Automatic Positioning and Design of a Variable Baseline Stereo Boom

Peter L. Fanto

Thesis submitted to the Faculty of the
Virginia Polytechnic Institute and State University
in partial fulfillment of the requirements for the degree of

Master of Science
in
Mechanical Engineering

Kevin B. Kochersberger, Chair
A. Lynn Abbott
Mani Golparvar-Fard
Mike J. Roan

July 17, 2012
Blacksburg, Virginia

Keywords: Stereo boom, Automatic Positioning, Variable Baseline, Unmanned Systems
Copyright 2012, Peter L. Fanto

Automatic Positioning and Design of a Variable Baseline Stereo Boom

Peter L. Fanto

ABSTRACT

Conventional stereo vision systems rely on two spatially fixed cameras to gather depth information about a scene. The cameras typically have a fixed distance between them, known as the baseline. As the baseline increases, the estimated 3D information becomes more accurate, which makes it advantageous to have as large a baseline as possible. However, large baselines have problems whenever objects approach the cameras. The objects begin to leave the field of view of the cameras, making it impossible to determine where they are located in 3D space. This becomes especially important if an object of interest must be actuated upon and is approached by a vehicle.

In an attempt to overcome this limitation, this thesis introduces a variable baseline stereo system that can adjust its baseline automatically based on the location of an object of interest. This allows accurate depth information to be gathered when an object is both near and far. The system was designed to operate under, and automatically move to a large range of different baselines.

This thesis presents the mechanical design of the stereo boom. This is followed by a derivation of a control scheme that adjusts the baseline based on an estimate object location, which is gathered from stereo vision. This algorithm ensures that a certain incident angle on an object of interest is never surpassed. This maximum angle is determined by where a stereo correspondence algorithm, Semi-Global Block Matching, fails to create full reconstructions.

Acknowledgments

I would like to take this time to acknowledge some of those that have helped along this arduous journey. I would like to begin by thanking my committee members for helping with polishing out my ideas and leading me in the right direction. Without your help, none of this would have been possible. I'd especially like to thank Dr. Kochersberger for providing me with the means to continue schooling and an awesome place work, not to mention all of the good times and laughs. It has been a pleasure.

That brings me to the Unmanned Systems Lab. I would like to thank all the guys that have been there for me through thick and thin. You have all made graduate school a very enjoyable experience. I don't know if I could have made it through without having the distractions and entertainment that you all provided. I would especially like to thank Jason Gassaway for teaching me about stereo vision and getting me up to speed with everything at the lab. Thanks for also taking the time to help me out even after you had left the lab and begun working. I really appreciate it. I would also like to thank Kenny Kroeger for knowing everything that was going on at the lab and for teaching me how to use the CNC. I would like to thank Bob Collins for being an awesome roommate and teaching me Abiqus. Thank you Bryan Krawiec for those semantical conversations and for also being an awesome roommate. Thanks also to Troy Probst for pretending to work for hours on end (inside joke, he does work occasionally). It appears that I must leave the room of pretend and use those skills elsewhere. All of you at the lab have been great friends and I truly appreciate it.

I would also like to thank Bryce Lee for helping me formulate the idea of a variable length stereo boom over lunch.

Lastly, I would like to thank my family for providing with a means and desire to pursue higher education. Thank you.

All images are from the author unless otherwise stated.

Contents

List of Figures

List of Tables

List of Algorithms

Acronyms and Abbreviations

2D	Two-dimensional
3D	Three-dimensional
BP	Belief Propogation
FAST	Features from Accelerated Segment Test
FEA	Finite Element Analysis
GPU	Graphical ProcessingUnit
ICP	Iterative Closest Point
ORB	Oriented-FAST Rotated BRIEF
SGBM	Semi-Global Block Matching
SIFT	Scale-invariant Feature Transform
SURF	Speeded Up Robust Features
USB	Universal Serial Bus
USL	Unmanned Systems Laboratory
deg	Degrees
in	Inches
m	Meters

Chapter 1

Introduction

The navigation of ground robotics oftentimes necessitates the use of sensors to gather three-dimensional(3D) information to describe the environment. The 3D information is important for both well organized environments as well as ones as vast as an extraterrestrial body. There are a few different ways to gather this information. Many systems rely on two spatially separated cameras that utilize two-dimensional imagery in order to estimate the depth of the scene. These systems are known as stereo vision systems. Essentially, they are attempting to gather 3D information in a similar fashion as eyes do. Stereo vision requires multiple cameras which are viewing the same environment and have a distance between them, or baseline, in order to collect these measurements. The larger the baseline, the better the estimates of the environment become. A larger baseline would therefore seem to be more advantageous. However, a large baseline introduces problems whenever an object of interest approaches the cameras. The object would be more likely to leave the field of view and make it impossible to gather depth information about the object.

