	4-10

4.9	Confidence in Detection for Second Stop Sign Target .....	4-11
4.10	Fisheye Effect .....	4-12
4.11	Cross-Hair Camera Alignment Calibration Setup .....	4-13
4.12	Corrected $\alpha$ Parameter Model .....	4-15
4.13	Corrected $\beta$ Parameter Model .....	4-15
4.14	Small Angle Perspective Model Applied to Range Errors .....	4-16
4.15	True Model Errors in $x$ -Direction Relative to Straight Line Distance to Target ...	4-17
4.16	True Model Errors in $y$ -Direction Relative to Straight Line Distance to Target ...	4-17
4.17	True Model Errors in $z$ -Direction Relative to Straight Line Distance to Target ...	4-18
4.18	Relative Corrective Model Errors in $x$ -Direction .....	4-18
4.19	Relative Corrective Model Errors in $y$ -Direction .....	4-19
4.20	Relative Corrective Model Errors in $z$ -Direction .....	4-19
4.21	Relative $x$ -Direction Errors for the SUV Target .....	4-21
4.22	Relative $y$ -Direction Errors for the SUV Target .....	4-21
4.23	Relative $z$ -Direction Errors for the SUV Target .....	4-22
4.24	Relative Width Measurement Errors for the SUV Target .....	4-22
4.25	Relative Height Measurement Errors for the SUV Target .....	4-23
4.26	Relative $x$ -Direction Errors for the Stop Sign Target .....	4-23
4.27	Relative $y$ -Direction Errors for the Stop Sign Target .....	4-24
4.28	Relative $z$ -Direction Errors for the Stop Sign Target .....	4-24
4.29	Relative Width Measurement Errors for the Stop Sign Target .....	4-25
4.30	Relative Height Measurement Errors for the Stop Sign Target .....	4-25
4.31	Object Library and Additional Test Objects .....	4-27
4.32	Threshold Characteristics of Cross Walk Signs .....	4-27
4.33	Threshold Characteristics of Do Not Enter Signs .....	4-28
4.34	Threshold Characteristics of Merge Left Signs .....	4-28
4.35	Threshold Characteristics of Stop Signs .....	4-29
4.36	Threshold Characteristics of Yield Signs .....	4-29
A.1	Pixel Class Correspondence .....	A-2
A.2	Feature-Related Data Structures .....	A-3
A.3	Contrast Pixel Classifier .....	A-4
A.4	Determining Color Likeness .....	A-5
A.5	Color Pixel Classifier .....	A-5
A.6	Correspondence Flow Diagram .....	A-7
A.7	Possible Correspondence Error Scenario .....	A-8
A.8	Histogramming Flow Diagram .....	A-9
A.9	Effects of Bin Sizes on Histogramming .....	A-10
A.10	Reducing Data by Cropping to World Boundaries .....	A-11
A.11	Disadvantages of $x$ and $y$ -Histogramming .....	A-12
A.12	Flow Diagram of Data Clustering Algorithm .....	A-12

*List of Tables*

Table	Page
4.1 Relative Errors in Measurements Before and After Corrective Modeling .....	4-16

## *Abstract*

Recent developments in aviation have made micro air vehicles (MAVs) a reality. These featherweight palm-sized radio-controlled flying saucers embody the future of air-to-ground combat. No one has ever successfully implemented an autonomous control system for MAVs. Because MAVs are physically small with limited energy supplies, video signals offer superiority over radar for navigational applications.

This research takes a step forward in realtime machine vision processing. It investigates techniques for implementing a realtime stereovision processing system using two miniature color cameras. The effects of poor-quality optics are overcome by a robust algorithm, which operates in realtime and achieves frame rates up to 10 fps in ideal conditions. The vision system implements innovative work in the following five areas of vision processing: fast image registration preprocessing, object detection, feature correspondence, distortion-compensated ranging, and multiscale nominal frequency-based object recognition.

Results indicate that the system can provide adequate obstacle avoidance feedback for autonomous vehicle control. However, typical relative position errors are about 10%—too high for surveillance applications. The range of operation is also limited to between 6 – 30 m. The root of this limitation is imprecise feature correspondence: with perfect feature correspondence the range would extend to between 0.5 – 30 m. Stereo camera separation limits the near range, while optical resolution limits the far range. Image frame sizes are 160x120 pixels. Increasing this size will improve far range characteristics but will also decrease frame rate. Image preprocessing proved to be less appropriate than precision camera alignment in this application. A proof of concept for object recognition shows promise for applications with more precise object detection. Future recommendations are offered in all five areas of vision processing.

# ***REALTIME COLOR STEREOVISION PROCESSING***

## ***1. INTRODUCTION***

### ***1.1 BACKGROUND***

Since the 1970s, image processing has been a hot topic on the fine edge between science and computing. Recent acceleration of computer technology has hailed an ongoing explosion in image processing research. Research began to escalate considerably in the late 1990s. The reason is simple—we now have the capacity to perform image processing real time, eliminating the need to spend hours or days on what the human eye can do in a split second.

The forerunner of digital image processing originated in the 1960s when National Aeronautics and Space Administration (NASA) Jet Propulsion Laboratory (JPL) designed a system to enhance images from the Ranger 7 mission [6:410]. This system corrected for image blurring, geometric distortions, and other sources of background noise [6:410]. Image processing has also found its way into many other fields, including medicine, biology, agriculture, physics, forensics, and geography. The research reported here applies image processing to automatic navigation and control of mobile defense systems. More specifically, the focus is on adding autonomy to cutting-edge technology already invested in micro air vehicle (MAV) design.

According to *Aviation Weekly*, the first MAV was constructed in 1997 as part of a \$35 million, four-year DARPA (Defense Advanced Research Projects Agency) effort to develop the latest in defense weaponry [2]. The current model operates for a minimum of 10 minutes with a 1-km radius. By March 2000 the MAV is expected to have a 3-km operating radius with 20 minutes of continuous flight.

Some applications for MAVs include micromunitions and covert surveillance. According to former Air Force Explosive Ordnance Disposal (EOD) Technician, B. Reece Tredway, a 20 g payload of plastic Composition-4 (C-4) explosive is sufficient to self destruct an MAV along with a sizeable enemy communications circuit [7].

Rapid advances in MAV technology have created the demand for an advanced control mechanism, allowing MAVs to fly with at least partial autonomy. DARPA's program manager for MAV research, James McMichael, has said that autonomous control is a major worry for the future of the program [2]. It makes good sense to develop a *vision system* as the core element of a control mechanism for such a platform. There are two reasons backing this up: 1) Vision systems use cameras—passive devices allowing a vessel to operate covertly; 2) Cameras are lighter and more efficient than other remote sensing elements. Bulky radar technology concedes to vision processing systems on both accounts.

The vision system developed in this research offers an alternative to radar, bridging the gap between traditional aircraft technology and more recent MAV

technology. By using a color stereovision apparatus, this system detects and locates objects in front of the vehicle. In the context of an MAV platform, the video would be transmitted back to a control center for processing. Then the control center would return the proper feedback required to appropriately adjust the motion of the aircraft. In this research, however, the test bed being used is not an MAV, and therefore it harbors a rudimentary onboard control system. The name given to this platform is “The RAVEN”—Remote-sensing Autonomous Vehicle ENgineering test bed. The RAVEN is actually a Club Car® golf cart, equipped with two additional batteries, two color cameras, an on-board dual processing computer with video capture board, and a low-powered liquid crystal display (LCD) heads-up display (HUD). The RAVEN is shown in Figure 1.1. Note the relative left-hand coordinate frame with the origin at the center of the right camera. In the future, the car will be modified to contain automatic steering, acceleration, and braking capabilities. Future research will focus on integrating the vision system with inertial navigation system (INS) and differential global positioning system (DGPS) data to extend the platform’s navigational capabilities.

## ***1.2 PROBLEM STATEMENT***

Research on real-time vision processing systems is directly applicable to cutting-edge military technology since MAV technology faces an impending limit until a control mechanism, sophisticated enough to provide autonomy, is developed. The process



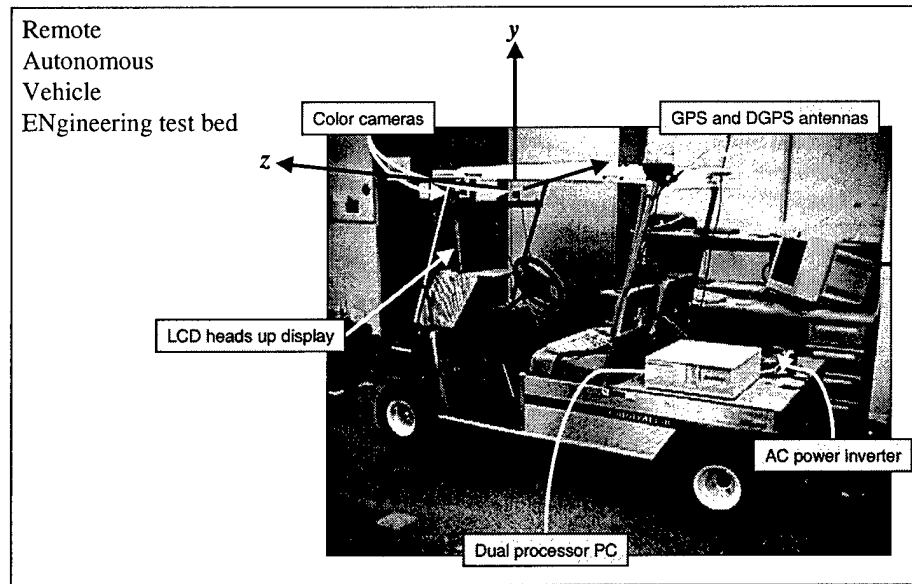


Figure 1.1 RAVEN test bed and left-hand local coordinate frame.

performed by any stereo vision system is fivefold. Figure 1.2 diagrams this generic process, as explained below.

The first step is feature extraction. During this phase, essential information is extracted from the stereo images in order to compare the left and right image pairs. Optionally, this step may involve some form of preprocessing to correct for optical distortions.

The next step is to find correspondences between the two feature sets from each image. This involves taking a feature from one image and searching for it in the other. The problem is difficult because some features are not in both images. It is also very easy to have false features appearing in both images. If improperly handled, these two

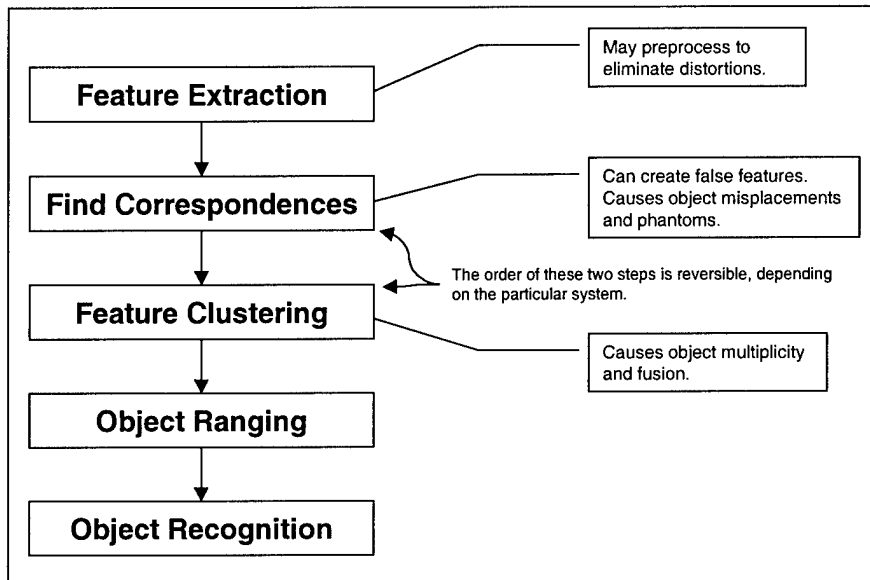


Figure 1.2 Generic procedure for stereo machine vision processing

problems can confuse the vision system into thinking that objects are misplaced. It can also convince the system that an object exists when it really does not. Such objects are called *phantoms*. This research introduces some advances, discussed later, that ensure stability for resolving these problems.

The third step in processing stereo image pairs is to cluster features in such a way that each cluster contains all the features representing a particular object. Errors in this step create multiple objects where only one really exists, or they can cause the system to mistake several smaller objects for a single larger object. Some systems, including the one onboard the RAVEN, perform the second and third steps in reverse order.

Fourth in the process is object ranging. In this step, the real-world boundaries for each object are determined, as well as object locations relative to the local coordinate frame.

The final step is object recognition. In this step the system attempts to recognize objects using some appropriate pattern recognition algorithm. This research reveals innovative techniques for each of the five steps in the process.

### ***1.3 SUMMARY OF CURRENT KNOWLEDGE***

In order to recognize the contributions of this research, it is important to examine current technology first. This section is dedicated to that purpose. It has been organized into two subsections—one outlines applications of current technology, and the other discusses the status of relevant theoretical knowledge.

#### ***1.3.1 APPLIED KNOWLEDGE***

In Germany, Daimler-Benz has made extensive progress in the field of autonomous vehicle control. Their Stop&Go system, powered by a network of four computers in a parallel configuration, enables a car to navigate its way autonomously through city traffic. It uses a vision system to recognize pedestrians, road signs, street markings, stoplights, and traffic patterns [3:40-41]. Most improvements to stereo machine vision presented in this research are modifications to current processes used by the Stop&Go system. The research reported here does not yet have autonomous control

mechanisms comparable to that of Stop&Go, but all the necessary bare-bone components are in place for improving the vision techniques used by that system.

NorthropGrumman has been developing a GPS-driven UAV (Unmanned Aerial Vehicle), named the Global Hawk Navigation System [4:1]. The technology used by Global Hawk might one day be combined with a vision system to improve navigational capabilities. NAVSYS Corporation has a video feedback system closely tied to this emerging technology [1:1]. This system fuses GPS and INS data for attitude and positioning determination [1:2]. Then the data is mapped to a database, containing locations of all known objects in the vicinity. Objects are pulled from the database and displayed on a HUD in the exact manner in which they would have been seen through the windshield under ideal visibility.

NAVSYS is developing this system in contractual agreement with the US Army and the ONR (Office of Naval Research) [1:8]. It has direct applications for MAV technology. Consider a fleet of networked MAVs. This low-profile flying team could swarm a tactical location, fusing their observations onto a central GPS-based map. Each MAV could provide coordinates and descriptions of all objects encountered. The resulting map would be a three-dimensional representation of the site encountered. This map could then be used to generate war-fighting simulations and thus more effective attack plans. Eventually, it could also allow critical targets to be accurately modeled at full scale for practicing an actual attack.

### **1.3.2 THEORETICAL KNOWLEDGE**

Theoretical knowledge supporting stereo vision systems falls into six major categories. The first category is *edge detection*. An edge detector creates a line drawing from of a photographic image. Perhaps the most popular theoretical contribution in edge detector design is a remarkable innovation credited to John Canny [8]. While there are at least a half a dozen other edge detectors available, most current systems seem to favor the characteristics of the Canny edge detector. The purpose of edge detection is to reduce the overall data set being used for dynamic vision processing, which speeds up the processing.

Another contribution that increases the utility of edge-detection is a polygonization procedure developed by T. J. Davis [9]. This algorithm accepts an edge-detected image as input and generates sets of endpoints for each edge as output. This step is useful because it further reduces the volume of data required for processing. Other techniques, such as artificial neural networks [10] and variable-scale smoothing [11], have been proposed for accelerating and enhancing edge detection. This research does not incorporate edge detection techniques because they are computationally too expensive to use in the system. A dedicated edge-detection/polygonization image-processing chip, however, may provide the additional computing power needed to make edge detection a viable alternative.

*Object detection* is another aspect of machine vision. Again, Stop&Go is recognized for its powerfully simple histogramming technique for detecting objects [12:340]. A completely different technique for object detection is presented in [13]. This novel method uses Haar wavelet energy distributions as inputs to a multi-layer perceptron (MLP) neural network for locating objects. The results are surprisingly accurate, but the computational intensity for wavelet processing is also high. Todd Williamson, et al., developed a specialized multibaseline stereo technique for obstacle detection [14]. This technique also has merit in its simplicity. A drawback is that it operates on the entire image instead of a small set of feature points. However, the multibaseline technique could be adapted to operate on only a subset of image points. The technique examines pixel disparities and calculates real-world positions, assigning a new pixel values which increase monotonically with both relative depth and height. The result is an image highlighting vertical objects.

*Disparity estimation* is the next category of stereo machine vision theory and is the most difficult aspect of vision processing. The *disparity* of a pixel is the horizontal shift of that pixel between the left and right image frames. The *epipolar constraint* is the notion that *corresponding* pixels between left and right image frames cannot have vertical offsets between them. The *perspective constraint* mandates that pixels in the left image frame corresponding to pixels in the right image frame must be shifted to the *right* of the corresponding pixel positions in the right image frame. Each of these constraints

simplifies the *correspondence problem*—the problem of determining which pixels correspond and what their disparities are. Disparity estimation is not as difficult as the correspondence problem that precedes it.

Stop&Go classifies every point in the image into eighty-one categories and then performs a columnwise search for correspondences within each category. Correspondences which generate the least disparity are chosen where discrepancies exist. While this approach is straightforward and simple, there are at least four others worth mentioning.

One alternative is to bypass the problem completely. Shinsaku Hiura and Takashi Matsuyama have experimented with a multifocus camera [35]. By determining the “blurriness” of various regions in the image at different foci, they determine a depth map of an image with some success. Their system does not require processing two images, but it does require the use of a specialized camera capable of capturing three images simultaneously, where each of the images is focused differently.

A novel method, developed as part of this research, determines the disparity between regions of homogeneous color. This method had limited success due to varied lighting effects, causing poor interimage color correspondence. No amount of color preprocessing corrected this problem, so the idea was abandoned. Rabie Tamer, et al., devised a method for aligning multiple images into a mosaic [15]. If this method had not been so computationally intensive, it may have been implemented to align corresponding

colored regions between left and right image frames. Another technique that failed was a modification to the Stop&Go contrast pixel classifier that used color channel variance and norm color differences to perform pixel classification. While this method did a good job of identifying key pixels in each image, it did not lead to robust feature correspondence. See Appendix A for details.

*Object recognition* is the next category of theoretical research supporting vision systems. Since the RAVEN is operating in a road-centered environment, it is prudent to examine road sign recognition techniques. Dan Ghica, et al., [16] and Giulia Piccioli, et al., [20] have both researched this specific application of object recognition.

More generally, D. Ernst, et al., is developing a classifier which extracts not only geometric data from tracked objects but also motion parameters of individual parts of the objects to assist in their recognition [17:1].

In 1991, Si Wei Lu and Andrew Wong outlined a method for recognizing partially occluded objects by a hypergraph representation [19], which means that they combined search techniques with graph theory and mapped it to the domain of occluded objects. Their work, although ahead of its time, was somewhat sketchy, and therefore not useable for application onboard the RAVEN.

D. M. Gavrila and L. S. Davis have researched a very fast phase-coded filtering method for correlating images from large databases [18]. This technology could be applied to locate objects of interest within a single image. In this adaptation a



predetermined target would be identified more quickly. The alternative is to identify every object detected and then determine whether the desired target has been acquired—a much more time-intensive process.

*Object tracking* is another discipline serving machine vision. Stop&Go uses a Kalman filter to estimate future motion of objects through a sequence of image frames [12:342-343]. Li Biao, et al., [21] and Haixin Chen, et al., [22] are two groups that have examined tracking methods using infrared imagery. Biao's team uses a weighted-average filter for tracking, while Chen's uses a variation of match filter tracking. Match filter tracking locates a point of maximum correlation of an object from one image shifted over a sequential image. T. Darrell, et al., [23] has succeeded at person tracking using apriori knowledge about the structure of the human body to extract and identify features of humans.

*Structure from motion* is a process where multiple frames in an image sequence allow for complete three-dimensional modeling of objects encountered. The structure is implied from observations of motion. John Oliensis [31] developed an algorithm that not only accomplishes this task but also corrects for situations where perspective effects are large. Unfortunately, his method is unacceptably slow.

Gideon Stein and Amnon Shashua [26] devised a unique twist to the traditional structure-from-motion problem. They used spatio-temporal derivatives between two sequential stereo image pairs to fuse the stereo structure information from two pairs of

sequential frames. This is done by removing the estimated motion effects determined from the optical flow of the sequence.

Another powerful technique related to the structure-from-motion problem uses multiple images from a sequence to segment the image into regions [30]. Each region corresponds to an object. This technique could be used with the proposed region disparity estimation method to utilize color information in stereo processing more fully. A drawback is that there must be relative motion between objects and the detector. Another fault is that the background is a difficult feature to identify for removal from the process.

Christopher Eveland, et al., [29] developed a method for statistically modeling foreground and background behaviors which is intended to solve this problem. For a brief explanation of the original structure-from-motion algorithm, refer to Min Shin and Kevin Bowyer's study, which uses this technique for comparing various edge detectors [32:191].

One final idea, related to the structure-from-motion category and created by John Oliensis [25], is to determine camera heading from multiple image frames. The difficulty with this and most of the other methods presented in this section is that they are computationally expensive. This observation does not mean they will never be applied to realtime stereovision; it just means that technology is not advanced enough to implement them yet.

#### ***1.4 ASSUMPTIONS***

It is assumed that no objects in front of the camera are distorted by the camera lens. It is also assumed that the cameras are oriented such that the ground is below the sky in all images and that the cameras are aimed parallel to each other in a direction normal to the front plane of the test bed. All objects of interest are assumed to contrast against their backgrounds and to have rectangular form with one pair of edges normal to the local ground plane. Color sensitivity is assumed to be identical for each camera, and it is assumed that no specular variations exist between left and right image frames. One final assumption is that left and right image frames are captured simultaneously, with all objects of interest appearing within the field of view (FOV) of both cameras.

#### ***1.5 SCOPE***

This research implements a stereo vision-processing scheme that accomplishes all of the five steps outlined in Section 1.2. Contributions are presented for each step of the process. No attempt is made to perform object tracking, and no GPS/INS data is being incorporated at this time. Innovative techniques are proposed for assisting camera alignment calibration, target detection, the correspondence problem, and object recognition. Shortcomings are explained, and suggestions are made for further refinement. Accuracies are well-documented and interpreted for object detection, location, and recognition. The resulting system is a realtime vision processing apparatus

that can be used readily to demonstrate the effectiveness of the principles detailed by this document.

## ***1.6 APPROACH***

The approach taken by this research was to design the requirements of the vision system and then to acquire appropriate hardware for implementing the design. After familiarization with the hardware, a large number of vision processing techniques were investigated for possible application to the project. The techniques used by the Daimler-Benz Stop&Go system were selected for implementation and analysis for baseline comparison. Unfortunately, the emulation of Stop&Go was unsuccessful. Then a new vision process was developed using techniques which are all potentially original works of the author (a continuing comprehensive search for similar techniques is in progress). While time constraints dictated that the RAVEN could not yet harbor a fully autonomous control system, much progress has been made toward this goal.

## ***1.7 MATERIALS AND EQUIPMENT***

The RAVEN consists of a battery-powered Club Car® golf cart with two auxiliary 12 V marine recreational continuous-duty lead acid batteries configured in series. A 24 V charger accompanies this battery set, while a separate charger is used for the car batteries. A 120 VAC power inverter operates on auxiliary power and is used to run an overclocked, homemade, dual 500 MHz Celeron computer with 128 MB RAM

and an internal high G-rated laptop harddrive. The computer runs Windows NT® 4.0 SP5 with MATLAB® 5.3 and Microsoft® Visual C++® 6.0. The computer also houses a Matrox® Meteor II® frame grabber. Two Marshall® V-1246T ¼” CCD color cameras are mounted on the front of the car and fed into the frame grabber. The cameras have fixed focal lengths of 3.7 mm. Maximum horizontal and vertical viewing angles are 57° and 41°, respectively. A flat-panel LCD monitor rests above the steering wheel in a bracket, which was made in-house. An Ashtech® differential Global Positioning System (DGPS) receiver collects relative positioning data from a local ground station for determining errors in vision measurements. Other standard miscellaneous tools were also used in modifying the platform.

## ***1.8 THESIS ORGANIZATION***

Chapter 2 reviews background theory as a prerequisite to understanding the methodology given in Chapter 3. Chapter 3 provides a detailed description of the ideas that were developed during this research. Chapter 4 presents qualifying results which validate the adequacy of this research, and Chapter 5 summarizes key contributions and proposes future endeavors. The Appendix contains information about the process of emulating the Stop&Go system for purposes of baseline comparison.

## **2. BACKGROUND THEORY**

### **2.1 CHAPTER OVERVIEW**

This chapter offers a tutorial in image processing fundamentals applicable to this research. Topics covered include image characteristics, feature extraction, stereo imaging, and feature analysis.

### **2.2 IMAGE CHARACTERISTICS**

An image is a two-dimensional sequence of real numbers between zero and one. Color images are really three images blended together. Each of the three images is assigned to a primary color—red, green, or blue. Every element of the sequence is called a picture element or pixel. The value of the each element corresponds to the brightness of the pixel. In this document, brightness and intensity are used interchangeably. Pixels are referenced by horizontal and vertical offsets within the image. Figure 2.1 illustrates the pixel ordering system. The ordering system is zero-based, with row numbers increasing downward and column numbers increasing to the right. Rows are designated by the abscissa while columns are indicated in the ordinate position.

Color images have three color planes. The first is red; the second is green; and the third is blue, as shown in Figure 2.2. Alternatively, a color pixel may be viewed as having three channels, each corresponding to a color plane. This alternative representation is also depicted in Figure 2.2.

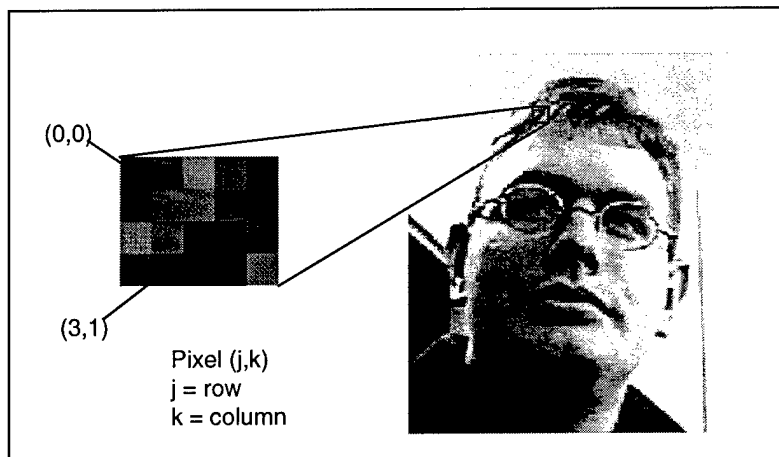


Figure 2.1 Pixel reference system

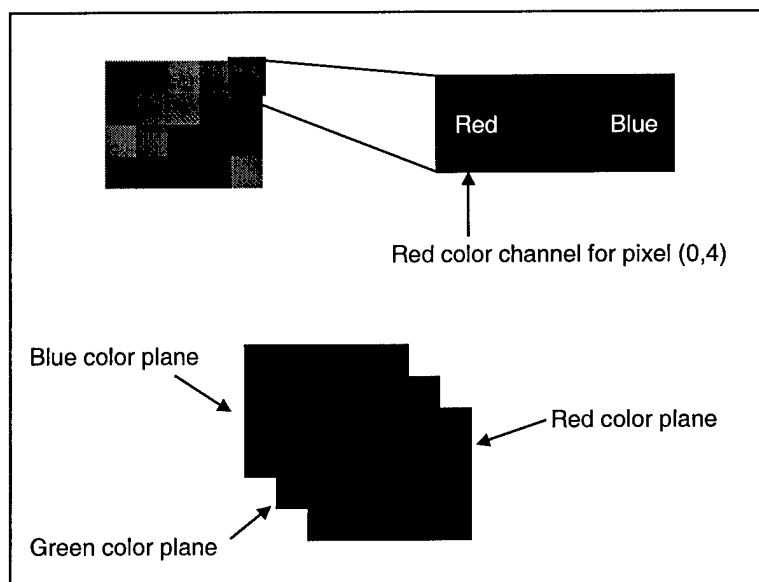


Figure 2.2 Color image representation

In a computer, images are stored in buffers. An image buffer is simply an array of pixel data stored contiguously by rows. This idea is illustrated in Figure 2.3.

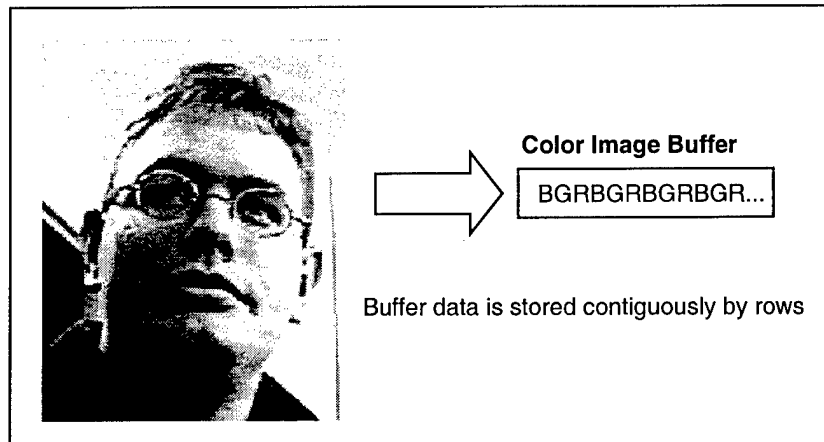


Figure 2.3 Image data structure

### 2.3 FEATURE EXTRACTION

Images are inherently expensive to process. Image data sets are reduced by a process called feature extraction. Feature extraction is the process of identifying and stripping key characteristics from a data set. These features are kept while the rest of the data is discarded for some or all of the remaining processing tasks.

Image segmentation is one category of feature extraction. Region growing is one common form of image segmentation. Region growing, as illustrated in Figure 2.4, begins by choosing a seed pixel at random. After the seed pixel is compared to its neighbors, the region grows in directions of homogenous color. When the region stops growing, a new seed pixel is chosen and the process repeats. The result is a small set of regions which segment the entire image according to color. Examples of other common forms of image segmentation are polygonization [9] (via edge detection) and split-merge



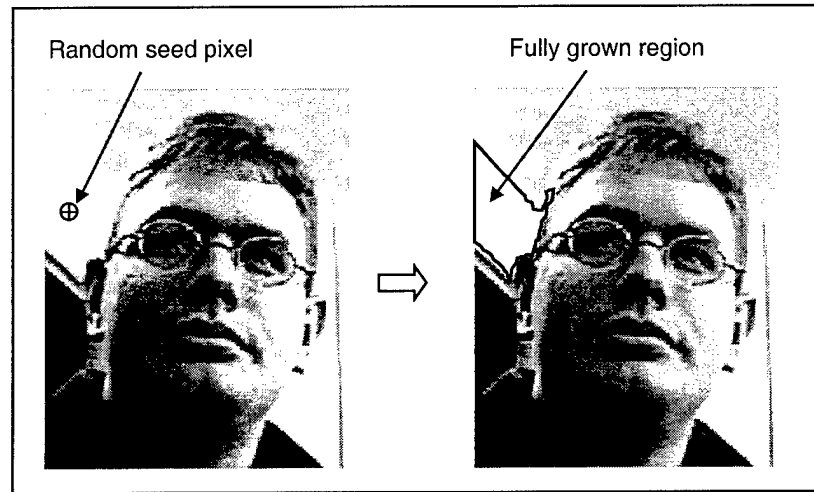


Figure 2.4 Region growing

[34:461-464]. In general, segmentation methods are computationally expensive, so they are not used in this research.

This research uses a technique from the other class of feature extraction. This class is called image point isolation because features correspond to isolated points versus segments. The form of image point isolation used by Stop&Go is contrast detection [3]. Contrast detection extracts pixels significantly different than their neighbors. Figure 2.5 shows features detected from a test image. Note that contrast features always occur in adjacent pairs, which follows from the fact that pixels are mutually similar or dissimilar.

## 2.4 STEREO IMAGING

Stereo imaging introduces additional image processing concepts. Stereo imaging is fusing information from two simultaneously captured images to obtain depth

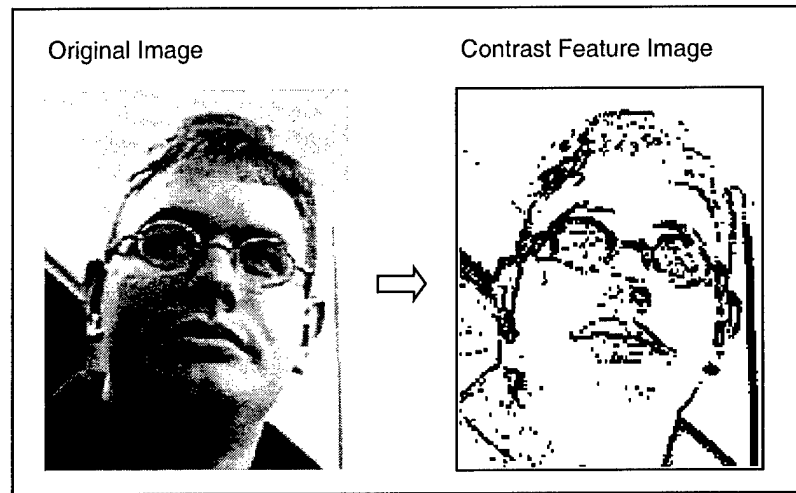


Figure 2.5 Contrast detection—an image point isolation technique

information. Two constraints of stereo imaging accelerate the information fusion process. The *epipolar constraint*, shown in Figure 2.6, preserves the order of corresponding rows between left and right images [12]. If the images are perfectly aligned, so are all their rows.

The *ordering constraint*, shown in Figure 2.7, requires the order of pixels within a row to be the same for two corresponding rows [3]. This constraint is an assumption, not a natural characteristic of all images, and it assumes that objects do not completely trade locations between the left and the right image. In general this assumption is valid. It works especially well for wide objects at relatively equal depths in the image.

Feature correspondence is the essence of stereo imaging. The previous section outlined feature extraction. Feature correspondence is deciding how to correctly pair

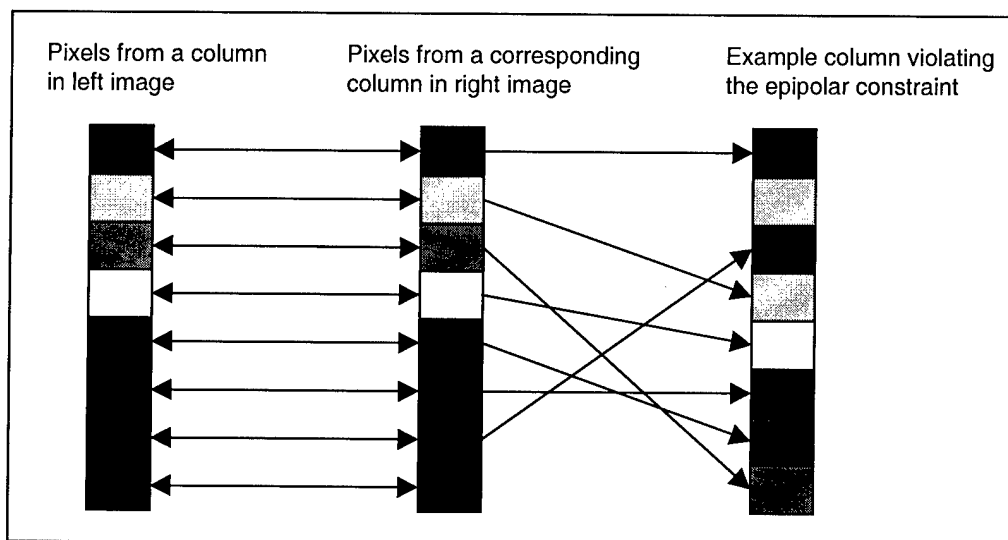


Figure 2.6 Epipolar constraint

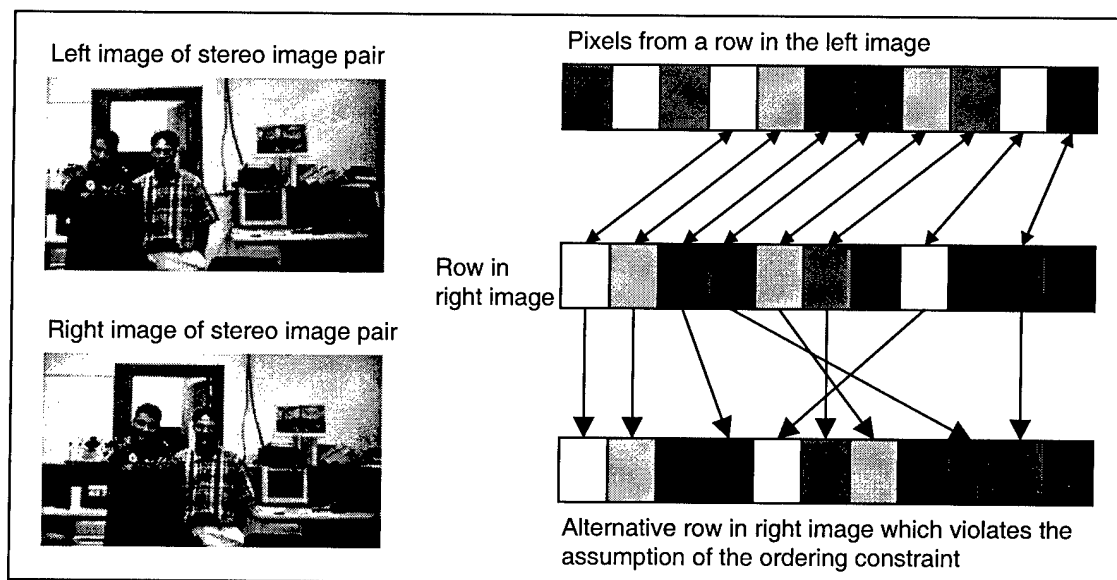


Figure 2.7 Ordering constraint (assumed)

features extracted from the two images and is the most difficult problem encountered in stereo imaging. If perfect feature correspondence were guaranteed, then stereo vision systems would extend to abundantly more applications. Although feature correspondence was the primary error contributor, it does not capture the focus of this research.

Feature correspondence results in a quantity called *disparity*. The disparity of a feature is the horizontal offset for that feature between two images. This concept is diagrammed in Figure 2.8.

A feature's disparity is the basis for determining the real-world coordinates of that feature. When depth information, obtained from feature coordinates, is superimposed on the image data, the result is a depth map for that image. A depth map generally does a good job isolating individual objects in an image. The example shown in Figure 2.9 illustrates how the depth map does not always isolate objects. This research also assumes that in outdoor settings objects are far enough apart to be resolved individually.

## **2.5 FEATURE ANALYSIS**

Object detection becomes trivial when it is assumed that individual objects are always resolvable from any given depth map. Detecting an object is merely locating contiguous regions of similar depth. If the depth map is coarse and complete, as in Figure 2.10, this process is identical to the image segmentation problem. Completeness implies that every pixel in the image has associated depth information. Unfortunately, depth maps are never coarse and complete. Chapter 3 outlines how to use histogramming

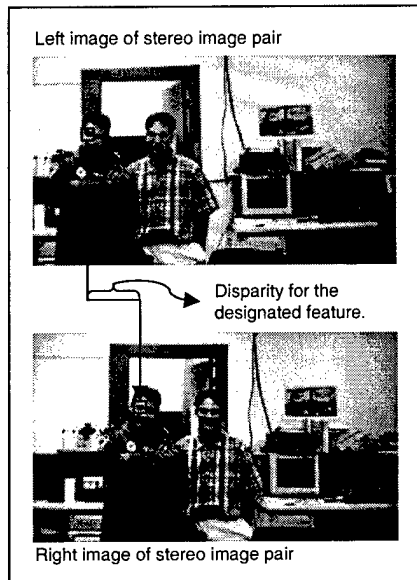


Figure 2.8 Disparity

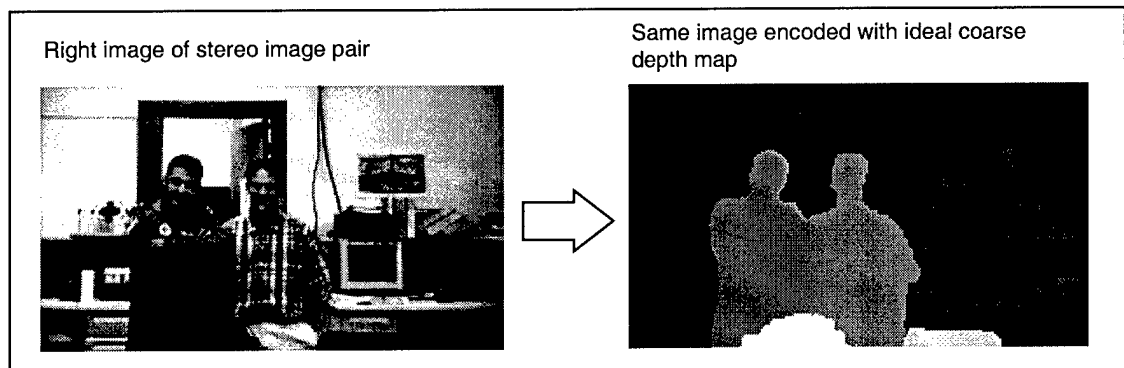


Figure 2.9 Depth map

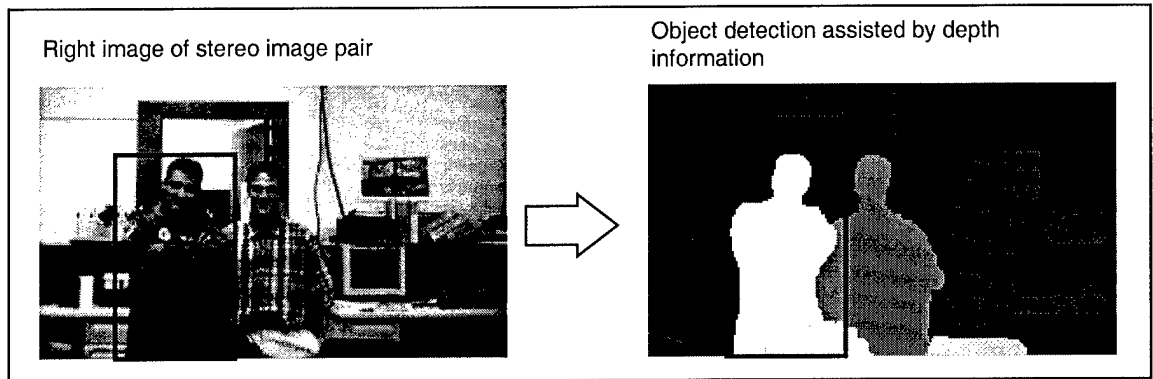


Figure 2.10 Depth-assisted object detection

to solve this problem.

Seldom does a vision system ever have a requirement for detecting objects without classifying them in some sense. This classification process is called object recognition. The two basic types of object recognition are *template matching* and *template filtering*. Template matching compares object features to features of various object templates. The collection of object templates is called an *object library*. Template matching associates the object with the closest matching object template within acceptable tolerance. Figure 2.11 illustrates the concept of template matching.

Template filtering methods differ from template matching methods because they compare features from the entire image to every object template in the object library. Figure 2.12 illustrates this method. Each object template filters the entire set of image features at once. This filtering results in a series of *probability density images (pdi)* for each object in the library. The local maxima of each pdi above some tolerable threshold

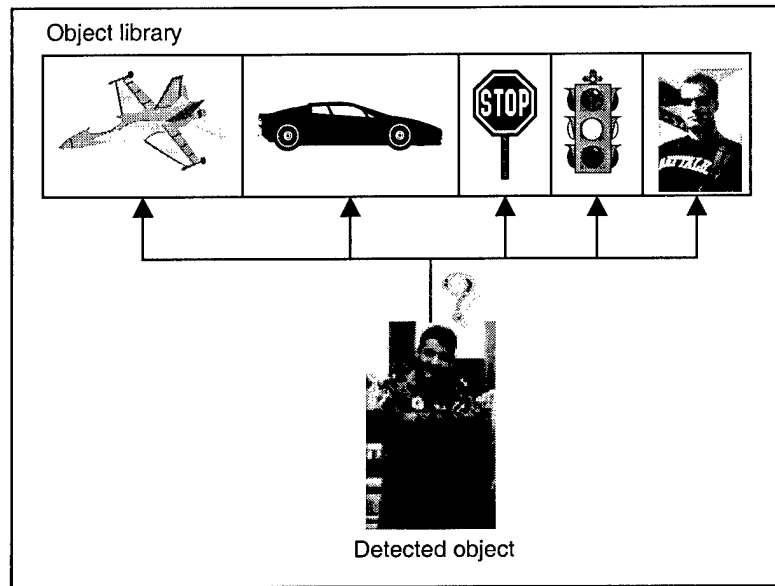


Figure 2.11 Template matching

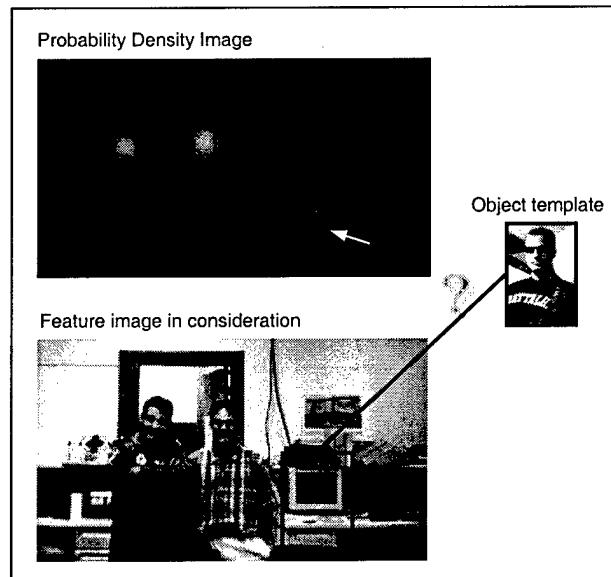


Figure 2.12 Template filtering

are identified as matches for the template used in generating the pdi. While template matching is efficient for classifying multiple objects in an image, template filtering is better for determining whether an image contains a given object.

## **2.6 CHAPTER SUMMARY**

This chapter provided an overview of the basic image processing concepts required for understanding the core contributions this research offers. The major concepts are image characteristics, feature extraction methods, stereo imaging, and object analysis. Chapter 3 explains how these concepts explicitly tie into this research.



### **3. METHODOLOGY**

#### **3.1 CHAPTER OVERVIEW**

This chapter considers theoretical methods and the contributions of this research. It outlines new methods for preprocessing, feature clustering, feature group correspondence, depth mapping, and object recognition.

#### **3.2 SYSTEM OVERVIEW**

The RAVEN vision system is a computer program divided into ten steps. This ten-step process is shown in Figure 3.1. In the first step a pair of stereo images is captured in computer memory. One of the images is immediately rotated and shifted vertically to compensate for any camera misalignment. Next both images are processed for contrast feature extraction using an image point isolation technique. Then features are clustered together in a process called histogram zooming. These feature groups or *clusters* are assumed to represent whole objects. Features groups are made to correspond between left and right image frames. Some of these pairings are inevitably erroneous, requiring the elimination of some correspondence data. A unique multiscale nominal frequency signature is generated for each object to fully characterize the object in highly compressed form. Object signatures are compared to object library templates and organized in variable-branch tree form. If a tolerable match is found the object is considered identified. Finally, all objects (identified or not) are graphically represented to

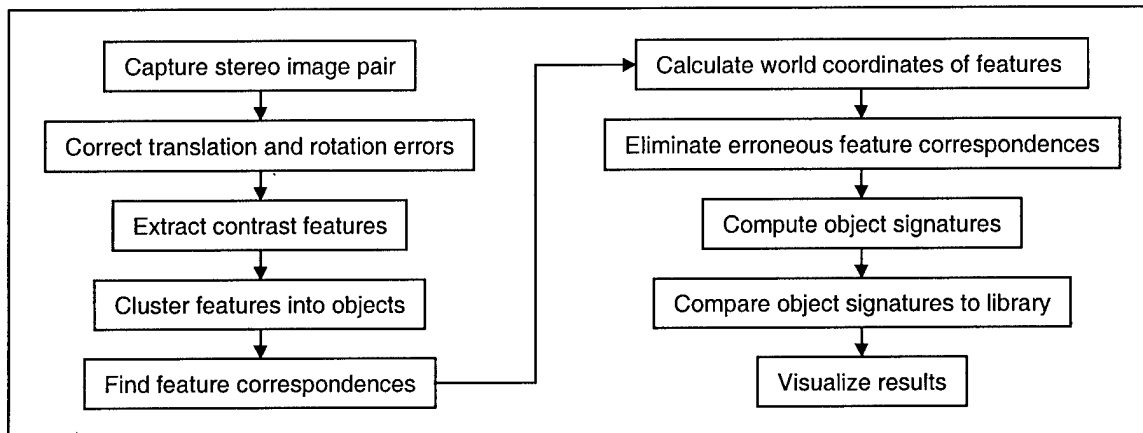


Figure 3.1 RAVEN stereovision process

the user in real time. The object recognition step described is not yet fully implemented in the realtime system. Instead, a proof of concept was developed and evaluated.

### 3.3 *HARDWARE CONFIGURATION*

The RAVEN is equipped with both onboard AC and DC power supplies, providing an easy way to transfer the vision system hardware on and off the RAVEN. The vision system consists of a dual processor computer equipped with a PCI card frame grabber. Ideally, a laptop would be used, but (presumably) no portable economical configuration supports both dual processing and stereo frame grabbing. The solution is to use a power inverter, allowing normal devices to be used with little-to-no modification. One necessary modification was to install a laptop harddrive inside the computer to increase the G-rating of the system.