This thesis will attempt to address this limitation by designing and constructing a variable baseline stereo vision system. The variable baseline stereo vision system will be able to adjust the width of its baseline, which would make it possible to gather accurate depth information while objects are far away, and yet still gather depth information when they approach the

cameras. A system of this nature will become especially important if an object of interest needs to be actuated upon. For instance, if you wanted to drive up to an object and then collect it for further analysis, this rig would allow you to draw near to it without losing it in the field of view of the cameras. It could also be used in the construction industry in order to track the position of personnel or machinery in the environment. This information could make the work site safer. It would be very advantageous to machinery operators if it were mounted on top of their machines, like an excavator for instance. They would be able quickly to determine if something is within their reach or not.

1.1 Contribution

The contributions from this work are twofold. First, it presents the design of a stereo boom that is capable of placing two cameras at a large range of baselines. This stereo boom is constructed of an outer aluminum tube that contains two cameras within it. These cameras rest on linear bearings and are connected to a gearing rack, which can be driven by a motor. This allows the cameras to be placed at many different baselines. The second contribution from this work is a control algorithm that is capable of automatically adjusting the baseline of the cameras based on an object's location in 3D space. This control technique relies heavily on the abilities of stereo correspondence algorithms. These algorithms are used to find matches between different images so that 3D information can be gathered. However, they begin to fail as an object looks more dissimilar in both images, which occurs whenever the object approaches the cameras. When an object gets too close, the stereo correspondence algorithms lose their ability to create 3D reconstructions altogether. This problem can be remedied by shrinking the baseline. This control algorithm automatically positions the cameras to ensure that stereo correspondence algorithms will be able to gather depth information about an object of interest.

1.2 Organization

This work will begin with a discussion about different aspects of computer vision. This discussion will introduce those topics that are important for understanding this work, and will mainly focus on gathering depth information from stereo vision. At the end of the section, previous works that are similar in nature to this one will be introduced. This will provide an understanding of what other people have done, and lend to the novelty of this project. The background information will be followed by a discussion of the mechanical design of the variable baseline stereo boom. It will attempt to use stereo vision parameters in order to define the design specifications. This section will also include a finite element analysis (FEA) to determine the structural integrity of the boom. It will conclude with a description of the final boom. This section will be followed by a look at whether the actual rig met the design specifications by analyzing different calibration results from the boom. All of this will be followed by the description of the control scheme that is capable of adjusting the width of the baseline automatically. After this, will be an analysis of how well the control algorithm would perform based on positioning estimates. The thesis will end with conclusions that can be drawn from the work as well as some recommendations for future improvements.

Chapter 2

Background and Previous Works

This chapter will present information that is vital for understanding the subsequent portions of this thesis. It will begin with a description of the pinhole camera model, which will provide insight into the projective geometry that is critical for stereo reconstructions. This will be followed by a discussion of the main principles of stereo vision, which includes a look at different stereo vision matching algorithms, and notably the one that will be used for many portions of this thesis, semi-global block matching (SGBM). Various sparse feature point detectors will then be presented. These will become important for determining how well the system functions. After creating a basic understanding of cameras and stereo vision, camera calibration will be presented. In particular, the Camera Calibration Toolbox for Matlab will be discussed [1]. The chapter will conclude with a brief description of the imaging hardware that was used as well as previous works that are similar to the variable baseline stereo boom.

2.1 Camera Geometry

This section will introduce the pinhole projective camera, which is the simplest camera model and also the most important to this work. The pinhole camera is one that projects 3D coordinates to the 2D image plane through a single point, or pinhole, as seen in Figure 2.1. **C** is the center of projection of the camera and is also known as the camera center. In other