A diagram of the RAVEN hardware configuration is shown in Figure 3.2. Power

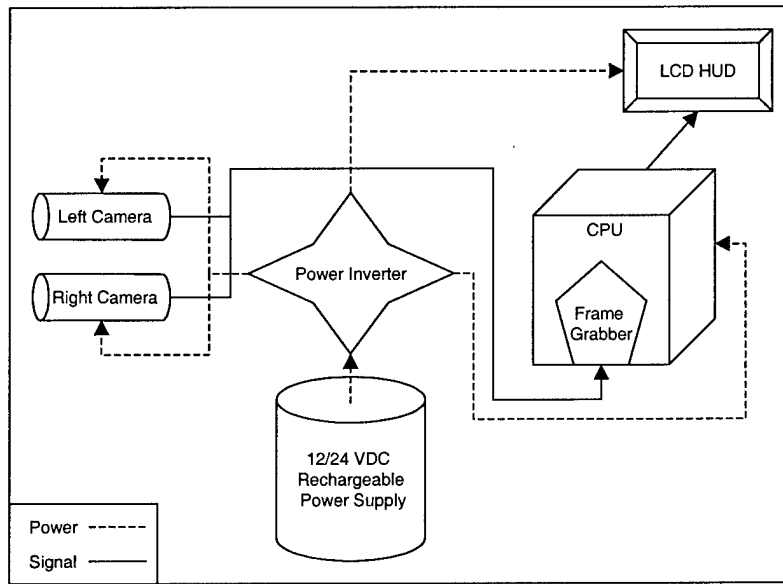


Figure 3.2 Hardware configuration diagram

is indicated by dashed lines and solid lines depict signals. The LCD HUD is a flat-panel 13.3" LCD monitor mounted directly above the steering wheel. Two color cameras are mounted on a bracket that runs crosswise in front of the canopy. The overall system is shown in Figure 1.1.

Due to limited funding, a frame grabber capable of continuous stereo color capture could not be obtained, complicating matters for the data acquisition phase. To compensate, an affordable multi-channel color capture board was selected. The purchased system did not meet manufacturer specifications, but the procedure outlined in Figure 3.3 proved to solve this complication. First the left camera is selected as the input device. Then a command is issued to wait for any previous asynchronous grabs to finish.

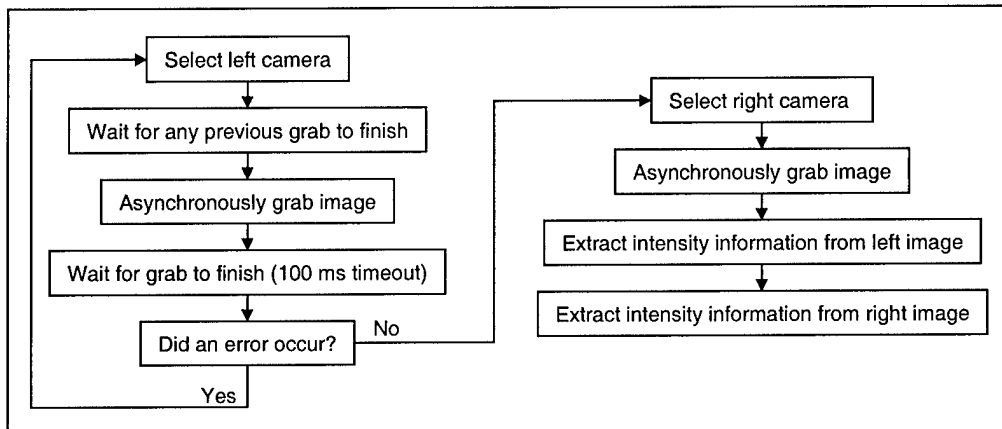


Figure 3.3 Stereo capture process

This step is a precautionary measure taken to compensate for occasional acquisition errors. Next an image is captured *asynchronously* from the left camera into memory. Then the system waits for the capture to complete. The image is captured asynchronously because significant additional overhead is associated with toggling between asynchronous and synchronous modes. The second image needs asynchronous capture for optimum performance. If the hardware times out after 100 ms, then successive attempts are made to recapture an image. An important note is that whenever the first image capture is errorless, the second image capture never times out—another design fluke of the capture board. After the left image is successfully captured, the right image is captured asynchronously while the left image is averaged across its color planes to create a supplementary grayscale image. Given the current processing speed, it is not necessary to issue another wait command before forming another supplementary grayscale image from

the right image. The right image capture is always complete by the time the left grayscale image is finished. Future research will presumably take advantage of more reliable data acquisition technologies.

### ***3.4 PREPROCESSING***

The need for preprocessing stems from the need for a self-calibrating vision system. Without preprocessing stereovision systems require precision alignment to take advantage of the epipolar constraint, meaning camera orientation about all three axes needs to be precise. By correcting for translation and rotation in preprocessing, this calibration reduces to the  $y$ -axis only. The  $y$ -axis shoots vertically upward from the platform; the  $x$ -axis points from left to right; and the  $z$ -axis is directly in front of the platform. Figure 1.1 shows the local coordinate frame.

This research presents an innovative preprocessor design, targeted for realtime systems. Without the assistance of a video-processing chip, the design is required to be extremely lean (computationally speaking). The idea, illustrated in Figure 3.4, uses the least squares to find the corrective line by comparing three vertical strips of each image.

Before delving into the mechanics of the actual technique, a few new concepts need to be introduced. The first new concept is that of a color band. A color band is a three-plane pixel vector, representing an image, or subimage. Figure 3.5 illustrates a vertical color band, while Figure 3.6 illustrates a horizontal band. Color bands are

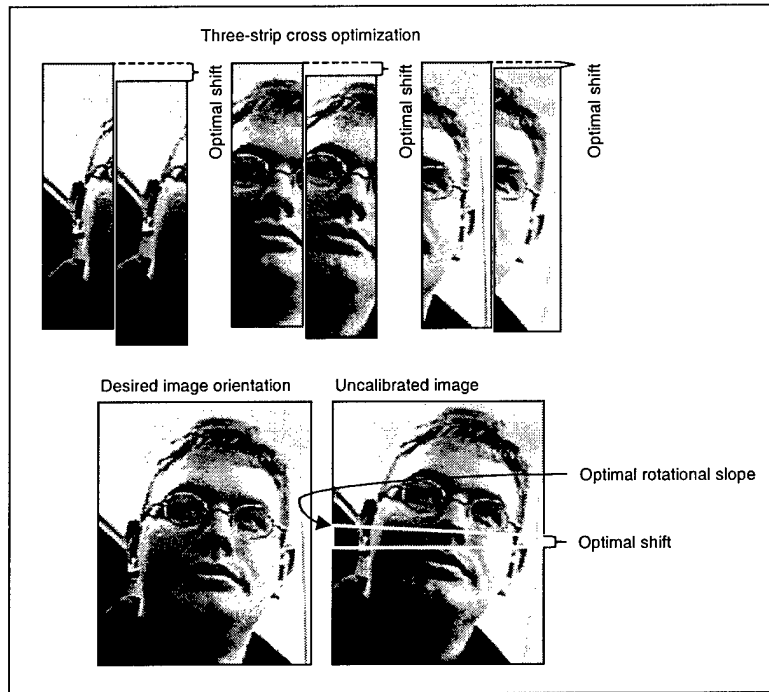


Figure 3.4 Preprocessing method

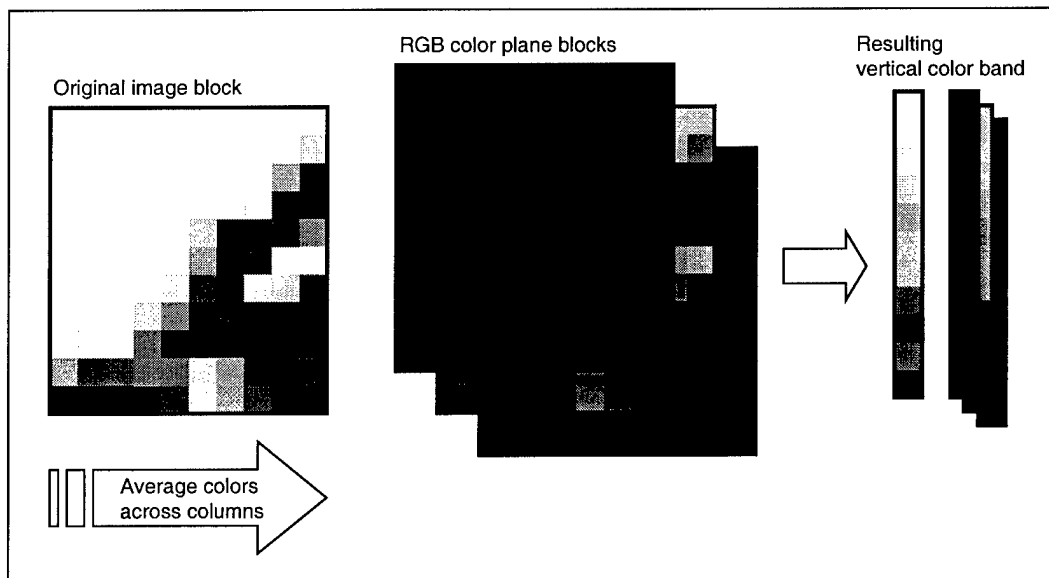


Figure 3.5 Vertical color band

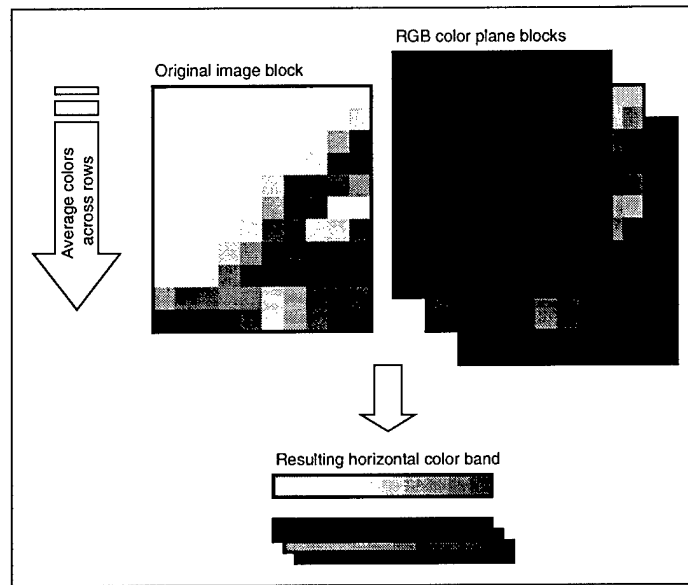


Figure 3.6 Horizontal color band

formed by averaging across either the rows or columns of an image or subimage. Averaging across rows produces a horizontal color band, and averaging across columns produces vertical color bands.

Color cross optimization is the process used to compare two color bands. The process is visually represented by sliding one color band past another until corresponding colors are best matched. Figure 3.7 shows the color cross optimization technique. The measure of error used for determining optimal alignment is called color distance. The color distance between any two colors is calculated by taking the norm difference of all color channels:

$$colordist = \sqrt{(r_a - r_b)^2 + (g_a - g_b)^2 + (b_a - b_b)^2} \quad (3-1)$$

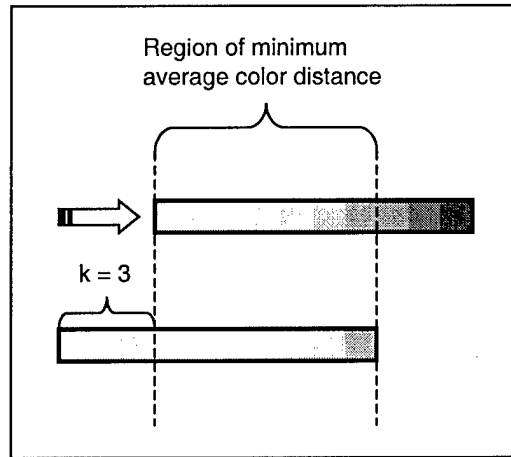


Figure 3.7 Color cross optimization

The total error function is the mean color distance of all pixels in the region of overlap. The purpose of color cross optimization is to determine the optimal shift needed to align two colored regions by minimizing the total error function.

Color cross optimization is the essence of the RAVEN vision preprocessor. Figure 3.4 outlines this process. First the left and right images are divided into three equal-width vertical strips. Each strip is converted to a vertical color band. Corresponding color bands are color cross optimized over one-fifth of the total image height. This prevents receiver noise from causing massive accidental error corrections and reduces the time required to preprocess a pair of images by a factor of five. One-fifth range cross-optimized processing uses the apriori knowledge that the cameras are nearly aligned. Refer to Figure 3.4 again for a diagram of this process.

The three optimal shifts resulting from the color cross optimizations are fed into



an optimized least squares (LSQ) solver. The LSQ solver is optimized for three data points using a hash table. The LSQ line is converted to an intercept and angle, used to shift and rotate pixel coordinates. The representation of the image in memory remains unaffected; only the way it is accessed changes. This procedure saves time that otherwise would be required to allocate and deallocate new large image data structures.

### **3.5 FEATURE CLUSTERING**

Each classified pixel is considered to be a feature. Feature correspondence is key for accurate range computations. An innovative technique for feature clustering used in this research is *histogram zooming*. Histogram zooming is performed on the right image frame only. Contrast feature point extraction must be performed on the image before this process can begin.

The process starts by dividing an image in grid fashion. Then a two-dimensional histogram is taken of the contrast feature points in each image bin. Bins having a number of features exceeding some threshold are considered for further resolution.

Mean geometric centers are calculated for each of the selected bins. Then a new rectangular *window*, having the same size as each histogram bin, is centered about this mean location. All contrast feature points located within the boundaries of this histogram window are used to zoom in on a particular object of interest. Objects of interest presumably contrast against the image background and are generally rectangular with one set of edges normal to the ground. This assumption makes it reasonable to further reduce

the size of each window using an *edge density* heuristic. Edge density is the ratio of the number of feature points within a small distance of an edge with respect to the length of that edge. Histogram window edge points are chosen for elimination if the edge they correspond to does not exceed some predetermined minimum threshold for edge density.

Windows are iteratively reduced in this fashion until the minimum edge density is met *collectively* by all edges or until the window size is below some predetermined minimum threshold. The latter case is disregarded as a false detection. Finally, all overlapping windows are unioned together, and a new window that bounds their union is used in their place. Each window now represents a potential object.

### **3.6 FEATURE GROUP CORRESPONDENCE**

The next step is to take each group of features created from histogram zooming and correspond them to the left image frame. Feature correspondence is the most difficult process in any stereovision system. This step also has the greatest potential for error, and thus it must be handled robustly. Frequently, simplicity is a trait associated with robustness. The *Occam's Razor* principle is applied here [33:14-15]. Section 3.4 explained the concept of color cross optimization. This technique is also used for corresponding feature groups, since each histogram window is easily converted into a horizontal color band. The *perspective constraint* says that a corresponding window in the left image must be further to the right from the window's location in the right image. Therefore, a horizontal color band is formed from the leftmost edge of the histogram

window extending to the right edge of the left image. These two color bands are color cross optimized to determine the disparity of the histogram window. If the minimum total error is too high, the window is ignored.

### 3.7 DEPTH MAP

Disparity information is used next for computing range information. Resolving a camera perspective of the world is equivalent to mapping from a small slice of a sphere onto a small rectangular region of a plane. Because light is radially focused onto a small rectangular charge-coupled device (CCD), each pixel of the CCD corresponds to a unique set of spherical angles. This mapping is visualized in Figure 3.8.

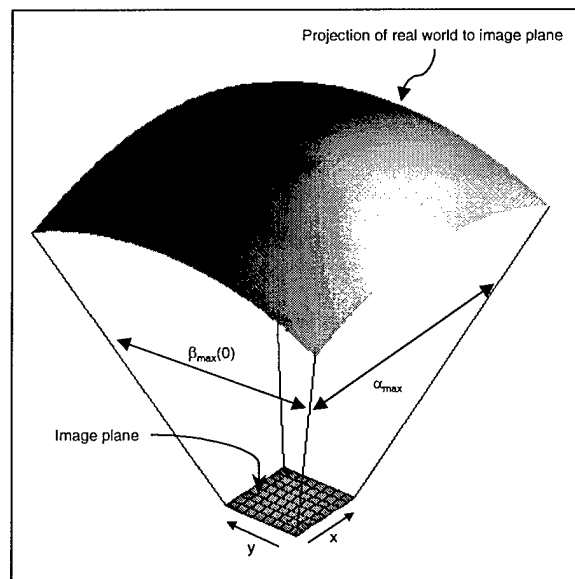


Figure 3.8 Depth map transformation

This research incorporates cameras aimed parallel to each other. In future MAV applications, this constraint may change to achieve resolvable disparities from cameras located closely together. In skew-aligned configurations the original depth map derived by this research would require simple angular corrections to  $\alpha$  and  $\beta$ .

Suppose that the center row of the CCD pixel sensor array constitutes the image plane. Let this row be a chord of a circle, spanning the angle  $\alpha_{max}$ . The circle drawn in Figure 3.9 has radius  $r$ , where  $h$  is the height of the chord above the center of the circle, and  $w$  is the width of the chord. There are  $n$  segments along the chord, each equal in length and representing one pixel. Define  $d$  to be the horizontal position of an illuminated pixel relative to the center column of the pixel array. Let  $k$  be the column index of the pixel, according to the convention defined in Figure 2.1. Then the following relationship applies:

$$v = k - \frac{n}{2}, \quad (3-2)$$

where  $v$  is the standard cartesian representation of  $k$ . Let  $\Delta r$  be the portion of the portion of  $w$  covered by one pixel, and let  $\alpha$  be the angle used to select the left edge of a pixel segment, measured counterclockwise from center. Then the following relationships apply:

$$\Delta r = \frac{w}{n} \quad (3-3)$$

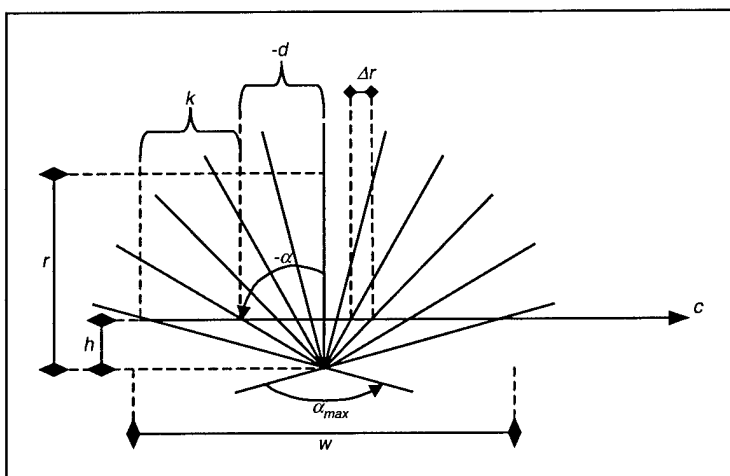


Figure 3.9 Center row anatomy

$$w = 2r \sin\left(\frac{\alpha_{\max}}{2}\right) \quad (3-4)$$

$$h = r \cos\left(\frac{\alpha_{\max}}{2}\right) \quad (3-5)$$

$$\alpha = \tan^{-1}\left(\frac{v\Delta r}{h}\right) \quad (3-6)$$

Using (3-2) through (3-6)  $\alpha$  may be obtained in terms of  $k$ ,  $n$ , and  $\alpha_{\max}$ :

$$\alpha = \tan^{-1}\left(\left(\frac{2k}{n} - 1\right) \tan\left(\frac{\alpha_{\max}}{2}\right)\right) \quad (3-7)$$

This relationship for  $\alpha$  is valid for any row in the image. Next, a vertical angle  $\beta$  is specified in terms of  $\alpha$ . Recall that  $\alpha_{\max}$  is the maximum horizontal viewing angle. It follows that  $\beta_{\max}(\alpha)$  is the maximum vertical viewing angle in some horizontal direction, measured  $\alpha$  radians counterclockwise from center. At this juncture the projection is best

conceptualized by placing a focal point behind the image plane/sensor array. A ray drawn from the focal point to the image plane is restricted such that it passes through the upper left-hand corner of a pixel in the sensor region of the plane. A top view shown in Figure 3.10, where  $l$  is the distance from the focal point to the image plane, and  $\Delta l(\alpha) + l$  is the distance from the focal point to a pixel in the center row at angle  $\alpha$ . Figure 3.11 shows how this model affects  $\beta_{max}(\alpha)$ . The height of the image plane  $h_\beta$  is directly related to  $\beta_{max}(\alpha)$ , the maximum vertical viewing angle. The following relationship describes  $\beta_{max}(\alpha)$ :

$$\beta_{max}(\alpha) = 2 \tan^{-1} \left( \frac{h_\beta}{2(l + \Delta l(\alpha))} \right) \quad (3-8)$$

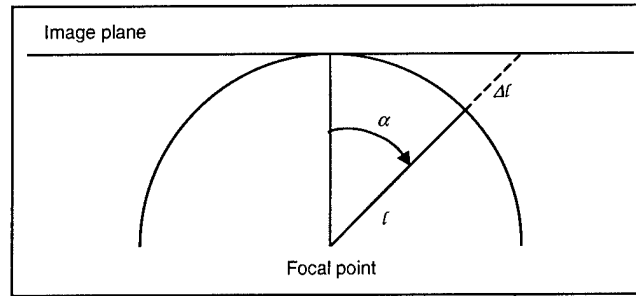


Figure 3.10 Top view of focal point projection

Figure 3.10 leads to the relationship:

$$l + \Delta l(\alpha) = \frac{l}{\cos(\alpha)}. \quad (3-9)$$

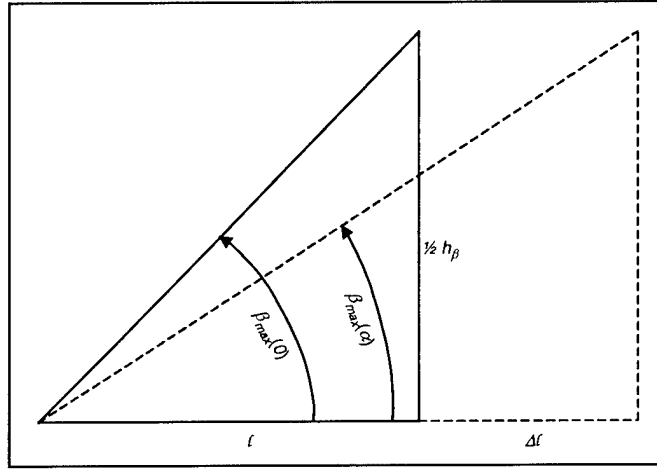


Figure 3.11 Cross-view diagram displaying vertical viewing angles

Figure 3.11 directly leads to

$$h_{\beta} = 2l \tan\left(\frac{\beta_{\max}(0)}{2}\right). \quad (3-10)$$

Substituting (3-9) and (3-10) into (3-8) yields

$$\beta_{\max}(\alpha) = 2 \tan^{-1}\left(\tan\left(\frac{\beta_{\max}(0)}{2}\right) \cos(\alpha)\right). \quad (3-11)$$

According to Figure 3.11, (3-9) may be used to express  $\beta(\alpha)$  as

$$\beta(\alpha) = \tan^{-1}\left(\frac{\left(\frac{m}{2} - j\right) \frac{h_{\beta}}{2}}{m \left(\frac{l}{\cos(\alpha)}\right)}\right) = \tan^{-1}\left(\left(\frac{1}{2} - \frac{j}{m}\right) \tan\left(\frac{\beta_{\max}(0)}{2}\right) \cos(\alpha)\right), \quad (3-12)$$

where  $j$  represents the row index and  $m$  is the total number of rows in the image plane.

The  $\alpha$  and  $\beta(\alpha)$  relationships are directly useable since the maximum horizontal and vertical viewing angles are specified for the cameras.

Let  $\Delta k$  be the disparity between two corresponding pixels in the left and right image planes, and let  $d$  represent the distance between a point in space and the center of the right image plane. Figure 3.12 shows this configuration with  $\alpha_1$  and  $\alpha_2$  representing the horizontal viewing angle of the point for the right and left images, respectively. Here,  $s$  is the distance of separation along the  $x$ -axis between the two image planes, and the right image plane is centered at the origin.

The goal is to calculate the distance in the  $x$ - $z$  plane,  $d_{xz}$ , from the right image plane origin to the point of interest using  $\alpha_1$  and  $\alpha_2$ . This goal is accomplished by determining the intersection of the two lines at the point of interest using the slopes and intercepts of the lines. The slopes of the right and left lines are

$$\begin{aligned} m_1 &= \cot(\alpha_1), \\ m_2 &= \cot(\alpha_2). \end{aligned} \tag{3-13}$$

The equations for lines 1 and 2 are

$$\begin{aligned} z_1 &= m_1 x_1, \\ z_2 &= m_2 (x_2 + s). \end{aligned} \tag{3-14}$$

Solving for the intersection yields

$$\begin{aligned} x &= \frac{m_2}{m_1 - m_2} s, \\ z &= \frac{m_1 m_2}{m_1 - m_2} s. \end{aligned} \tag{3-15}$$



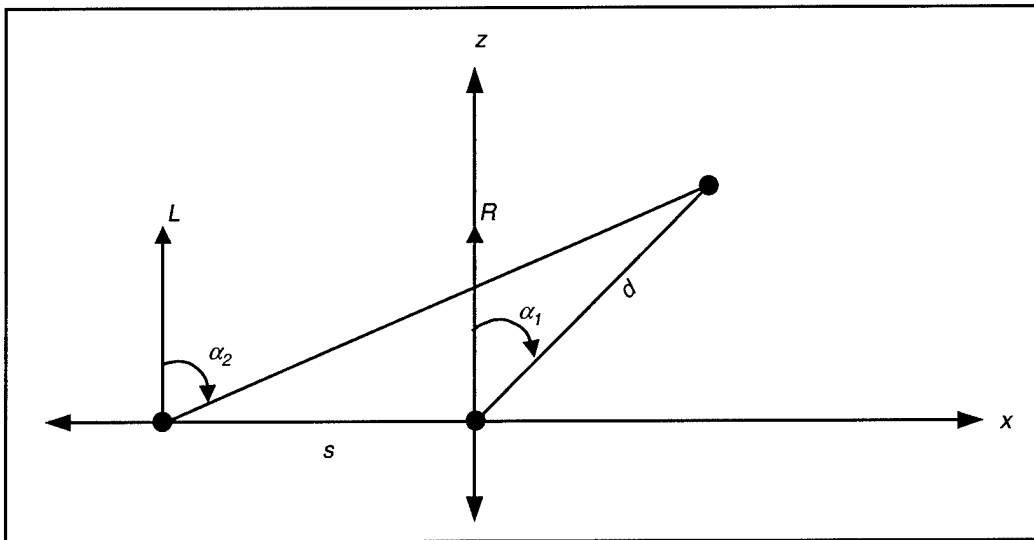


Figure 3.12 Dual image plane diagram

Solving for norm in terms of  $\alpha_1$  and  $\alpha_2$ , gives

$$d_{xz} = s \left( \frac{\cot(\alpha_2) \csc(\alpha_1)}{\cot(\alpha_1) - \cot(\alpha_2)} \right), \quad (3-16)$$

Next,  $d$  is calculated from this, using

$$d = \frac{d_{xz}}{\cos(\beta_1)} = \frac{s}{\cos(\beta_1)} \left( \frac{\cot(\alpha_2) \csc(\alpha_1)}{\cot(\alpha_1) - \cot(\alpha_2)} \right) \quad (3-17)$$

The real-world coordinates of any point are simply related to  $d$ ,  $\alpha$ , and  $\beta$  by

$$\begin{aligned} x &= d \sin(\alpha) \cos(\beta), \\ y &= d \sin(\beta), \\ z &= d \cos(\alpha) \cos(\beta). \end{aligned} \quad (3-18)$$

Combining (3-17) and (3-18) gives the final result:

$$\begin{aligned}
 x &= s \left( \frac{\cot(\alpha_2)}{\cot(\alpha_1) - \cot(\alpha_2)} \right), \\
 y &= s \tan(\beta_1) \left( \frac{\cot(\alpha_2) \csc(\alpha_1)}{\cot(\alpha_1) - \cot(\alpha_2)} \right), \\
 z &= s \left( \frac{\cot(\alpha_2) \cot(\alpha_1)}{\cot(\alpha_1) - \cot(\alpha_2)} \right).
 \end{aligned} \tag{3-19}$$

Equations (3-19) complete the depth map transformation derivation. Relative positioning data computed using this technique is used to help eliminate potential targets, which are markings on the road surface.

### 3.8 OBJECT RECOGNITION

Next to the correspondence problem, object recognition is the most difficult step to perform correctly. This research introduces a novel template-matching scheme for object recognition. This technique is called *multiscale nominal frequency-based recognition*. It uses a tree-based method to rapidly compare object signatures to templates from the object library. An object signature is created from the pixels inside the object-bounding rectangle. The signature is created using multiple scales of phase and frequency information. The technique operates under the assumption that an object is recognizable according to how frequencies are distributed throughout an image of that object. The process is charted in Figure 3.13. The idea is to apply successively finer short-space discrete Fourier transforms to the object image. Each increase in detail

introduces more phase information into the signature. Pruning the search space, beginning with coarse information, allows rapid convergence of the recognition process.

The process diagramed in Figure 3.13 begins by choosing an object to process. This object is classified according to its aspect ratio. There are five discrete aspect ratios:  $\frac{1}{4}$ ,  $\frac{1}{2}$ , 1, 2, 4. The object is distorted and scaled to fit the closest matching aspect ratio with the greater dimension measuring sixteen pixels. This forces the representative image to have horizontal and vertical dimensions that are powers of two. The *Fast Fourier Transform* (FFT) algorithm operates optimally on vector lengths which are powers of two.

The coarsest operation performs the minimum number of short-space FFT operations on square subimages, and for an aspect ratio of  $\frac{1}{2}$  this equates to two 8x8 operations (see Figure 3.14). Resulting phase information is discarded and coefficient magnitudes are saved. The magnitudes are used to determine the nominal spatial frequency. This number is saved for comparison. Equation 3-20 expresses the process of calculating the nominal frequency.

$$F = \frac{\sum_{j=-r/2+1}^{r/2} \sum_{k=-r/2+1}^{r/2} \sqrt{j^2 + k^2} |C_{jk}|}{\frac{r}{\sqrt{2}} \sum_{j=-r/2+1}^{r/2} \sum_{k=-r/2+1}^{r/2} |C_{jk}|} , \quad (3-20)$$

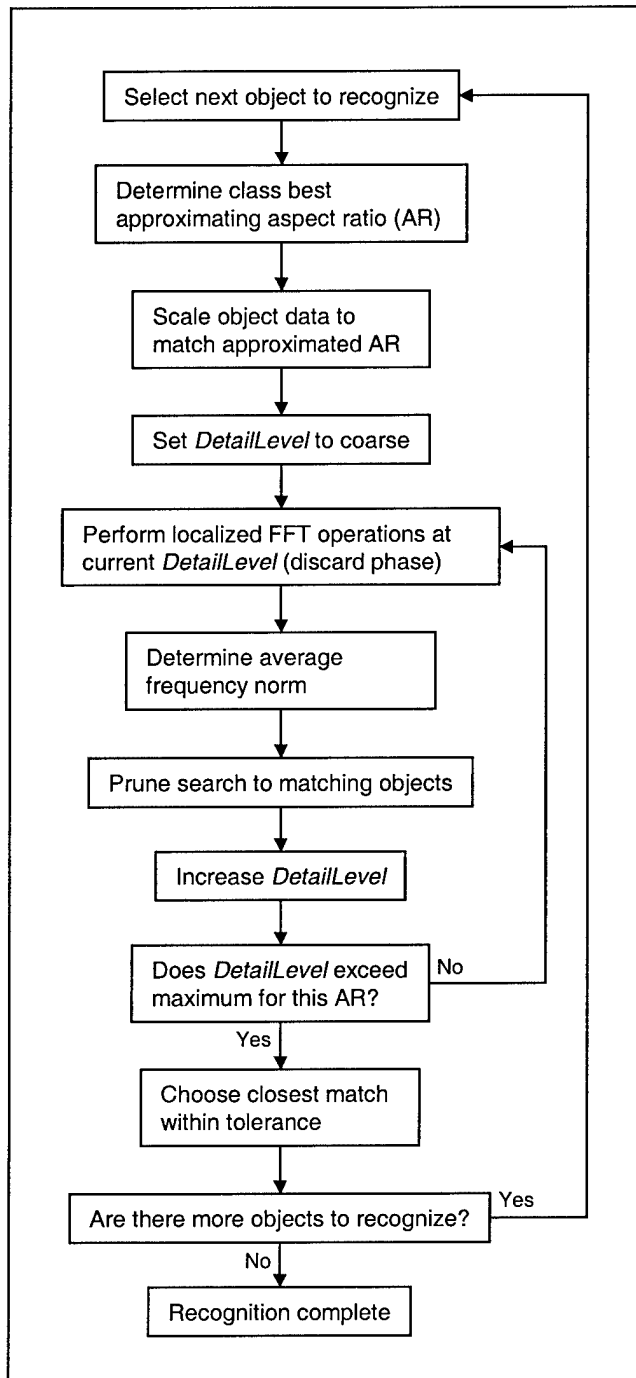


Figure 3.13 Multiscale nominal frequency-based recognition flow diagram

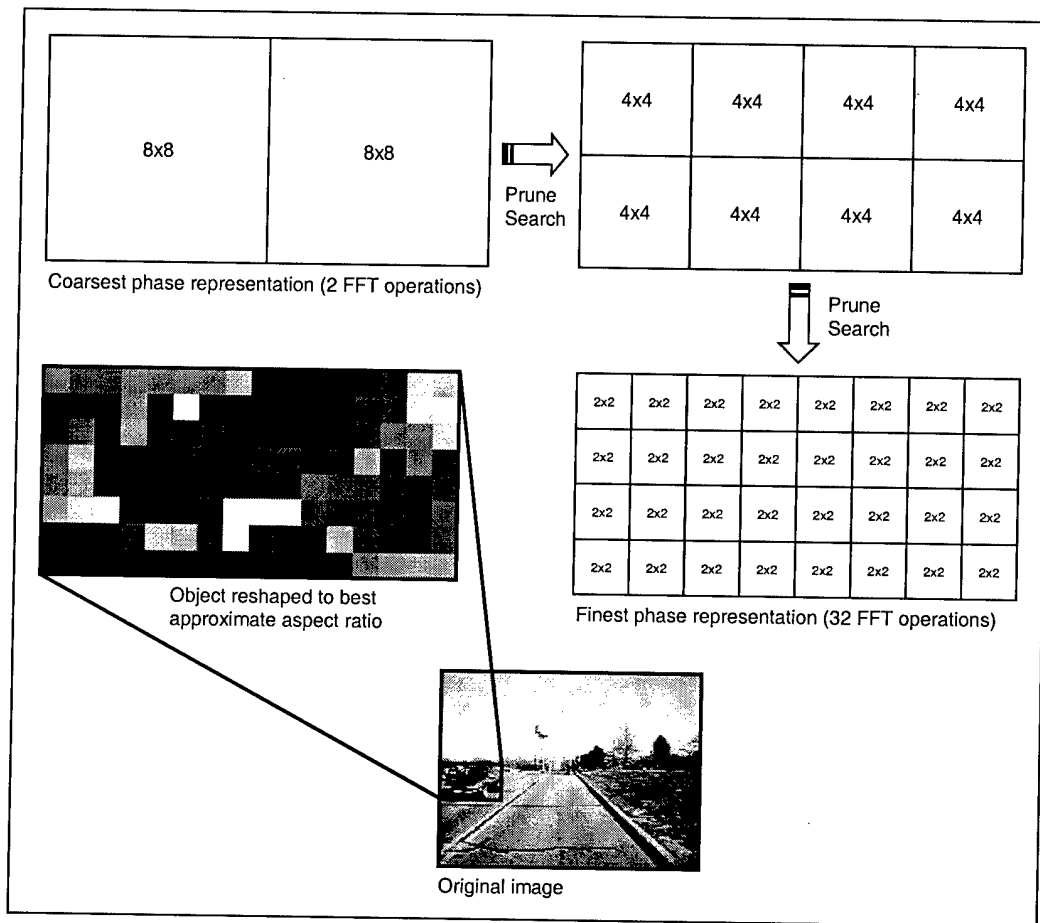


Figure 3.14 Multiscale nominal frequency-based recognition—detail pruning.

where

$r$  = Subimage dimensions (power of 2)

$\frac{r}{\sqrt{2}}$  = Maximum spatial frequency magnitude

$j$  = Vertical spatial frequency

$k$  = Horizontal spatial frequency

$\sqrt{j^2 + k^2}$  = Spatial frequency magnitude

$|C_{jk}|$  = FFT coefficients used as weights.

When the level of detail is increased the number of FFT operations grows by a factor of four, and each operation's dimensions are cut in half. The maximum level of detail occurs when FFT operations reach the lower bound of  $2 \times 2$  in size. After the object being classified has a complete signature, it is compared level-by-level to signatures of objects in the library. Each comparison prunes the search until a match is found unless the object cannot be identified.

### **3.9 CHAPTER SUMMARY**

This chapter provided low-level details about the theoretical contributions of this research. Key topics included preprocessing, feature clustering, feature group correspondence, depth map transformations, and the multiscale nominal frequency-based method of object recognition. Chapter 4 documents the results of these contributions.

## **4. RESULTS AND ANALYSIS**

### **4.1 CHAPTER OVERVIEW**

This chapter provides results, analysis, and future recommendations for each of the five vision processing functions presented in Chapter 3. Realtime testing was performed for the first four of these functions. Realtime software was implemented in a graphical user interface (GUI) using Microsoft® Visual C++™. The name coined for this software is *RavenVision*. All other results were generated using MATLAB®. All frame images processed are 160x120 pixels. Realtime processing ranged between 90 and 576 ms/frame, depending on the degree of background clutter in each frame. Nominal processing time was 262 ms/frame with preprocessing disabled. This time is less than desirable, but frame rates may be improved significantly by dual processing.

An important note is that time constraints did not permit a baseline comparison of the performance of *RavenVision* to other vision processing systems on identical data. Testing was performed on three different targets; two were stop signs, and one was a sport utility vehicle (SUV). All results are tied to one of these three targets.

### **4.2 PREPROCESSING**

Preprocessing consistently provided results accurate within  $\pm 1$  pixel with an equally consistent overhead time of 20 ms per frame pair. Unfortunately, variance information was unavailable for pixel errors; thus only the mode was determined.

Although the methods adopted do an effective job of determining the translational and rotational camera misalignments, they also require additional processor time whenever the altered image is accessed. The reason is that the actual image is never altered. Instead, coordinates are transformed when needed. Some image operations require access to the original untransformed image. This procedure preserves the entire image without clipping due to rotation, allowing all image operations to function properly. Finally, preprocessing does not account for rotational errors about the y-axis—the most critical source of errors for ranging data.

#### ***4.2.1 FUTURE RECOMMENDATIONS***

A preferred solution to preprocessing or calibration is manufacturer precalibration. In the MAV context, significant translational epipolar camera separation is not allowed, making depth mapping more prone to errors. This type of system would be more like that of flying insects, such as the dragon fly. In such a compact system, image resolution is higher to compensate for small interimage disparities. Figure 4.1 displays an original computer rendered solution. This precalibrated solution is more compact and provides more accurate results, eliminating the need for preprocessing. The tradeoff is that the higher resolution images take longer to process.

#### ***4.3 FEATURE CLUSTERING***

One prerequisite to getting consistent results from any feature clustering algorithm



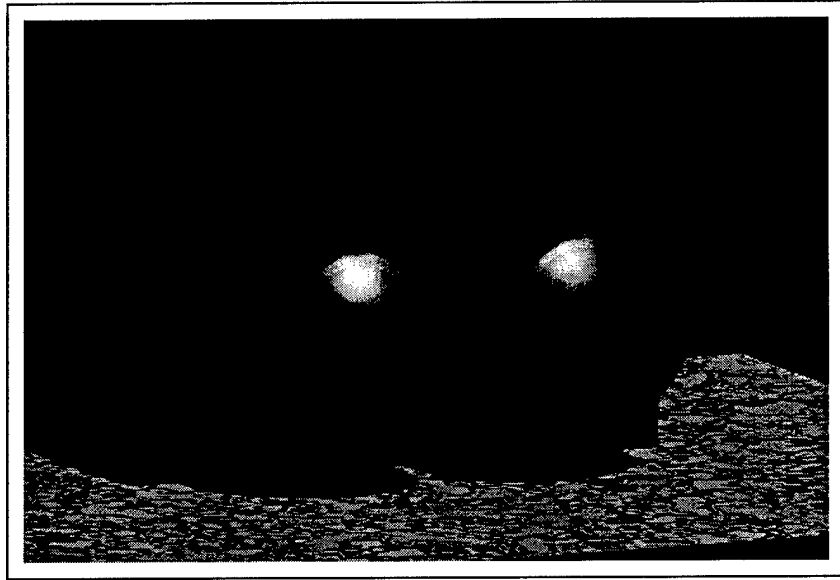


Figure 4.1 Futuristic rendition of miniature precalibrated stereo camera rig.

is to have input from a robust feature extraction algorithm. This research examined four variations of contrast point feature extraction algorithms.

#### ***4.3.1 FEATURE EXTRACTION***

All of the feature extraction methods are shown in Figure 4.2. The original image is the leftmost image in Figure 4.2. The target in this image is the barely visible stop sign indicated by a bounding rectangle. The idea is to eliminate as much background clutter as possible while picking out a large number of points from the desired target. The first method is the one used in the Stop&Go system [3]. This full-contrast detection system is implemented in Stop&Go as a horizontal contrast classifier. Therefore it has large redundancy in the points selected by the feature classification scheme. In an effort to

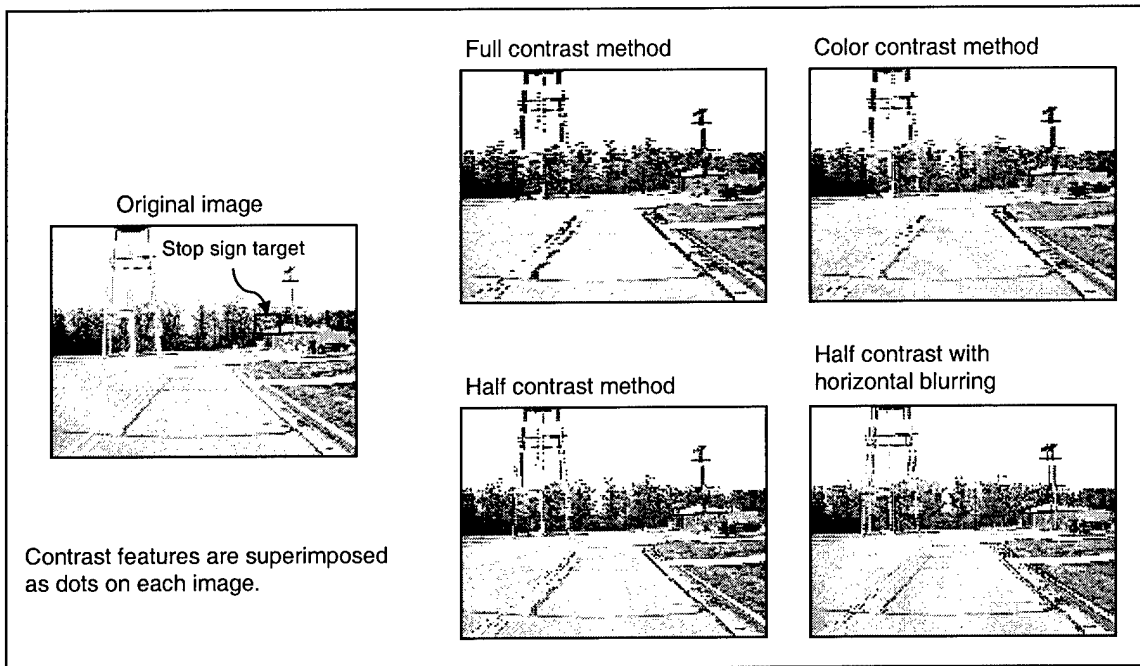


Figure 4.2 Contrast point feature extraction schemes.

create a compatible baseline scheme, a color-based contrast classifier was developed to be completely compatible with the Stop&Go system. This scheme is shown in the upper right image of Figure 4.2. The color contrast method is detailed in Appendix A. Notice how all features occur in groups of two or more for each of these first two methods. RavenVision needs a scheme to extract fewer points. This need led to the half contrast method, which yields only half as many feature points. Notice that there is still a significant amount of clutter in the background caused by trees near the stop sign. By blurring the image horizontally, fewer background points are extracted, and at least the pole supporting the stop sign is illuminated with features. This result is the best that can

be expected for a contrast feature extraction scheme operating under such poor conditions.

### **4.3.2 HISTOGRAM ZOOMING**

The results of histogram zooming on feature sets, taken from horizontally-blurred half contrast point feature extraction, are surprisingly good. Figure 4.3 shows a typical right image frame for the SUV target. The left image in the figure shows contrast point features superimposed on top of the right image frame. The right image in the figure shows the succession of histogram zooming windows. The innermost window accurately bounds the target of interest.

Figure 4.4 shows complete results from this particular image frame. Notice how the telephone pole, part of the building, and part of the road surface are also selected as potential targets. The road surface point is eliminated later by determining its *y*-position relative to the car. However, the other two objects are kept for further analysis. At this time, no method has been developed for eliminating false targets that are not part of the road surface.

Another problem that occurs when a target is very close to the RAVEN is *blocking*. Blocking effects occur when a target is broken into several smaller targets and each is treated individually. Figure 4.5 illustrates this problem for the SUV target.

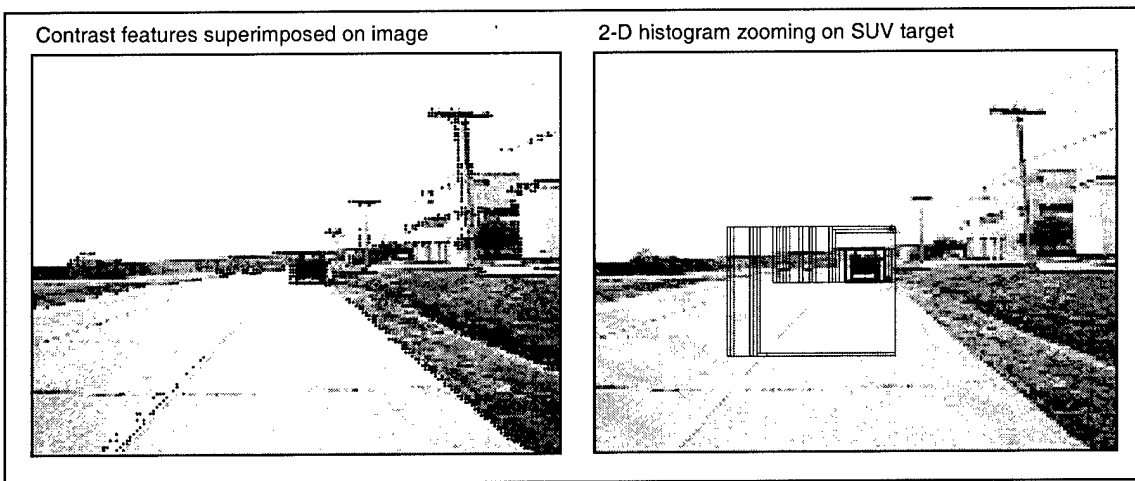


Figure 4.3 Histogram zooming results overlaid on right SUV target image frame.

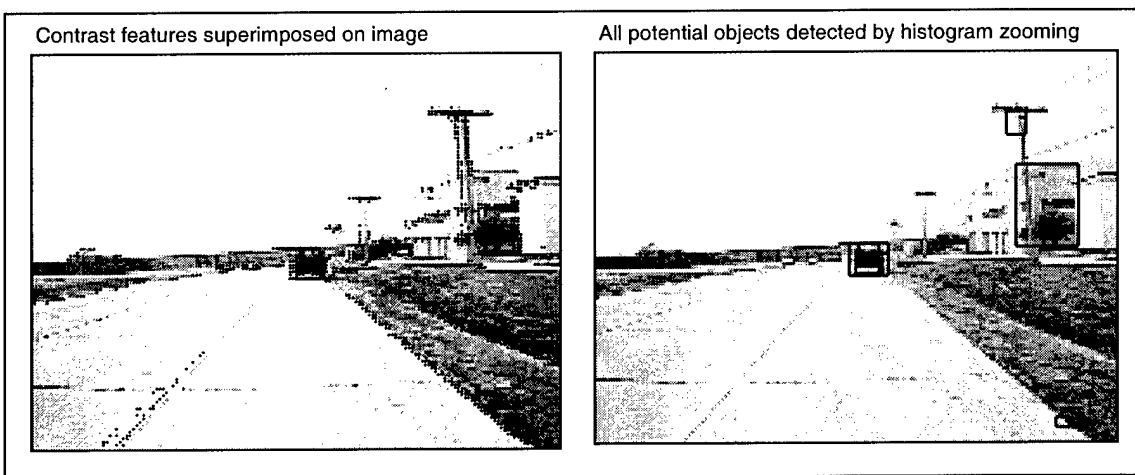


Figure 4.4 Complete results of histogram zooming for image frame of Figure 4.3.

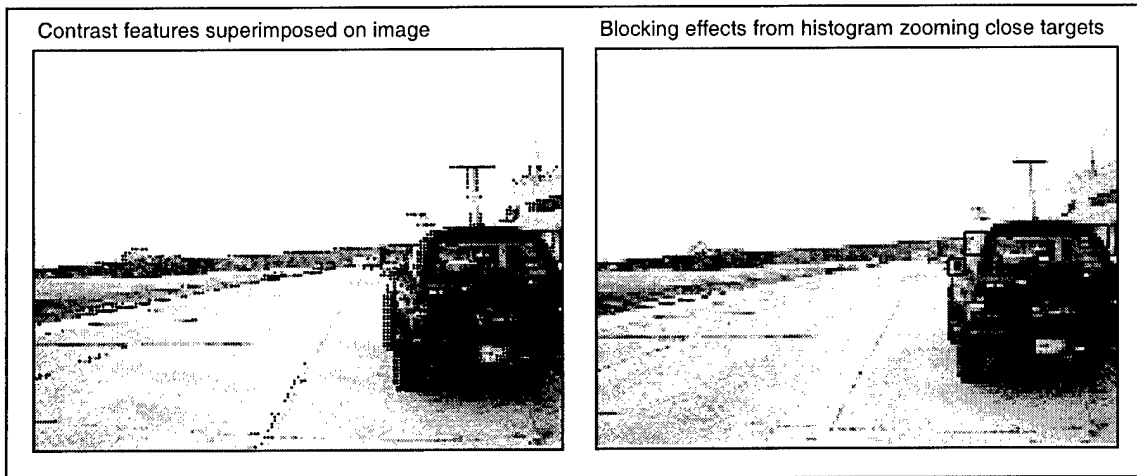


Figure 4.5 Blocking effects on close range targets.

### 4.3.3 FUTURE RECOMMENDATIONS

The blocking effect could be partially eliminated by fusing windows that have approximately the same location. This fusing would properly identify the SUV (excluding tires) given the image frame in Figure 4.5. The drawback is that two targets located near each other are treated as one. A laser ranger may be useful in this case. After detecting targets visually and approximating their locations, a laser ranging device may be used to fine-tune results.

Another possible improvement is to perform line detection on the portion of the image frame represented by each histogram cluster before zooming, which would eliminate more ground clusters. Additionally, after blocking effects are removed through fusing cluster windows, the line detection step may be iterated once more to verify that the fused targets are really one target and not multiple proximately located targets. Line

detection should not be performed on the entire image frame because of its computational expense.

A final recommendation is the incorporation of GPS information and previous target history to predict where in the image to look for a previously detected target. This incorporation may improve confidence in target detection.

#### **4.4 FEATURE GROUP CORRESPONDENCE**

Once a target is properly selected in the right image frame, cross color optimization does an accurate job of feature group correspondence. Figure 4.6 shows one typical result of this process. The error in this case is one pixel. Notice how the inaccurate window size has little effect on the robust correspondence process.

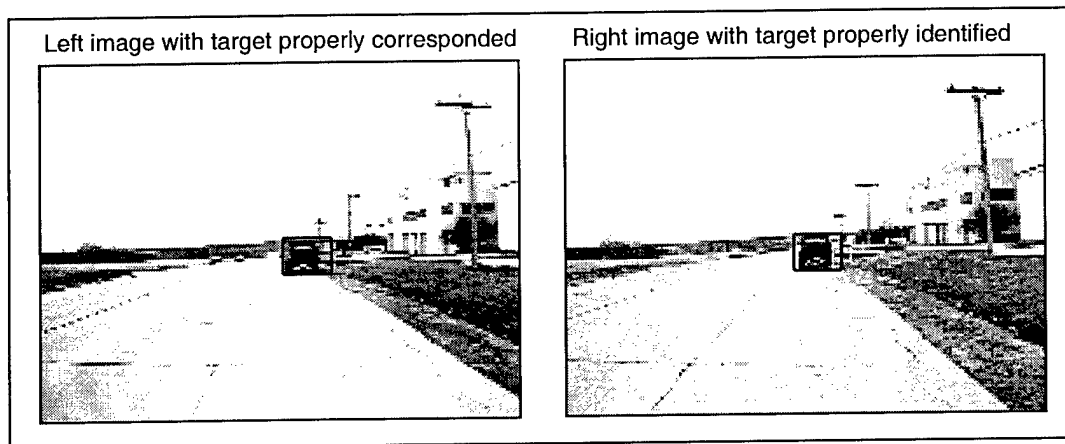


Figure 4.6 Proper feature group correspondence on SUV target with one pixel error.

A target is *detected* when it is identified by histogram zooming and then properly corresponded. To characterize the quality of target detection *confidence* plots were made. Confidence is defined as the portion of frames in some symmetric interval surrounding the current frame which have both the target of interest properly selected in the right image frame and the target window correctly corresponded to the left image frame. Blocking is considered an acceptable form of target detection if the individual blocks are corresponded properly. All plots show an eleven-point confidence interval—five frames on either side of the frame of interest.

Figure 4.7 shows confidence in detection for the first stop sign target. This result is quite poor. A sample image frame for this poor-quality target is shown in Figure 4.2. Figure 4.8 shows the confidence for the SUV target, and Figure 4.9 characterizes the second stop sign target. Notice the trend in each of these plots. At very far range, the confidence in detection is low, then confidence peaks somewhere between 15 and 25 units of normalized distance from the target. Units of distance are normalized to the geometric mean of the target dimensions. Confidence in detection bottoms out somewhere between 3 and 7 units due to severe blocking effects. In the given plotting convention, distance decreases to the right as a function of increasing time.

#### **4.4.1 FUTURE RECOMMENDATIONS**

Feature group correspondence does not work properly in all cases, as inferred from Figures 4.7-4.9. When the process fails it is usually due to varying backgrounds

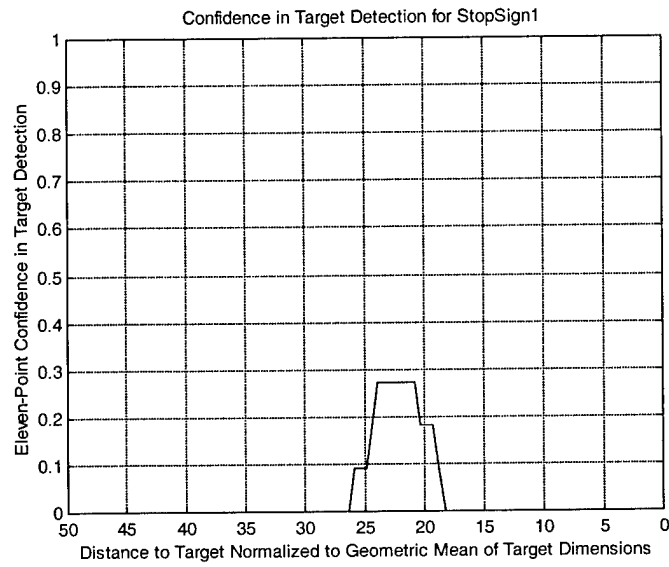


Figure 4.7 Confidence in target detection for the first stop sign target.

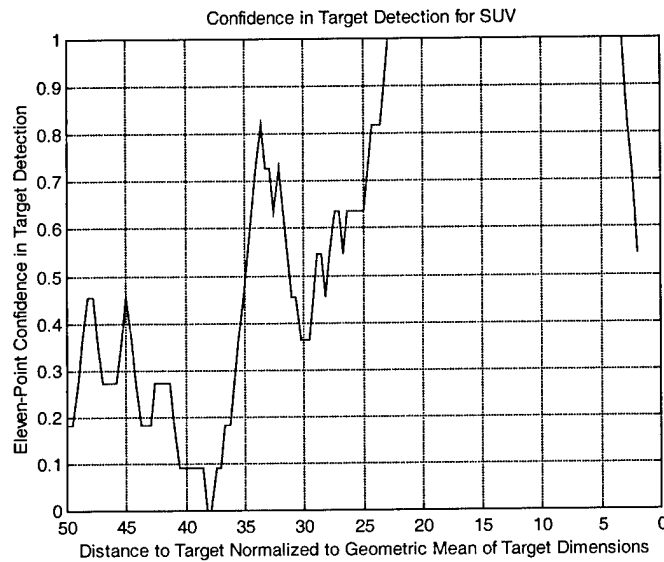


Figure 4.8 Confidence in detection for the SUV target.



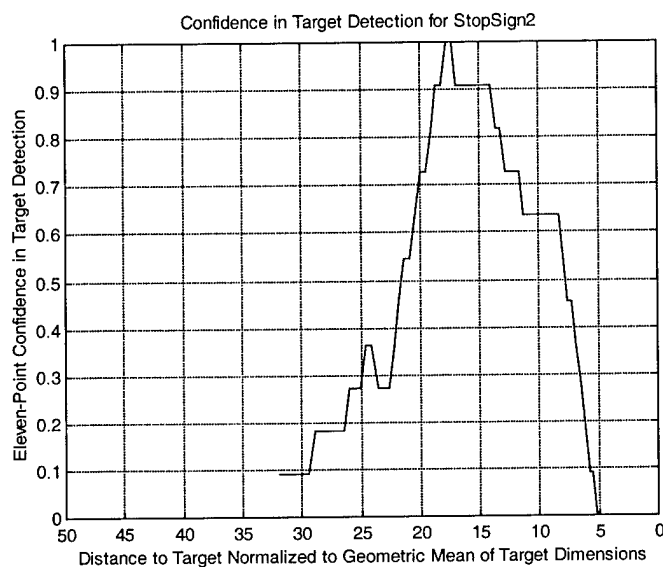


Figure 4.9 Confidence in target detection for the second stop sign target.

between perspectives or because of unmatched color characteristics between charge coupled device (CCD) sensors. One way to eliminate both of these causes is to perform line detection on the portion of the image represented by the feature group, polygonize the lines into endpoint sets, and then correspond matching line patterns with the left image frame. In the left image frame line detection should be performed on a strip that would otherwise be used in color cross optimization. Making the strip 10% taller might be a favorable way to reduce errors in this new process. For details on line detection and polygonization, refer to [8, 9, 10, 11]. The recommended process would also eliminate the need to acquire left image frames in color, offering a significant speed advantage. For further reading on alternate object detection schemes, refer to [12, 13, 14]. For

information on interframe object tracking, see [21, 22, 23, 24, 25, 26, 27, 28].

#### 4.5 DEPTH MAPPING

A problem was discovered with using the direct depth mapping model derived in Chapter 3: the cameras being used have optical distortion. The most prominent is the fisheye effect depicted in Figure 4.10. Another contributor to this problem is the error in manufacturer specifications on field of view ratings for the cameras.

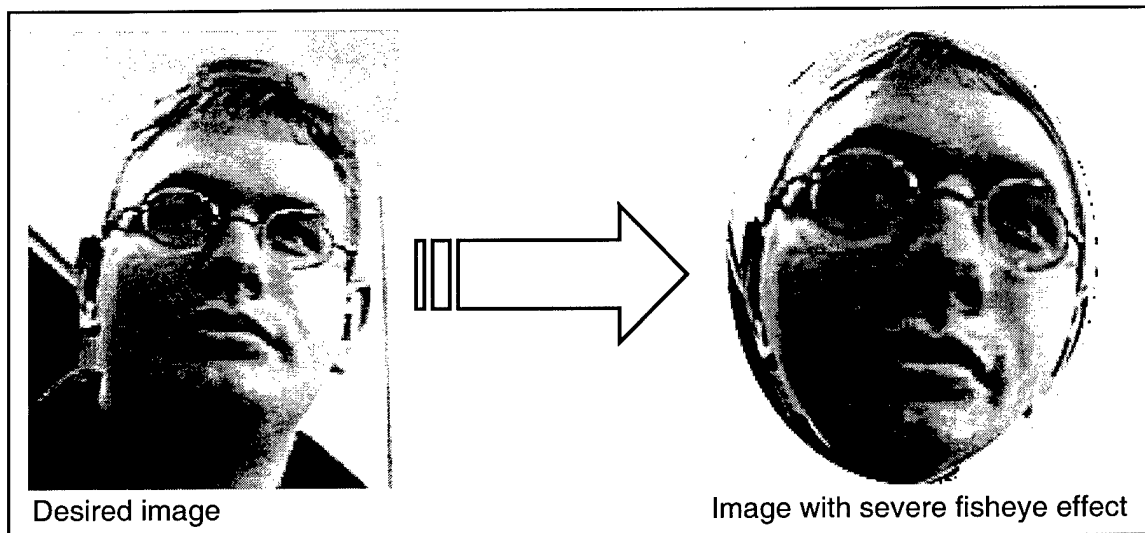


Figure 4.10 Fisheye effect.

Camera alignment was calibrated to test the significance of these errors once the problem was discovered. Cross-hairs were precisely taped in black on a white wall directly in front of the RAVEN in the laboratory with an accuracy of 13 mm—subpixel

accuracy at a distance of 3.327 m from each corresponding camera. These cross hairs were placed normal to each CCD surface. Figure 4.11 shows the calibration setup.

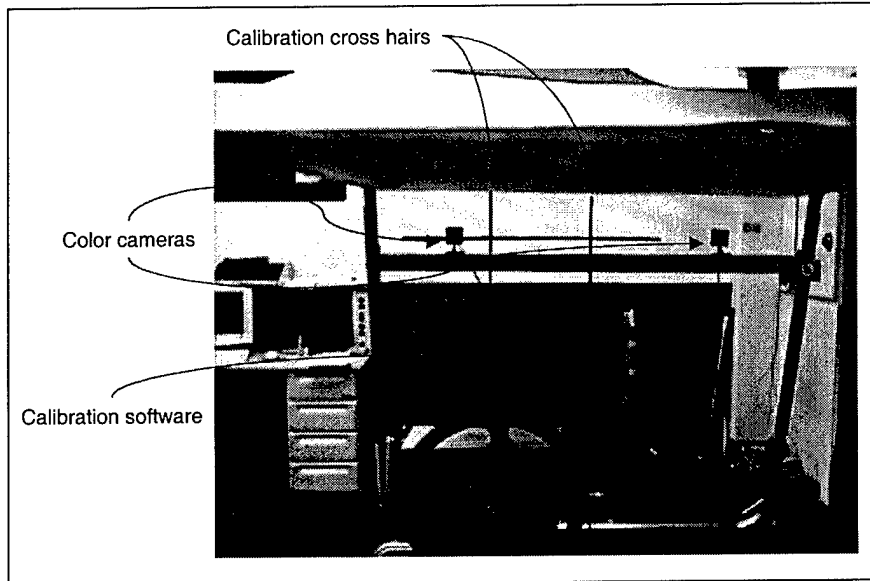


Figure 4.11 Cross-hair camera alignment calibration setup.

To calibrate camera alignment, RavenVision was placed in calibration mode, where stereocapture is performed with red cross hairs drawn over the center of the image. The cameras are oriented about their ball joints until the red and black cross hairs line up, and then they are fastened into place. This system is subject to vibrations during driving, which may not occur in the same severity onboard an MAV.

Image frame pairs were captured for 60 different target locations—3 different horizontal positions, 4 different depths, and 5 different heights. These 60 test images

were reduced to single point contrast images and fed into the RavenVision depth map processing algorithm. Figure 4.12 shows results for the  $\alpha$  calibration and Figure 4.13 shows results for the  $\beta$  calibration. A first-order least squares fit was made to the data in each case. This approximation is not only more accurate but also faster to implement than the true depth mapping model. Once the approximation parameters were calculated, these new models were used to compute the effective manufacturer specifications for  $\alpha_{\max}$  and  $\beta_{\max}$ , which yield the least squares error between the true model and the estimated model. The results are in (4-1), below. The fisheye effect is most pronounced in the vertical direction.

$$\begin{aligned} \alpha_{\max} &= 57.0^\circ & \alpha_{\max(\text{effective})} &= 55.023^\circ \\ \beta_{\max}(0) &= 41.0^\circ & \beta_{\max(\text{effective})}(0) &= 73.610^\circ \end{aligned} \quad (4-1)$$

Given this new model for computing  $\alpha$  and  $\beta$  for image frames, the near range accuracies were much improved. Table 4-1 gives the errors for the 60 test images before and after corrective modeling. These errors are relative to the straight-line distance of each test point. Notice the significant change in the  $z$ -error, shown in Figure 4.14. The horizontal viewing angles of the target within the right and left image frames are  $\alpha_1$  and  $\alpha_2$ , respectively. The reason for the large change in the  $z$ -error is that target points are generally far away from the RAVEN with respect to the translational disparity between left and right image frames.

Uncompensated close range errors are shown in Figures 4.15 – 4.17, and

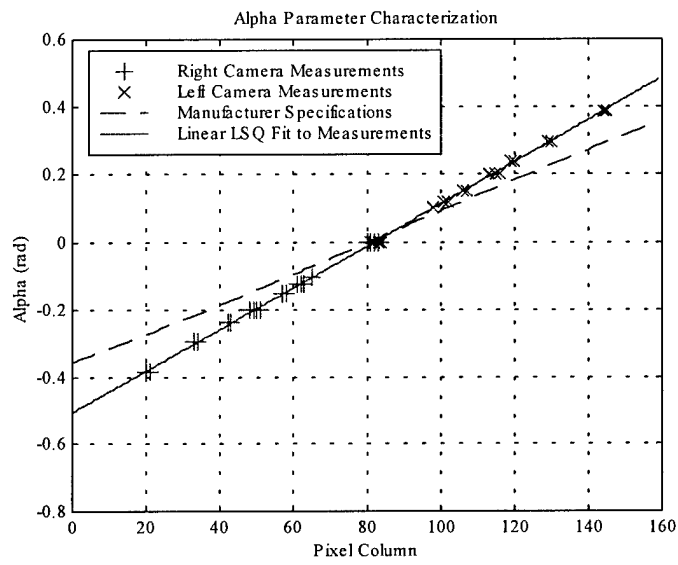


Figure 4.12 Corrected  $\alpha$  parameter model.

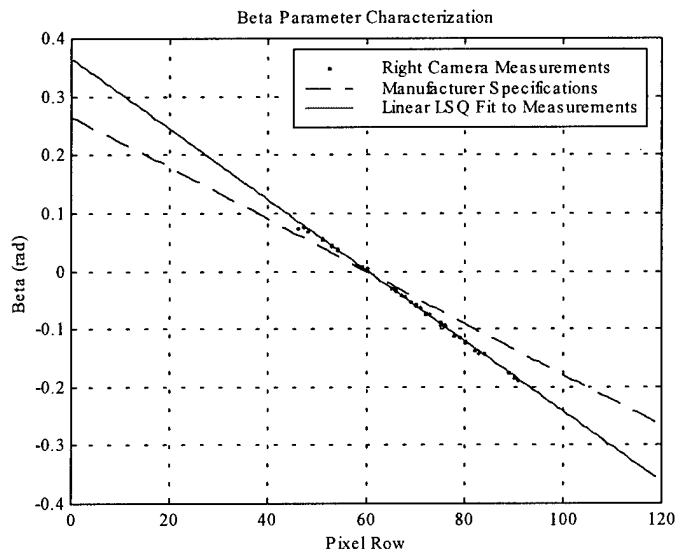


Figure 4.13 Corrected  $\beta$  parameter model.

	<i>Before</i>	<i>After</i>
<b>X</b>	2.96%	1.91%
<b>Y</b>	1.50%	1.12%
<b>Z</b>	42.55%	2.42%

Table 4.1 Relative errors in measurements before and after corrective modeling.

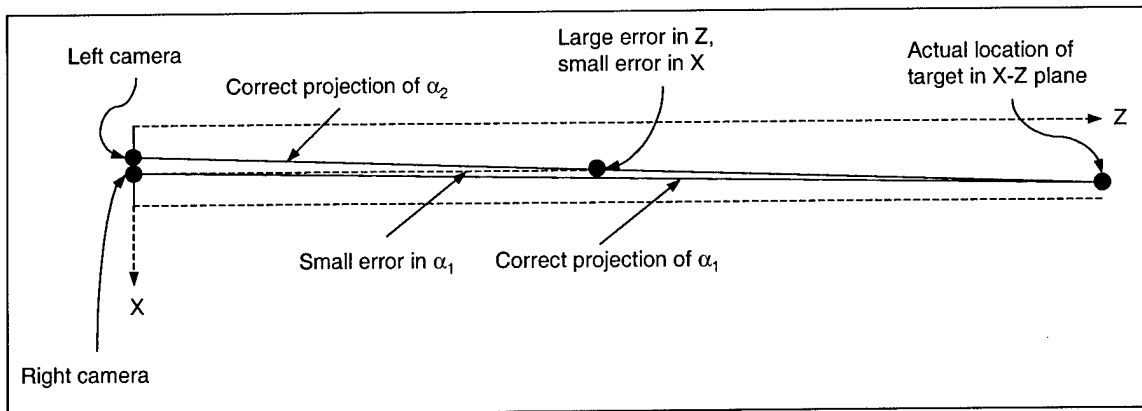


Figure 4.14 Small angle perspective model applied to range errors.

Figures 4.18 – 4.20 show close range errors after applying the corrective  $\alpha$  and  $\beta$  models. Note that the points with  $\alpha = 0$  tend to have greater error. This result occurs because the small angle perspective model has an increased effect for small angles.

Once calibrated for close range measurements, the RAVEN was tested for long-range relative position measurements with differential GPS (DGPS) as a baseline for comparison. Due to lack of confidence in target detection for the first stop sign target, it is eliminated from this discussion. All future references to the stop sign target refer to

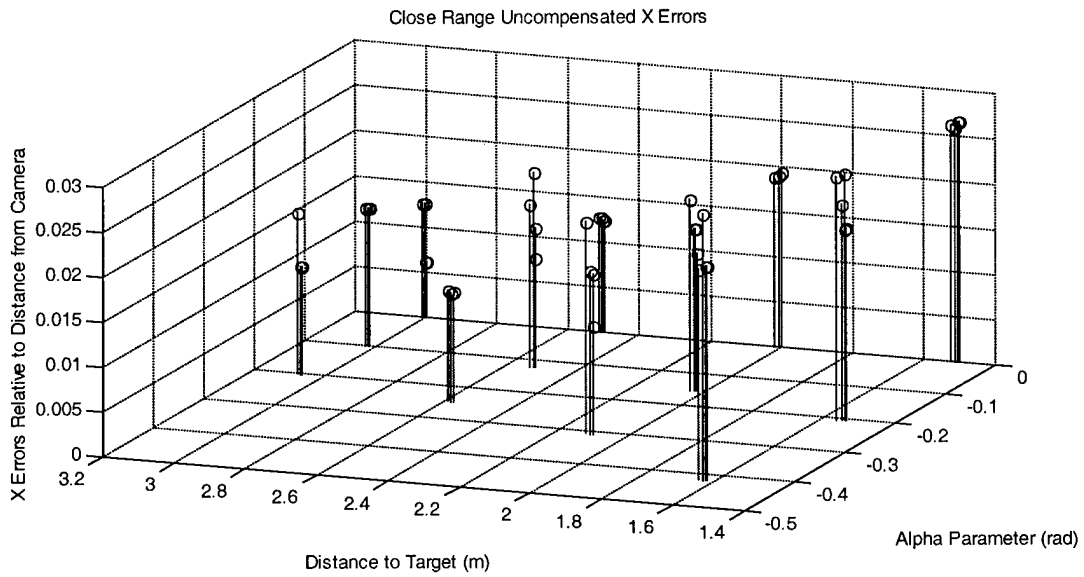


Figure 4.15 True model errors in  $x$ -direction relative to straight line distance to target.

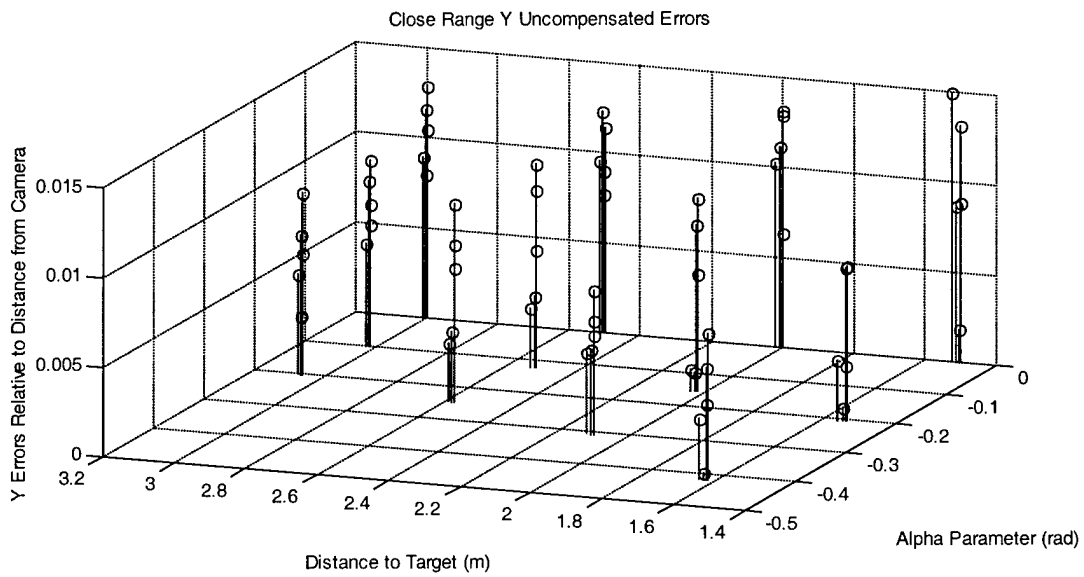


Figure 4.16 True model errors in  $y$ -direction relative to straight line distance to target.

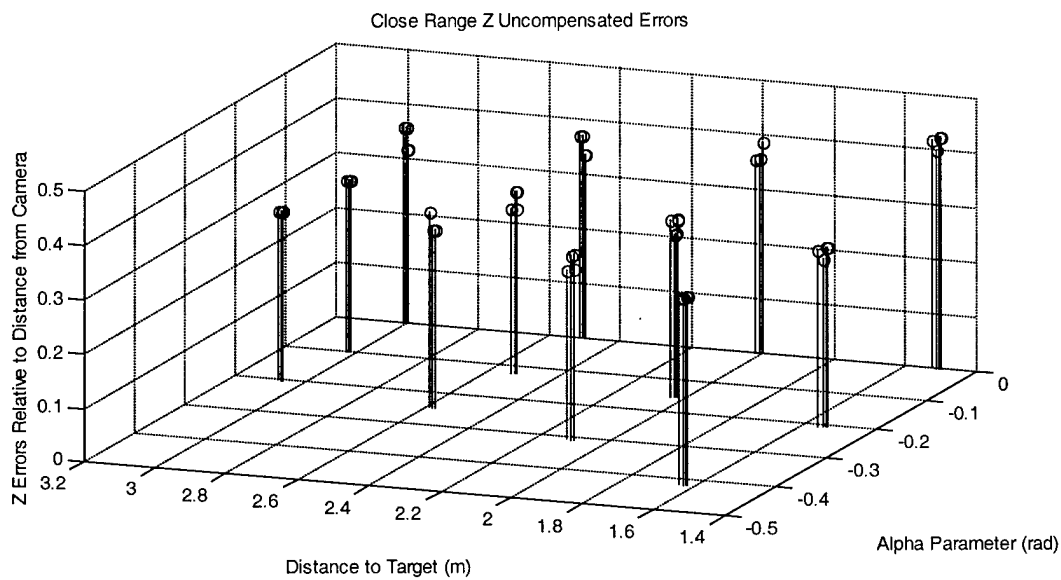


Figure 4.17 True model errors in z-direction relative to straight line distance to target.

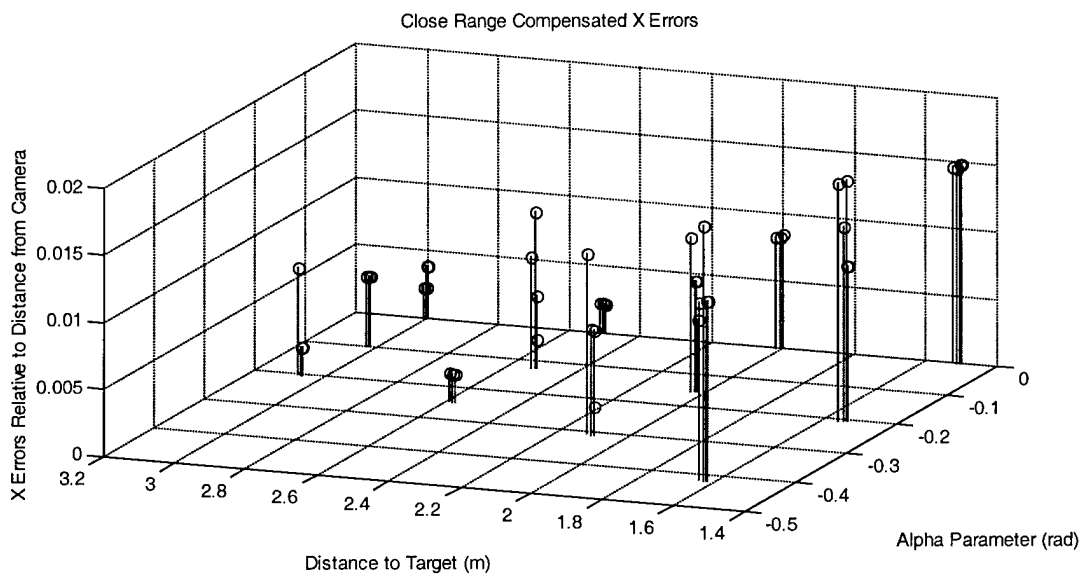


Figure 4.18 Relative corrective model errors in x-direction.



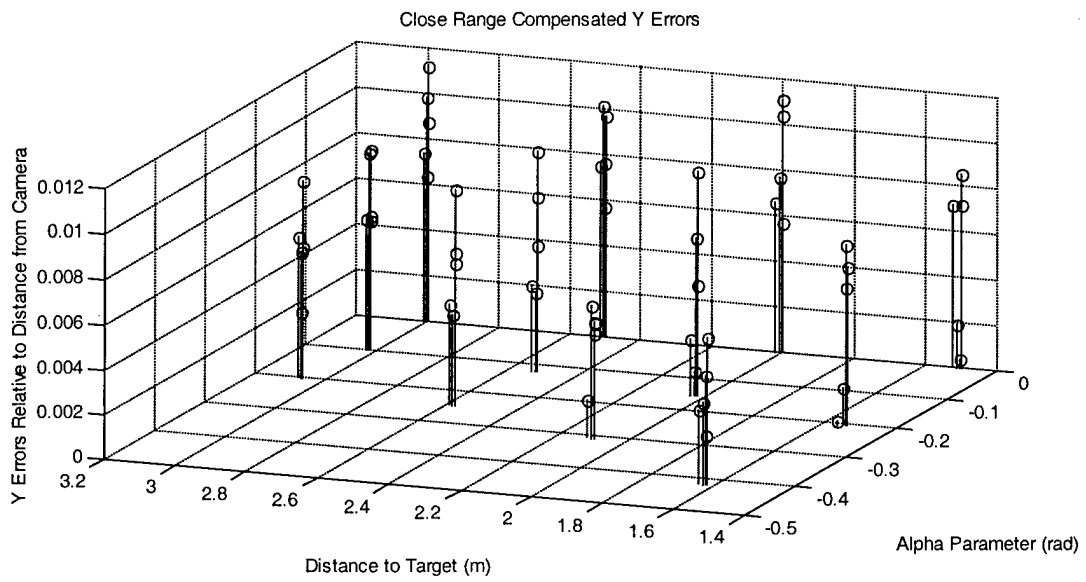


Figure 4.19 Relative corrective model errors in y-direction.

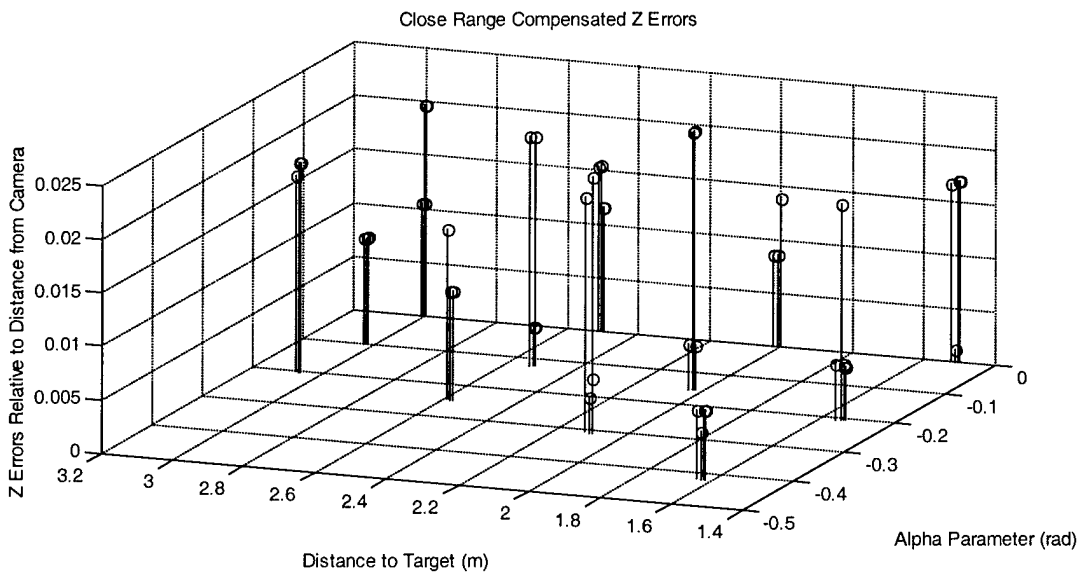


Figure 4.20 Relative corrective model errors in z-direction.

the second stop sign target. DGPS positional RMS errors for the SUV and stop sign target are 2.25 cm and 4.2 cm, respectively. Figures 4.21 – 4.25 show positional and dimensional measurement errors for the SUV target with respect to normalized target distance. Target detection confidence is superimposed over each plot. Figures 4.26 – 4.30 correspond to the same results for the stop sign target. Notice the general trend that target errors decrease where confidence in detection is high. This trend makes intuitive sense at a physical level despite the fact that target detection errors and measurement errors are algorithmically independent events.

#### ***4.5.1 FUTURE RECOMMENDATIONS***

Depth mapping may be improved across multiple image frames by incorporating DGPS information. If a target is recognized and has a tracking history, the vision model is assisted in estimating the new target location. Using this technique definitely makes the correlation between confidence in detection and positional error more pronounced. Dimensional errors are especially high compared to positional errors. This effect is attributed to target window fusing. The problem is rooted in histogram zooming, and may be improved by simply eliminating all overlapping target windows except the one with the largest area. The result is tighter target windows, leading to better dimensional measurements. Another recommendation is to investigate structure-from-motion techniques to determine target orientation and depth.

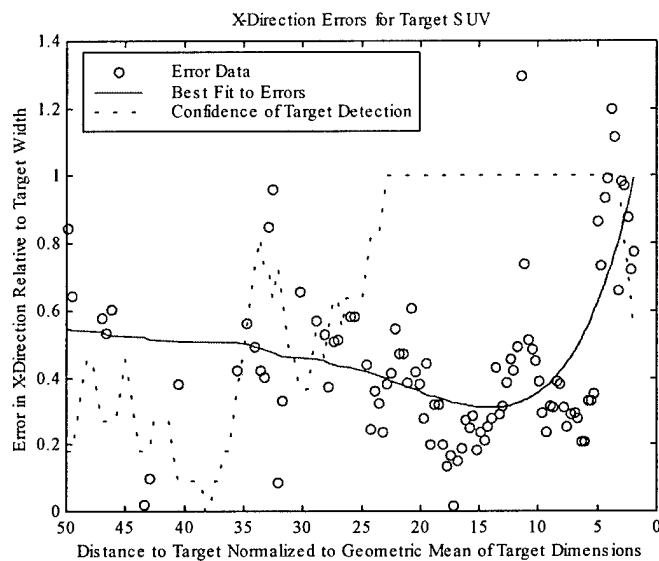


Figure 4.21 Relative  $x$ -direction errors for the SUV target.

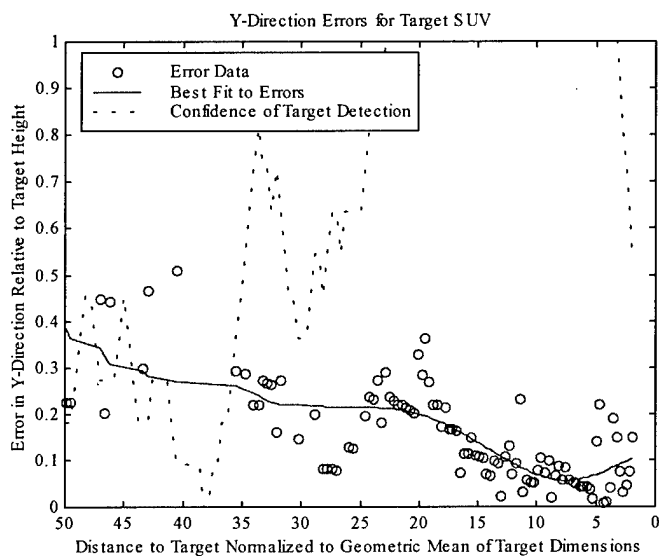


Figure 4.22 Relative  $y$ -direction errors for the SUV target.

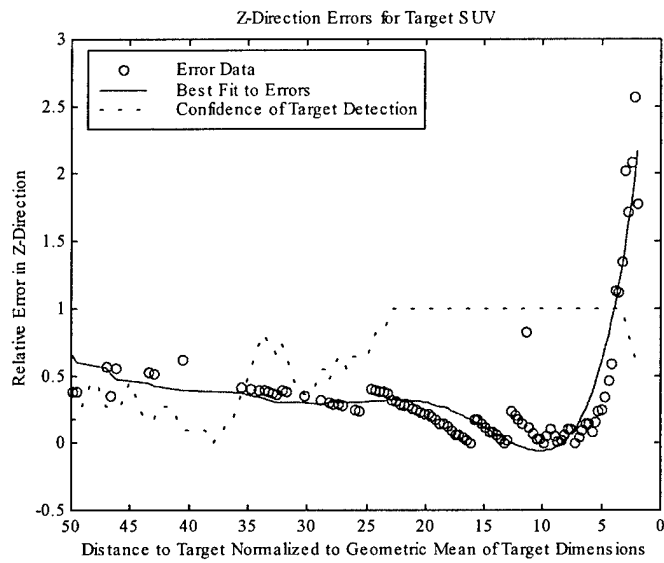


Figure 4.23 Relative z-direction errors for the SUV target.

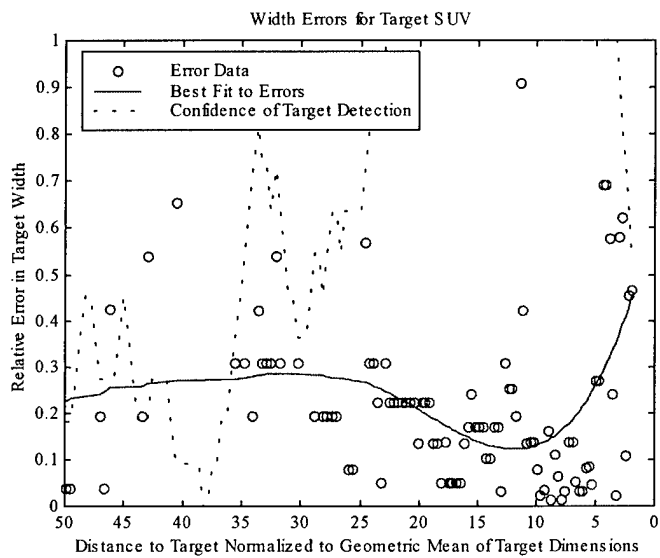


Figure 4.24 Relative width measurement errors for the SUV target.

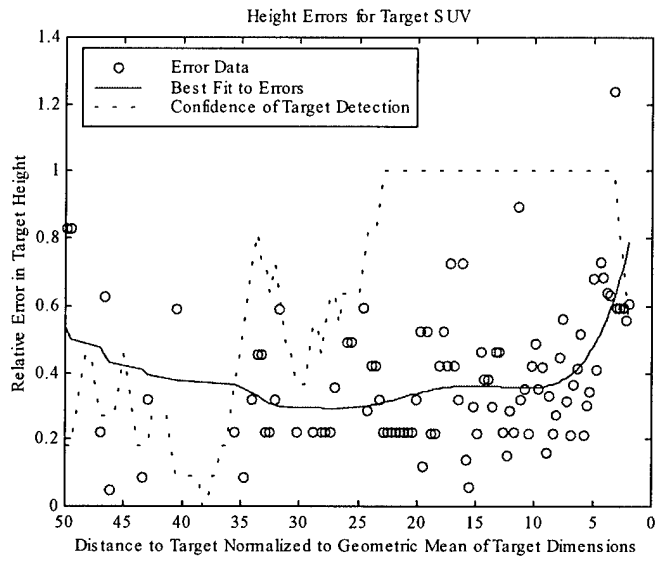


Figure 4.25 Relative height measurement errors for the SUV target.

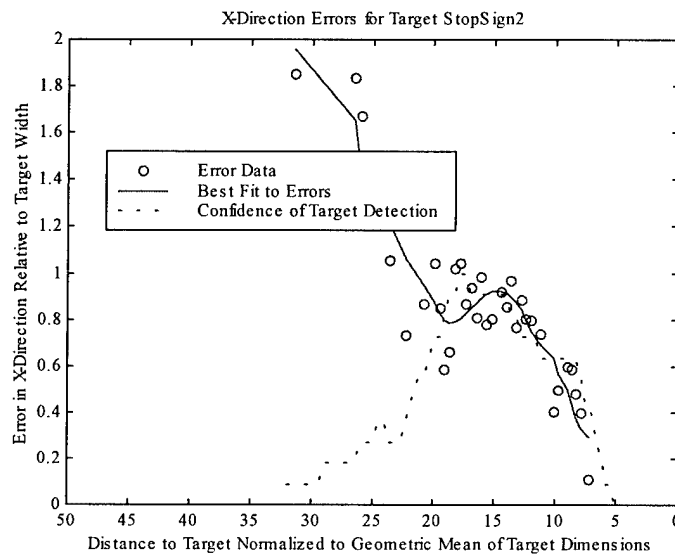


Figure 4.26 Relative x-direction errors for the stop sign target.

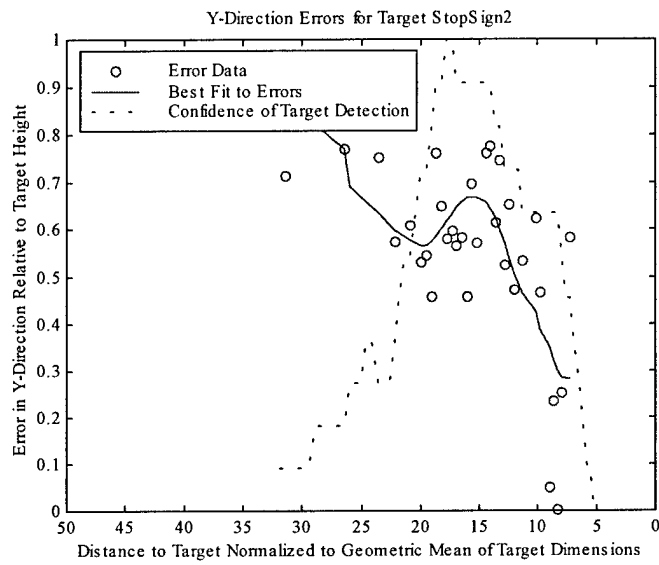


Figure 4.27 Relative y-direction errors for the stop sign target.

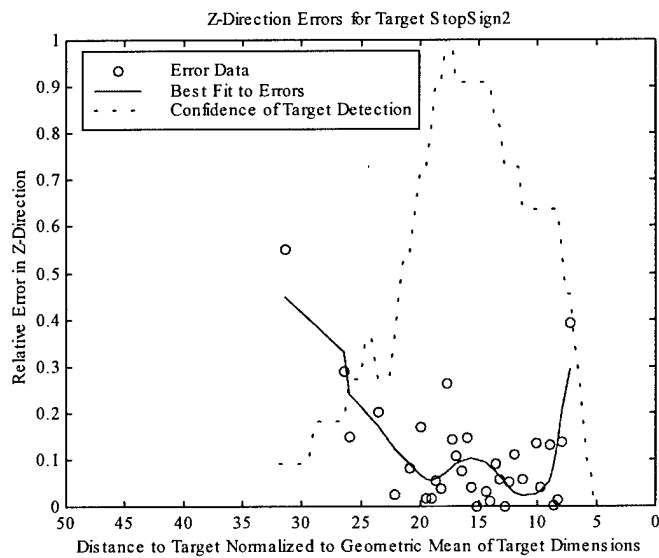


Figure 4.28 Relative z-direction errors for the stop sign target.

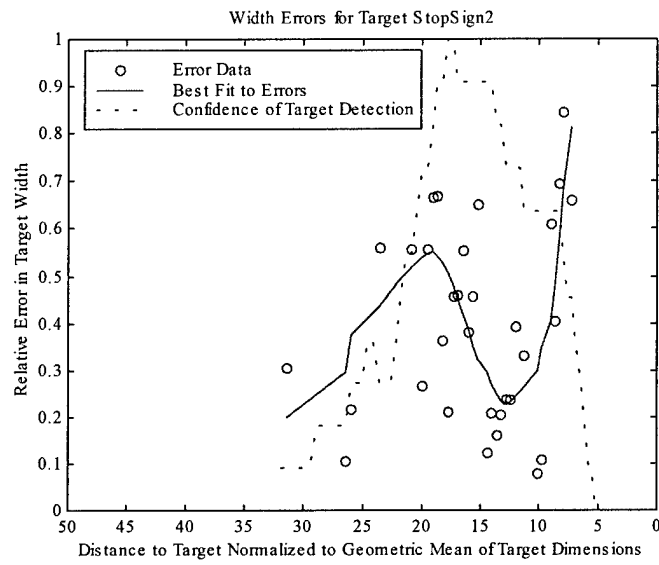


Figure 4.29 Relative width measurement errors for the stop sign target.

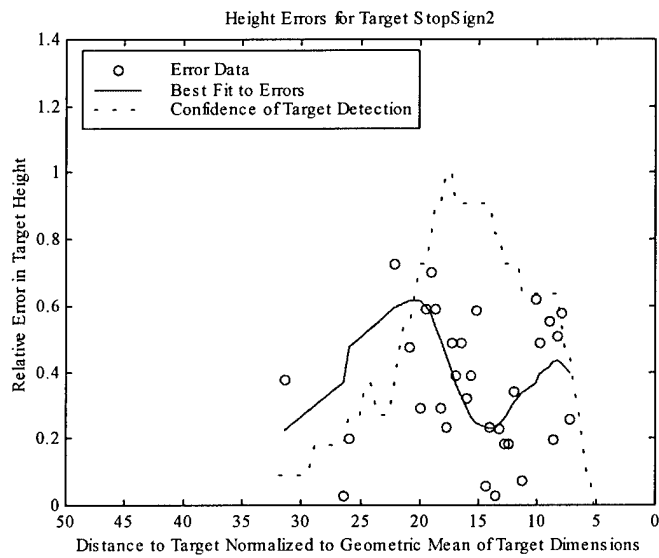


Figure 4.30 Relative height measurement errors for the stop sign target.

Finally, depth mapping may be improved by designing multiple range calibration models. The model used for this research is optimized for objects at distances between 1.5 – 3.0 m from the cameras. Selecting from different models based on disparity measurements may reduce errors in range calculations.

#### **4.6 OBJECT RECOGNITION**

The multiscale nominal frequency-based recognition described in Chapter 3 proved quite successful. This method was tested as a proof of concept only. An object library consisting of five different road signs was used for experiments. Four additional road signs were used in testing the object library. Three images were taken of each road sign at 10 m, 25 m, and 50 m (these measurements were approximate; see Figure 4.31). A single threshold was chosen for each object template to minimize the number of mistakes in classifying a particular sign. Then the thresholds were applied to the library and all objects were classified accordingly. Plots showing the effect of threshold choices on classification error are shown in Figures 4.32 – 4.36. False alarms increased as the tolerance threshold increased, while missed detections decrease. Ideally, there should be a point where no mistakes could be made. However, this situation only occurred in the case of the merge left sign. In all other cases mistakes were limited to one missed detection, and in each case, this was the sign photographed at 50 m away. But note that the image quality was greatly reduced at 50 m, and it should be expected that errors decrease with distance to the target.



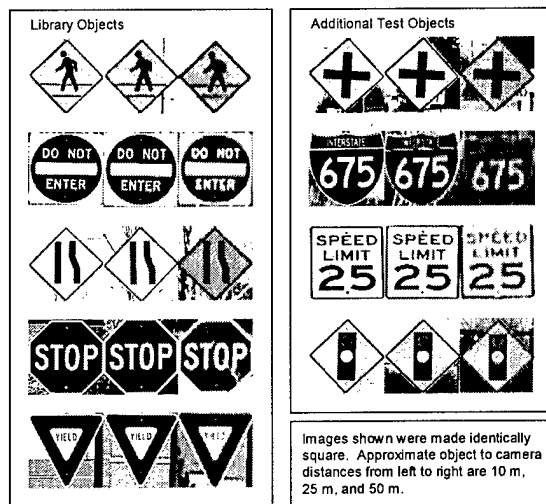


Figure 4.31 Object library and additional test objects.

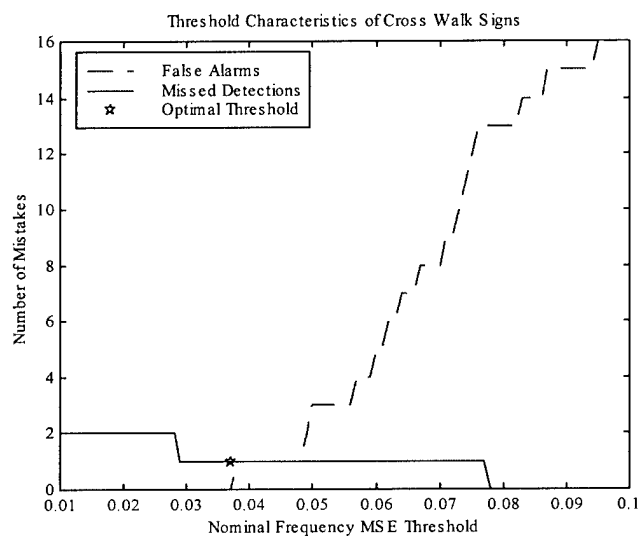


Figure 4.32 Threshold characteristics of cross walk signs.

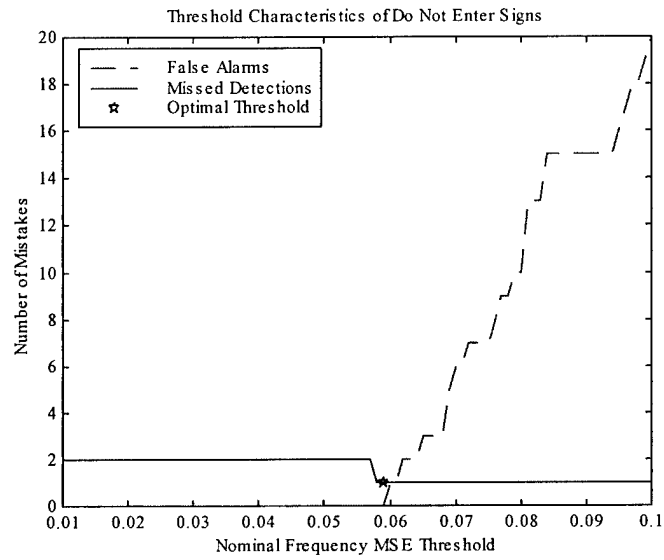


Figure 4.33 Threshold characteristics of do not enter signs.

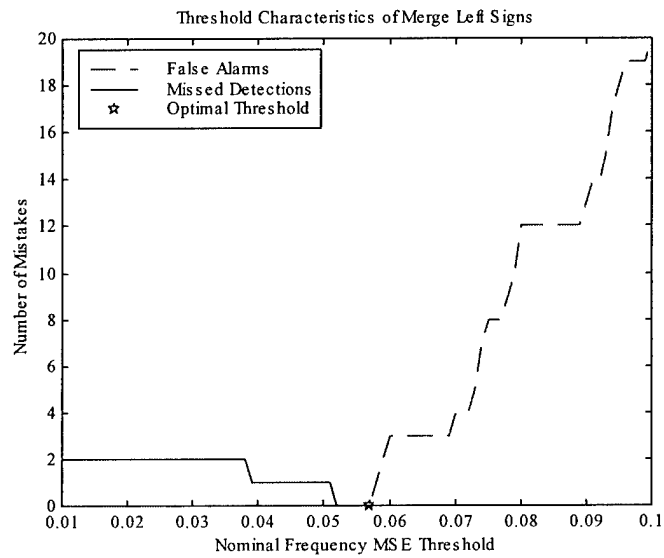


Figure 4.34 Threshold characteristics of merge left signs.

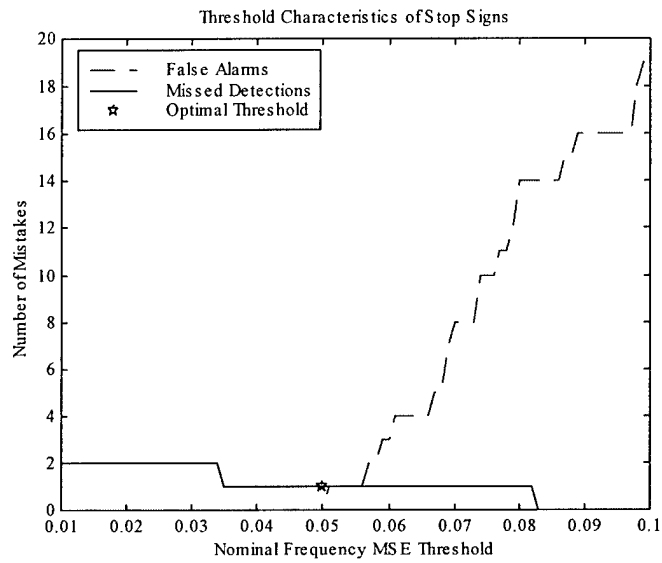


Figure 4.35 Threshold characteristics of stop signs.

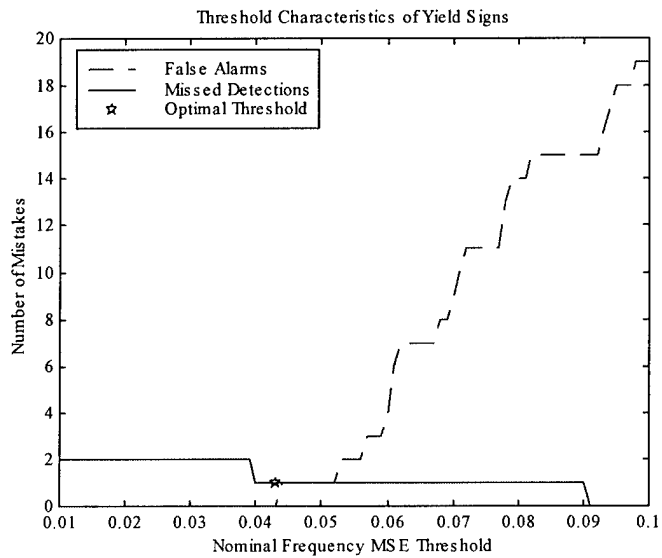


Figure 4.36 Threshold characteristics of yield signs.

#### **4.6.1 FUTURE RECOMMENDATIONS**

The multiscale nominal frequency-based recognition approach shows great promise for future applications in target recognition. One drawback is that a new set of template data is required for each view of a target, reducing the speed at which objects can be classified. Classification errors may be fine tuned by applying different thresholds that vary with frequency scale and that vary spatially for each object. This fine-tuning will help to eliminate the effect of background patterns on classification. Eventually templates could be designed to recognize varying generalizations of object classes. For instance, one template may be designed to recognize a particular person, while another may be designed to simply identify that the target is a person. Such a scheme may be used to further improve the speed of object classification. Note, however, that a prerequisite to robust target classification is to have precise target boundary definitions. Currently RavenVision does not produce target boundaries with enough precision to meet this criterion. For further information on the current state of the art in road sign recognition, see [5, 16].

#### **4.7 CONCLUSIONS**

This chapter discussed results and future improvements for all aspects of RavenVision. Most attention was directed toward results from target ranging and target recognition. All five aspects of the RavenVision system worked with some degree of success, and all have well-defined recommendations which will lead to future improvements in the overall system.

## 5. CONCLUSIONS

### 5.1 SUMMARY

Realtime stereovision processing consists of five steps. This research provided innovative contributions to the last four steps in addition to preprocessing. Preprocessing was shown to be ineffective for stereovision systems. The preprocessing method contributed by this research works; however, stereo rigs precalibrated by the manufacturer would be a more efficient avenue for correcting camera alignment.

Histogram zooming is a novel solution to feature clustering. In RavenVision, this process is performed on the right image frame prior to feature group correspondence with the left image frame. Histogram zooming functioned properly in most cases. Failures due to false detections and blocking effects. Depth mapping helped to alleviate false detections on road surface markings.

Feature group correspondence was accomplished by color cross optimizing the epipolar disparity (subject to the perspective constraint) of each histogram window in the left image frame. Improper feature correspondence was usually due to inadequate modeling of background noise surrounding the target.

Depth mapping was shown to have little meaning in the context of real-world constraints. Poor manufacturer specifications coupled with poor optics made a linear least squares model the clear choice for performing range calculations on target disparities. Due to severely band-limited detector characteristics, correspondence errors

of one pixel are significant, even at close range. In the MAV context, small camera separation is a complication. The only obvious solution is to process larger images at the expense of speed.

Multiscale nominal frequency-based recognition was shown to be a concept worthy of development for target recognition. This technique gets its speed advantage from branch elimination at each scale of comparison—the more data to be compared, the fewer candidate classes to compare. No false alarms occurred, but missed detections resulted from variations in image quality.

## **5.2 FUTURE RECOMMENDATIONS**

The first step in continuing this research should be to implement at least one baseline of comparison. This step would help to gage the progress of all future improvements relative to other systems.

Preprocessing should be replaced with a custom-made miniature stereo rig. Then histogram zooming should be modified to operate on line-detected windows for better clustering characteristics. Additionally, further research should be done on methods for tracking objects through multiple frames. The line-detected results should be polygonized and compared to corresponding polygonal patterns in the left image frame. Perhaps this new technique and color cross optimization could be used as part of a weighted expert systems model for determining the correct disparity for objects. Some

form of predictive modeling should be used to further assist the feature group correspondence process.

Depth mapping may be improved by designing multiple range calibration models. The one used in this research is optimized for objects between 1.5 – 3.0 m. Selecting an appropriate model that depends on disparity measurements may reduce ranging errors. Coupling this model with a predictive model may assist in interpolating target range information between successive image frames.

Future improvements to multiscale nominal frequency-based object recognition should include multithreshold support dependent on both frequency and scale. This improvement will make the recognition scheme more robust, possibly at the expense of speed.

## ***A. EMULATING STOP&GO***

### ***A.1 OVERVIEW***

This appendix discusses the theory used to generate a model for Stop&Go for purposes of baseline comparison. Unfortunately, the system was not implemented with measurable success. These details are provided as supplementary information only, and are referenced elsewhere in the document as background information. Refer to [3, 12] for information regarding the Stop&Go system.

### ***A.2 PIXEL CLASSIFIER***

After preprocessing has taken place, feature extraction needs to be performed. Stop&Go has a pixel classifier for feature extraction that uses information about a pixel and its neighbors to determine in which class the pixel belongs. The epipolar constraint requires that rows in one image frame correspond to the respective rows in the other image frame. Therefore the correspondence problem is limited to a row-by-row search. Pixel classification further narrows this search to a class-by-class search within each row; see Figure A.1 for a diagram. In general, having more classes equates to faster correspondence. The trade off is that having more classes also equates to increased potential for error from mismatched classes. These errors occur when the elements of a feature pair are classified differently. Too few classes, however, also introduces more error when unpaired features arise. This problem is solved by allowing small misclassification errors. The unpaired features remaining outside the misclassification



tolerance are discarded. The optimal number of classes varies with the average number of features extracted per image row. Stop&Go uses 81 classes—72 that are actually valid.

Compact data structures are key to fast feature correspondence, but they are also required for pixel classifier design. The RavenVision system data structures are organized by row, class, and column, in that order. Figure A.2 shows the data structures used. There is an array of rows, each pointing to an array of all possible classes. Each class points to a variable-length list of all pixels corresponding to that class sorted by column. The data structure for each pixel contains fields for representing the row and column indices for that pixel. This data structure also includes additional fields for information such as the class in which the pixel belongs.

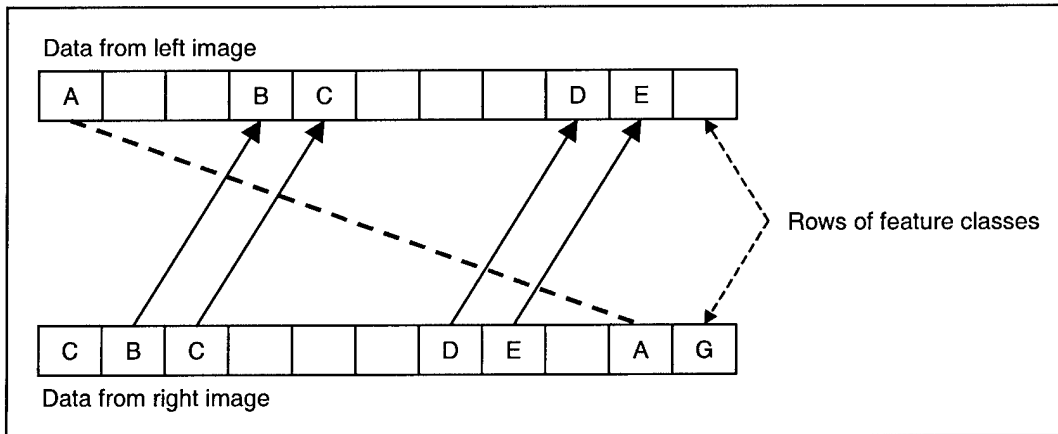


Figure A.1 Pixel class correspondence

Stop&Go uses a contrast pixel classifier. This research also supplies an original color-contrast pixel classifier. The Stop&Go contrast classifier compares a pixel to its four nearest neighbors (top, bottom, left, and right adjacent pixels); see Figure A.3. Each neighboring pixel is determined to be either darker, lighter, or the same shade of gray as the center pixel. This presents 81 possible classes of pixels. To narrow the scope of detection to vertical edges, the eight classes with only horizontal edges are discarded, and the homogeneous class is also discarded. The contrast pixel classifier operates in the context of the remaining 72 classes.

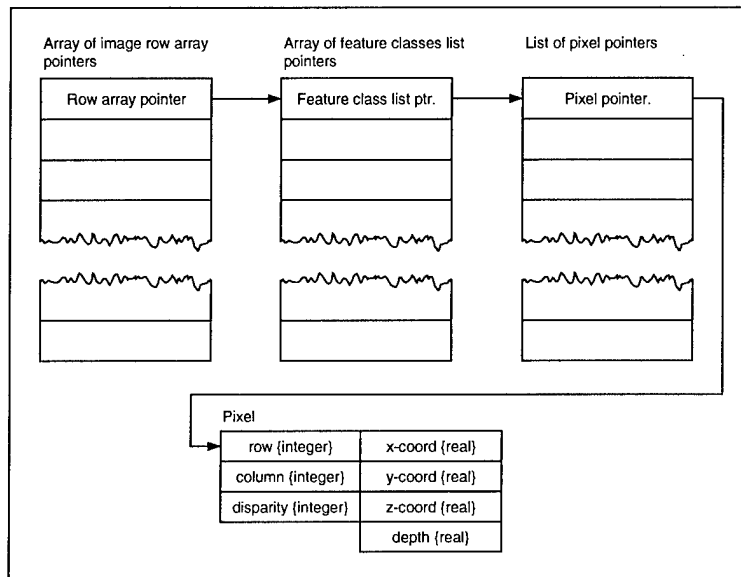


Figure A.2 Feature-related data structures

While the contrast classifier used by Stop&Go is fast because of its monochrome

input, this research presents an alternate design for a color classifier. The color classifier uses two heuristics. The first is a Boolean decision regarding the similarity of two colors.

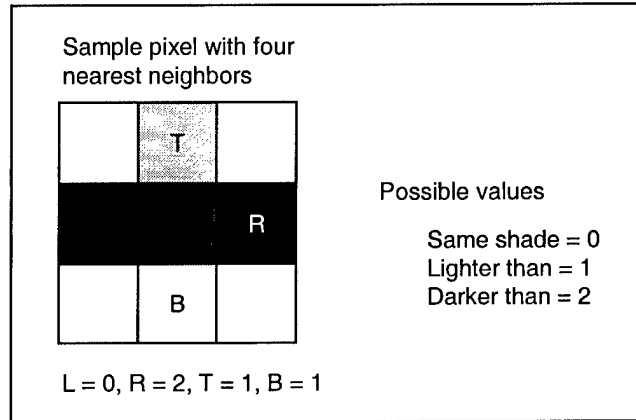


Figure A.3 Contrast pixel classifier

The second is a decision regarding the dullness of two colors. The details of this decision are presented in the flow diagram of Figure A.4. The decision about similarity between colors hinges on whether the reference pixel is gray or not. The grayness or dullness of a pixel is inversely related to the variance of the pixel channels. The variance is used in a Boolean decision as to whether the reference pixel is duller or more colorful than the neighboring pixel.

The organization of the color classes is less trivial than that of the contrast classifier. Relative dullness only introduces a new class when the corresponding parent pixels are also different colors. The resulting 81 classes are compatible with Stop&Go.

Again, only the vertical edges are kept, so only 72 classes are actually used by the classifier. Figure A.5 shows how classes are organized.

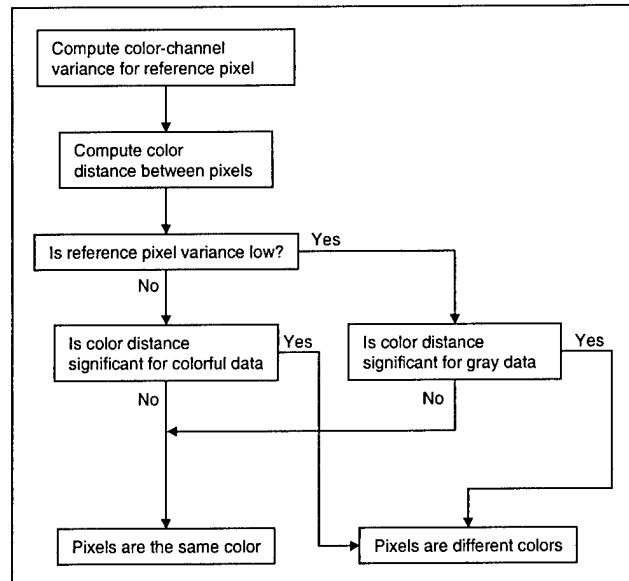


Figure A.4 Determining color likeness

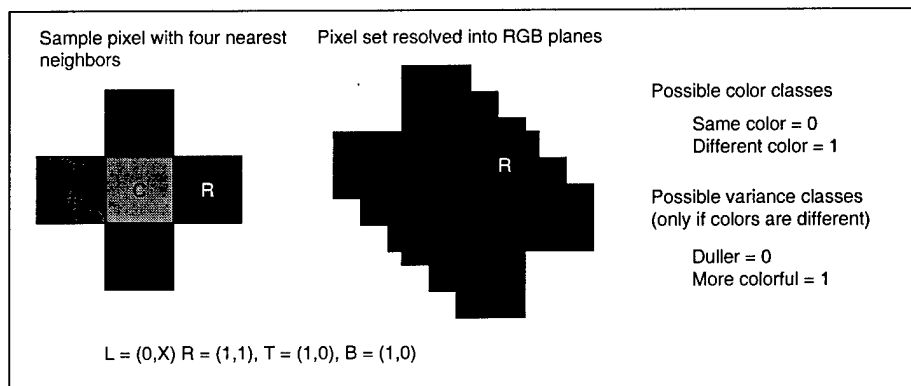


Figure A.5 Color pixel classifier

### A.3 CORRESPONDENCE

Feature correspondence is the trickiest process in any stereovision system. This step has the greatest potential for error, so it is vital to handle it in a robust manner. Figure A.6 contains a flow diagram of the correspondence process.

The process begins by selecting the first row of the image set. Within that row the first feature class is selected. If there are features belonging to this class in both images for the row under consideration, then an attempt is made to match the features into pairs. This matching process begins by selecting the *leftmost* remaining candidate feature in the *right* image. This feature is paired to the first valid unmatched candidate in the *left* image. Candidacy is valid for all features in the *left* image positioned to the *right* of the feature being matched from the *right* image. By pairing the first feature encountered minimum disparity is guaranteed. This guarantee is important because periodic structures can interfere with the correspondence problem [3].

After the two features are matched, the process repeats for all remaining features in the same class. If no valid unmatched features are available, existing matches are evaluated for rematch. If matching a leftover feature to an existing matched feature reduces disparity, then the old match is broken and the new leftover feature is treated as an erroneous response. Attempting to rematch the more recent leftover feature would violate the ordering constraint (see to Figure 2.7). After no more matches can be made, the process repeats for other classes and rows until the image feature set is completely corresponded.

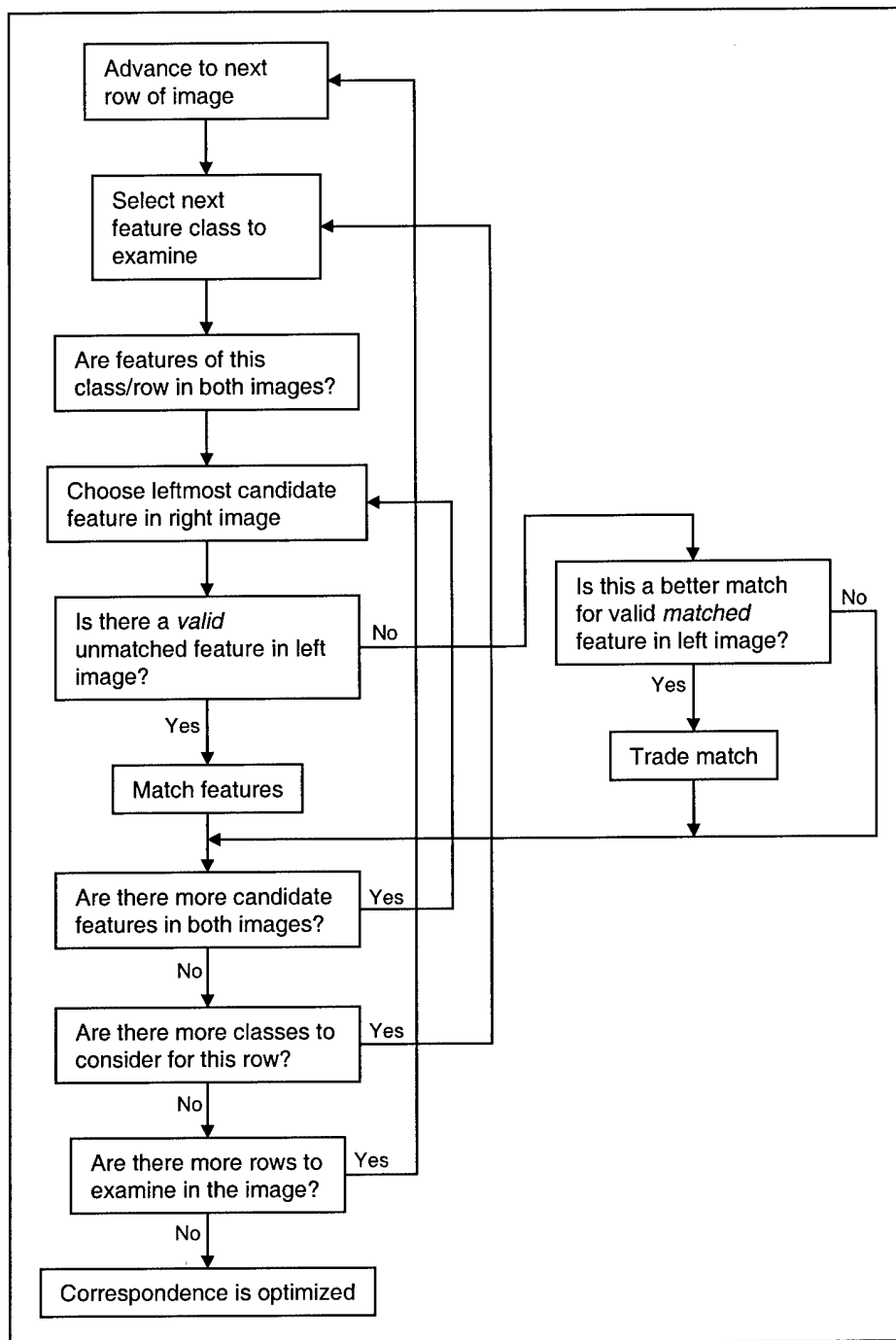


Figure A.6 Correspondence flow diagram

#### A.4 DATA VALIDATION

Inherent errors always plague data, and give rise to the need for a robust validation process. The best validation methods account for apriori error characteristics. Figure A.7 illustrates one potential error scenario. In this situation feature  $E_1$  in the right image is matched to  $E_1$  in the left image. Then, according to the flow diagram in Figure A.6, the match is traded later for  $E_2$  in the right image. Here  $E_2$  should have been mapped to feature  $A_2$ , but this feature was misclassified due to noise in the detector. The result is an inappropriate disparity. Low disparity values mean feature coordinates far from the image plane and usually results in mapping to coordinates not within the FOV of either camera.

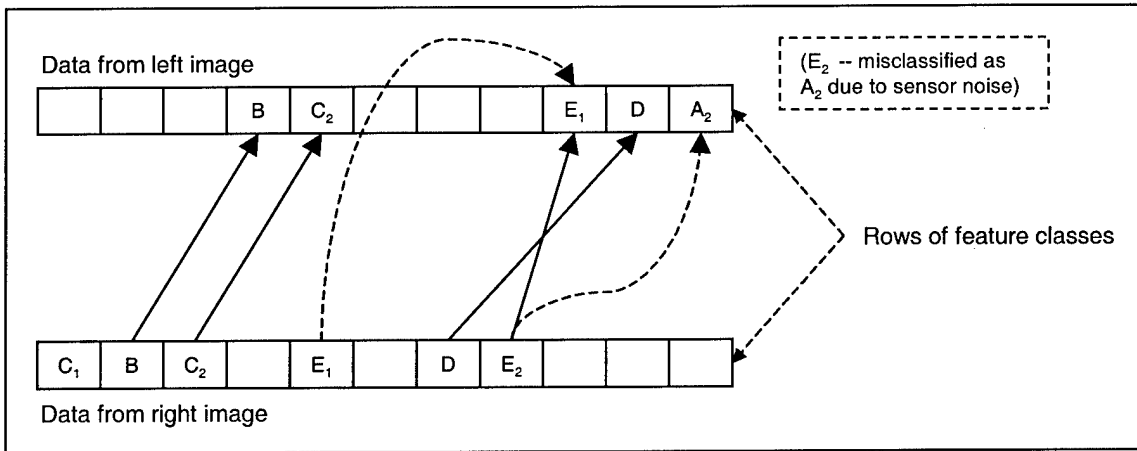


Figure A.7 Possible correspondence error scenario

Histogramming is a technique that can help detect and remove erroneous data from the feature set. Consider a set of feature points in an image that displays the ground plane. There will probably be false points having coordinates below the ground plane. In a system that does not use apriori details about camera height, the ground plane is unknown. A histogram of the data will show that the first peak always corresponds to the ground plane. All data below this plane is corrupted and must be removed. A more detailed representation of this process is shown in Figure A.8.

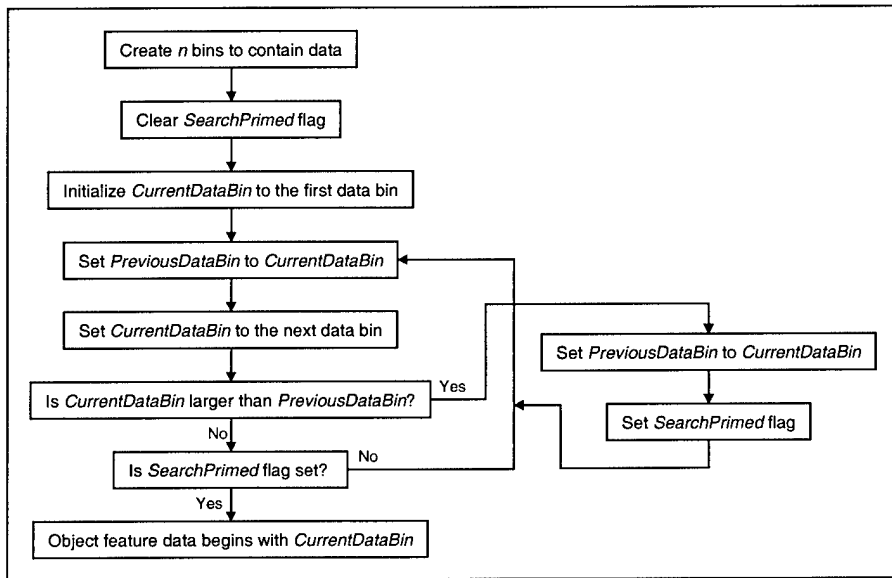


Figure A.8 Histogramming flow diagram

The difficulty with histogramming is choosing an appropriate bin size. Undersized bins cause peaks to go unnoticed, while oversized bins result in extraneous



peaks. Figure A.9 shows how inadequate bin size can hamper the effectiveness of the histogramming technique for ground plane detection.

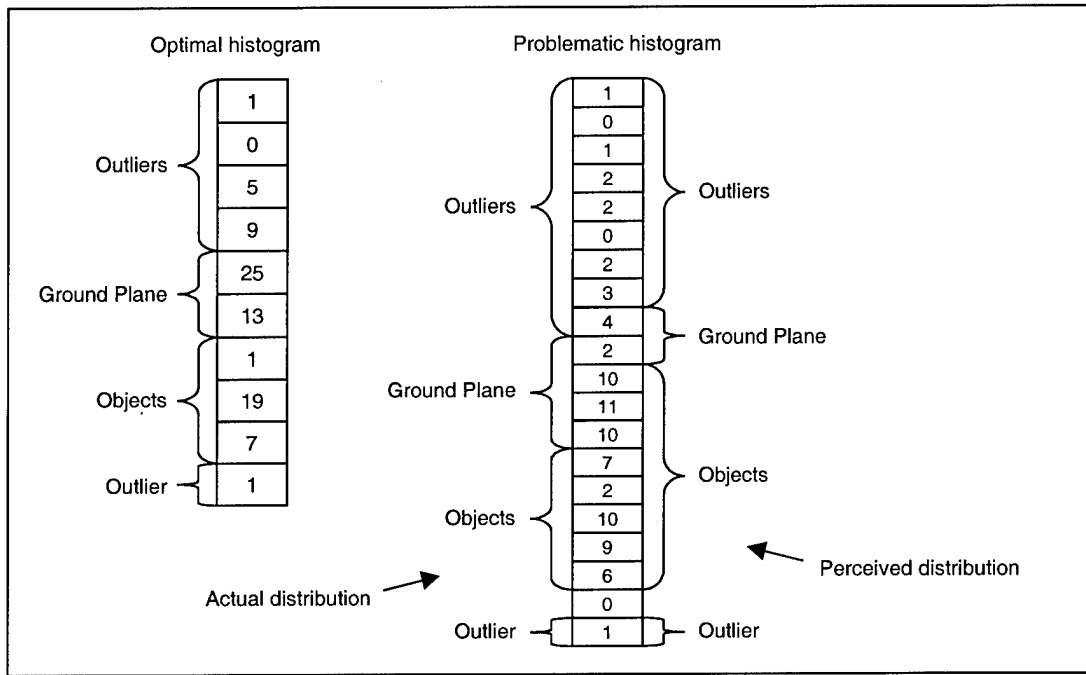


Figure A.9 Effects of bin sizes on histogramming

Histogramming only provides error correction for feature coordinates below the image plane. Imposing  $x$ ,  $y$ , and  $z$  world limits is used to crop most of the remaining errors. Figure A.10 presents this idea.

### A.5 DATA CLUSTERING

Once a set of valid features is obtained, the features are ready to be clumped together into objects. However, histogramming is well suited for this task.

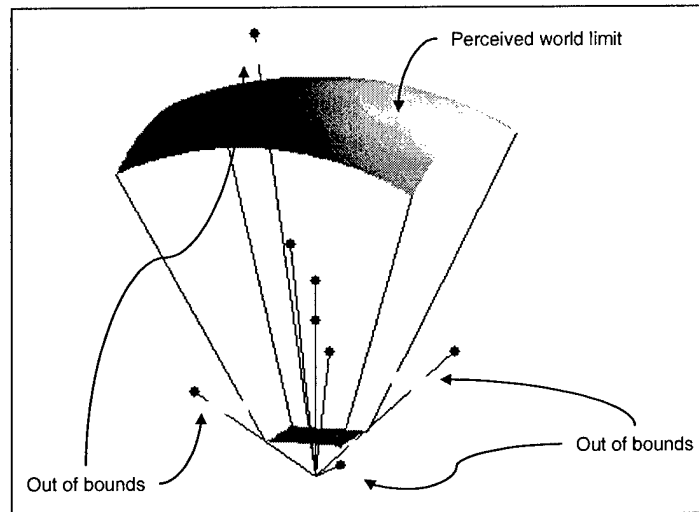


Figure A.10 Reducing data by cropping to world boundaries

Histogramming is only used in the  $z$ -direction. The  $x$  and  $y$ -directions have multiple peaks per object, as illustrated by Figure A.11. Consequently, there is no way to know how many peaks occur per object, so  $x$  and  $y$  histogramming is fruitless.

Figure A.12 diagrams the clustering algorithm. The first step is to  $z$ -histogram the data. Then all local maxima and surrounding foothills are grouped as individual objects. Each object location with respect to the image plane is determined, using an average of all coordinates of features representing the object. A bounding rectangle is fit to the features on the image plane. The calculated distance is applied to the center pixel of the bounding rectangle to determine the real-world coordinates of the object. This is accomplished until all objects have been located.

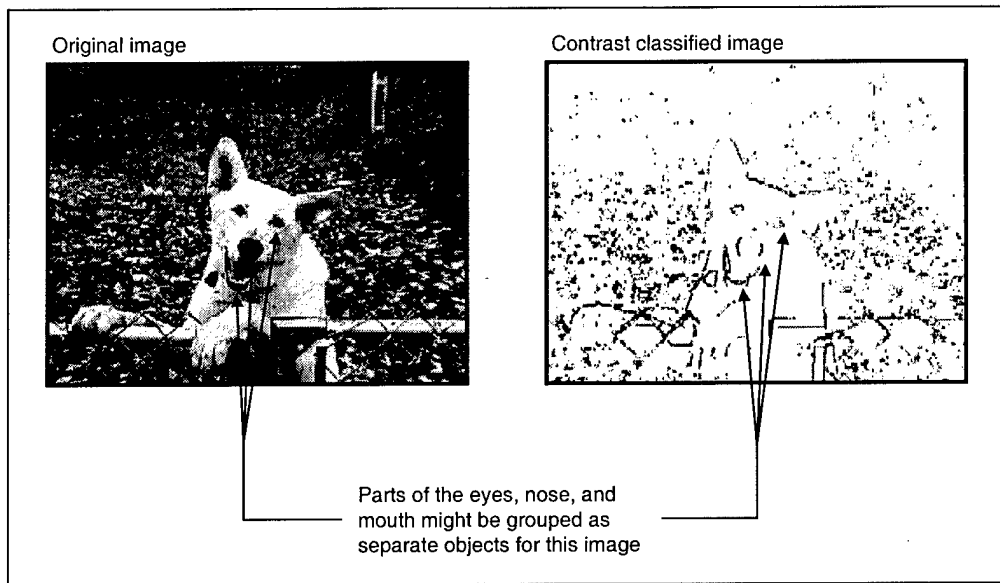


Figure A.11 Disadvantages of  $x$  and  $y$ -histogramming

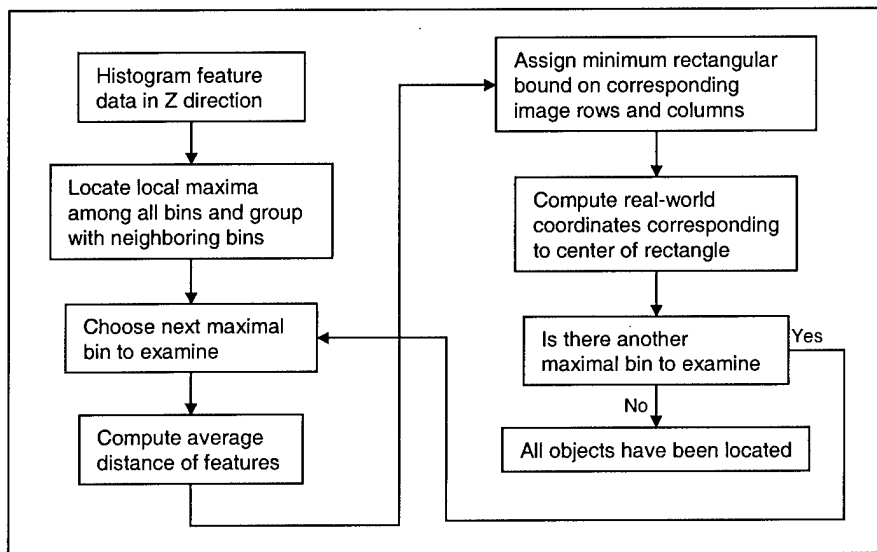


Figure A.12 Flow diagram of data clustering algorithm

## **A.6 SUMMARY**

This appendix presented the feature extraction and correspondence processes used to emulate Stop&Go, as described in [3, 12]. Although the emulation was unsuccessful, a novel color classifier compatible with Stop&Go was also designed and implemented. This appendix serves to assist the reader's understanding of the material presented in Chapter 3 and Chapter 4.

## *GLOSSARY*

### **Charge Coupled Device (CCD)**

An array of photosensors which produce charges in response to being exposed to light. These charges are read off sequentially and simultaneously discharged before successive readings are taken.

### **Color Band**

A vector of colors formed by averaging individual color planes across either the rows or columns of a subimage.

### **Color Channel**

Either the red, green, or blue component of a colored pixel.

### **Color Cross Optimization**

The process of minimizing the average color distance between two color bands.

### **Color Distance**

The square root of the sum squared differences between color channels of two pixels.

### **Color Plane**

A monochrome image representing the values of one particular color channel of all pixels in an image or subimage.

### **Confidence of Detection**

The ratio of number of data points in some interval, indicating that a target has been detected, to the total number of points in that interval.

### **Correspondence Problem**

The problem of determining which features in the right image frame correspond to which features in the left image frame.

### **Depth Mapping**

The application of a model for calculating relative position, given a pair of properly corresponded features.

**Differential Global Positioning System (DGPS)**

A GPS system which uses a precisely located base station to correct for local errors in GPS data so that a roving receiver may be surveyed with greater precision.

**Edge Density**

Ratio of the number of feature points near an edge to the length of that edge.

**Edge Detection**

The process of creating a one-bit line drawing from an image.

**Epipolar Constraint**

The constraint that objects seen by both image sensors of a stereo camera rig are characterized by features which differ only by horizontal location within simultaneously captured image frames.

**Fast Fourier Transform (FFT)**

A very fast radix-two algorithm for computing the discrete Fourier transform.

**Feature**

A filtered characteristic of an image.

**Feature Clustering**

The process of locating groups of features which correspond to the same object.

**Feature Extraction**

The process of identifying features in an image or subimage.

**Feature Group Correspondence**

The process of solving the correspondence problem for a group or cluster of features.

**Field of View**

The horizontal or vertical viewing angle for an optical system.

**Global Positioning System (GPS)**

A constellation of satellites transmitting time-synchronized information used to calculate the global position of a receiver.

**Haar Wavelet Energy Distribution**

The energy distribution associated with the Haar wavelet transform. This is similar to an energy distribution of frequencies.

**Heads Up Display (HUD)**

A dynamic eye-level display for visually passing information to a pilot in a timely fashion.

**Histogram Zooming**

A process seeded by a two-dimensional histogram of feature points that uses an edge density heuristic to perform feature clustering.

**Image**

A two dimensional array of pixels.

**Image Frame**

Either the left or right image that was captured during stereo data collection.

**Image Point Isolation**

A technique for feature extraction in which each feature consists only of pixel coordinates for a single row and column.

**Image Segmentation**

A technique for feature extraction where multiple sets of pixel coordinates make up each feature.

**Inertial Navigation System (INS)**

An electronic feedback mechanism which senses three-space acceleration.

**Micro Air Vehicle (MAV)**

A miniature flying machine.

**Multilayer Perceptron (MLP)**

A back-propagation neural network consisting of at least one hidden layer, used for multidimensional function modeling.

**Object Ranging**

*See depth mapping.*

**Object Recognition**

The process of classifying objects according to feature representations.

**Occam's Razor**

The principle that the simpler of two functional solutions is always better.

**Ordering Constraint**

The assumed constraint that the horizontal ordering of corresponding features will not change significantly between left and right image frames.

**Perspective Constraint**

The constraint that a feature detected in the *right* image frame must correspond to a feature positioned further right in the *left* image frame.

**Phantoms**

Detected objects that do not really exist.

**Pixel (Picture Element)**

A vector of color brightness values corresponding to each color channel of the image.

**Probability Density Image (pdi)**

A monochrome image used in template filtering with high pixel values corresponding to high probability that an object of interest is centered about that pixel.

**Stop&Go**

A stereovision-centric autonomous automobile research program currently at Daimler-Benz in Germany.

**Subimage**

A matrix of pixels representing a portion of an image.

**Template Filtering**

The process of filtering an image for particular features that are characteristic of a given object in order to locate that object within the image. This results in generating a probability density image.

**Template Matching**

The process of identifying an object according to some predetermined template of features.



## *BIBLIOGRAPHY*

### **TECHNOLOGY REVIEW**

- [1] Brown, Alison and Randy Silva. "Video-Aided GPS/INS Positioning and Attitude Determination," NAVSYS Corporation, (1999).
- [2] Dornheim, Michael A. "Tiny Drones May Be Soldier's New Tool," Aviation Week and Space Technology, 42-48 (June 8, 1998).
- [3] Franke, Uwe, et al. "Autonomous Driving Goes Downtown," *IEEE Intelligent Systems*, 40-48 (1998).
- [4] Loegering, Greg. "The Global Hawk Navigation System: An Odyssey in the Development of an Unmanned Aerial Vehicle," Ryan Aeronautical Center: Northrup-Grumman. (1999).
- [5] Lalonde, Marc and Ying Li. "Road Sign Recognition: Survey of the State of the Art for Sub-Project 2.4," Collection Scientifique et Technique. Montreal: Centre de Recherche Informatique, (1995).
- [6] Lim, Jae S. Two-Dimensional Signal and Image Processing. Upper Saddle River, NJ: Prentice Hall, (1990).
- [7] Tredway, B. Reece. 12 years experience as a USAF Explosive Ordinance Disposal Technician, National Air Intelligence Center, Wright-Patterson AFB OH. Personal interview. July 17, 1999.

### **EDGE DETECTION**

- [8] Canny, John. "A Computational Approach to Edge Detection," IEEE Transactions on Pattern Analysis and Machine Intelligence, 679-698 (November 1986).
- [9] Davis, T. J. "Fast Decomposition of Digital Curves into Polygons Using the Haar Transform," IEEE Transactions on Pattern Analysis and Machine Intelligence, 786-790 (August 1999).
- [10] Lu, Siwei and Anthony Szeto. "Artificial Neural Networks for Edge Enhancement," IASTED Proceedings from the International Conference on Artificial Intelligence Application and Neural Networks, 74-81 (1991).

[11] Olson, Clark F. "Variable-Scale Smoothing and Edge Detection Guided by Stereoscopy," IEEE Computer Society Conference Proceedings on Computer Vision and Pattern Recognition, 80-85 (1998).

### **OBJECT DETECTION**

[12] Franke, U. and I. Kutzbach. "Fast Stereo Based Object Detection or Stop&Go Traffic," IEEE Proceedings of the Intelligent Vehicles Conference, 339-344 (1996).

[13] Tagliarini, Gene A., et al. "Application of Wavelet and Neural Processing to Automatic Target Recognition," SPIE Proceedings on Automatic Target Recognition, Vol. VII, 154-160 (April 1997).

[14] Williamson, Todd, et al. "A Specialized Multibaseline Stereo Technique for Obstacle Detection," IEEE Computer Society Conference Proceedings on Computer Vision and Pattern Recognition, 238-244 (1998).

### **DISPARITY ESTIMATION**

[15] Rabie, Tamer F. and Demetri Terzopoulos. "Stereo Color Analysis for Dynamic Obstacle Avoidance," IEEE Computer Society Conference Proceedings on Computer Vision and Pattern Recognition, 245-252 (1998).

### **OBJECT RECOGNITION**

[16] Ghica, Dan, et al. "Recognition of Traffic Signs by Artificial Neural Network," IEEE International Conference Proceedings on Neural Networks, 1444-1449 (1995).

[17] Ernst, D., et al. "Improvement of Object Classification in Image Sequences," SPIE Proceedings on Automatic Target Recognition, Vol. VII, 424-433 (April 1997).

[18] Gavrilu, D. M. and L. S. Davis. "Fast Correlation Matching in Large (Edge) Image Databases," College Park, MD: Computer Vision Laboratory Center for Automation Research, University of Maryland, (August 1994).

[19] Lu, Si Wei, and Andrew K. C. Wong. "Recognition and Locating Partially Occluded Objects by Hypergraph Representation," IASTED Proceedings from the Fourteenth International Symposium on Manufacturing and Robotics, 83-86 (1991).

[20] Piccioli, Giulia, et al. "A Robust Method for Road Sign Detection and Recognition," Proceedings from the Third European Conference on Computer Vision, 495-500 (1994).

## **TRACKING**

[21] Biao, Li, et al. "Automatic Target Detection and Tracking System Using Infrared Imagery," SPIE Proceedings on Automatic Target Recognition, Vol. VII, 526-532 (April 1997).

[22] Chen, Haixin, et al. "A New target Tracking Method for Optical/IR Image Sequences," SPIE Proceedings on Automatic Target Recognition, Vol. VII, 518-525 (April 1997).

[23] Darrell, T., et al. "Integrated Person Tracking Using Stereo, Color, and Pattern Detection," IEEE Computer Society Conference Proceedings on Computer Vision and Pattern Recognition, 601-608 (1998).

[24] Morris, Daniel D. and James M. Rehg. "Singularity Analysis for Articulated Object Tracking," IEEE Computer Society Conference Proceedings on Computer Vision and Pattern Recognition, 289-296 (1998).

[25] Oliensis, John. "Computing the Camera Heading from Multiple Frames," IEEE Computer Society Conference Proceedings on Computer Vision and Pattern Recognition, 203-209 (1998).

[26] Stein, Gideon and Amnon Shashua. "Direct Estimation of Motion and Extended Scene Structure from a Moving Stereo Rig," IEEE Computer Society Conference Proceedings on Computer Vision and Pattern Recognition, 211-218 (1998).

[27] Tommosini, Tiziano, et al. "Making Good Features Track Better," IEEE Computer Society Conference Proceedings on Computer Vision and Pattern Recognition, 178-183 (1998).

[28] Zhao, Liang and Chuck Thorpe. "Qualitative and Quantitative Car tracking from a Range Image Sequence," IEEE Computer Society Conference Proceedings on Computer Vision and Pattern Recognition, 496-501 (1998).

## **STRUCTURE FROM MOTION**

[29] Eveland, Christopher, et al. "Background Modeling for Segmentation of Video-Rate Stereo Sequences," IEEE Computer Society Conference Proceedings on Computer Vision and Pattern Recognition, 266-271 (1998).

[30] Feng, Xiaolin and Pietro Perona. "Scene Segmentation from 3D Motion," IEEE Computer Society Conference Proceedings on Computer Vision and Pattern Recognition, 225-231 (1998).

[31] Oliensis, John. "Multiframe Structure from Motion in Perspective," Proceedings from the First Annual NECI Vision Workshop. (1995).

[32] Shin, Min C. and Kevin W. Bowyer. "An Objective Comparison Methodology of Edge Detection Algorithms Using a Structure from Motion Task," IEEE Computer Society Conference Proceedings on Computer Vision and Pattern Recognition, 190-195 (1998).

#### **OTHER**

[33] Bishop, Christopher M. Neural Networks for Pattern Recognition. New York, NY: Oxford University Press Inc., (1995).

[34] Gonzalez, Rafael C. and Richard E. Woods. Digital Image Processing. Reading, MA: Addison-Wesley Publishing Company, (1992).

[35] Hiura, Shinsaku and Takashi Matsuyama. "Depth Measurement by the Multi-Focus Camera," IEEE Computer Society Conference Proceedings on Computer Vision and Pattern Recognition, 953-959 (1998).

**REPORT DOCUMENTATION PAGE**

*Form Approved*  
OMB No. 0704-0188

The public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing the burden, to Department of Defense, Washington Headquarters Services, Directorate for Information Operations and Reports (0704-0188), 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302. Respondents should be aware that notwithstanding any other provision of law, no person shall be subject to any penalty for failing to comply with a collection of information if it does not display a currently valid OMB control number.  
**PLEASE DO NOT RETURN YOUR FORM TO THE ABOVE ADDRESS.**

1. REPORT DATE (DD-MM-YYYY) 15-03-2000	2. REPORT TYPE Master's Thesis	3. DATES COVERED (From - To) Apr 1999 - Mar 2000
---	-----------------------------------	---

4. TITLE AND SUBTITLE REALTIME COLOR STEREOVISION PROCESSING	5a. CONTRACT NUMBER
	5b. GRANT NUMBER
	5c. PROGRAM ELEMENT NUMBER

6. AUTHOR(S) Formwalt, Byron P., 1 Lt, USAF	5d. PROJECT NUMBER
	5e. TASK NUMBER
	5f. WORK UNIT NUMBER

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) Air Force Institute of Technology 2950 P Street Wright-Patterson AFB, OH 45433-7765	8. PERFORMING ORGANIZATION REPORT NUMBER AFIT/GE/ENG/00M-08
---	--

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES) Mr. James S. Morgan AFRL/SNAT 2241 Avionics Cl, Building 620, Rm S3-S10 WPAFB OH 45433 COMM: (937) 255-1491 x3328 DSN: 785-1491 x3328	10. SPONSOR/MONITOR'S ACRONYM(S) AFRL/SNAT
	11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION/AVAILABILITY STATEMENT  
Approved for public release; Distribution unlimited.

13. SUPPLEMENTARY NOTES

14. ABSTRACT  
Recent developments in aviation have made micro air vehicles (MAVs) a reality. These featherweight palm-sized radio-controlled flying saucers embody the future of air-to-ground combat. No one has ever successfully implemented an autonomous control system for MAVs. Because MAVs are physically small with limited energy supplies, video signals offer superiority over radar for navigational applications.

This research takes a step forward in realtime machine vision processing. It investigates techniques for implementing a realtime stereovision processing system using two miniature color cameras. The effects of poor-quality optics are overcome by a robust algorithm, which operates in realtime and achieves frame rates up to 10 fps in ideal conditions. The vision system implements innovative work in the following five areas of vision processing: fast image registration preprocessing, object detection, feature correspondence, distortion-compensated ranging, and multiscale nominal frequency-based object recognition.

15. SUBJECT TERMS  
Stereovision, Color Stereovision, Realtime Stereovision, Computer Vision, Object Recognition, Pattern Recognition, Road Sign Recognition, Autonomous Vehicle Control, Color Image Processing, Realtime Image Processing

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT U	b. ABSTRACT U	c. THIS PAGE U	U	105	Mikel M. Miller, Lt Col, USAF
					19b. TELEPHONE NUMBER (include area code) (937) 255-6565 x4278