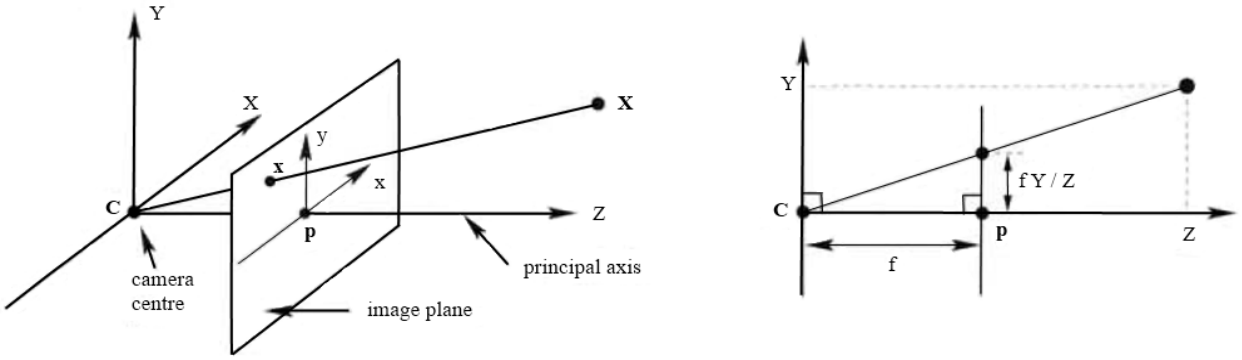


Figure 2.1: Projective geometry of a pinhole camera. Image courtesy of Hartley et al. [2] [used under fair use guidelines].

words, it is the point that all of the light passes through before striking the imaging sensor of the camera (CCD array). The image plane is the plane that contains the projections of the 3D world. It is what would be thought of as the image. The image plane is depicted along the positive Z axis in Figure 2.1. However, in a physical camera the image plane would be located along the negative Z axis. All of the light would pass through the camera center and strike the image plane causing the image to be inverted. Similar triangles allows the image plane to be represented along the positive Z axis making it simpler to conceptualize. The location of the imaging plane along the Z axis is dependent on the focal length, f, which is in terms of physical distance. **p** is the principal point of the camera and is the point where the principal axis intersects the image plane. The principal axis is the line which is orthogonal to image plane and passes through the camera center. The principal axis is parallel to the Z axis in terms of camera geometry .

Let $\mathbf{X} = [X \quad Y \quad Z \quad 1]^T$ be the coordinates of a point in 3D space in homogeneous

coordinates and $\mathbf{x} = [x \quad y \quad 1]^T$ be the homogeneous coordinates of the projected object on the image plane relative to the principal point. Homogeneous coordinates allow nonlinear mappings to be written in terms of linear algebra [2]. Let P be the camera projection matrix. From similar triangles, it can be shown that $\mathbf{X}$ can be projected onto the image plane by way of the projection matrix, as shown in equation 2.2.

$$Z\mathbf{x} = P\mathbf{X} \tag{2.1}$$

$$Z \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = \begin{bmatrix} f & 0 & 0 & 0 \\ 0 & f & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} \tag{2.2}$$

The projection matrix shown above relates 3D space to the image plane with the origin of the coordinate frame located at the principal point. However, the position of the principal point can change between cameras and it is therefore easier to relate the 3D coordinates to the x and y position in terms of pixel coordinates. The origin in terms of pixel coordinates is the top left corner of the image, with the positive x axis in the rightward direction and the positive y axis in the downward direction. The projection matrix shown above also neglects the fact that cameras could have non-square pixels. This could lead to different focal lengths in both the x and y directions, i.e., $f_x \neq f_y$. Incorporating these additional parameters gives a projection matrix of

$$\mathbf{x} = P\mathbf{X} = \begin{bmatrix} f_x & \alpha & c_x & 0 \\ 0 & f_y & c_y & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \mathbf{X}$$

$$\mathbf{x} = K \begin{bmatrix} \mathbf{I} & \mathbf{0} \end{bmatrix} \mathbf{X} \tag{2.3}$$

where

$$K = \begin{bmatrix} f_x & \alpha & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix} \tag{2.4}$$

K is known as the camera calibration matrix and describes all of the intrisic properties of a pinhole camera. These are the characteristics of the camera that will not change from one image to the next, even if the camera changes its orientation. f_x and f_y are the focal lengths of the camera in terms of pixels along their respective directions. The point (c_x, c_y) is the pixel location of the principal point in the image. The last parameter, α, describes the skewness of the pixels. Skewness becomes important if the pixels are non-rectangular and is equal to zero in most modern cameras [2].

Equation 2.3 maps a 3D position in camera coordinates to a location in 2D pixel coordinates. Sometimes it becomes important to relate the camera position to some sort of global coordinate system. For instance, stereo vision uses global coordinates to relate the one camera to the other. This requires that the projection matrix include a rotation, R, and translation, T. The final projection matrix that takes the rotation and translation into account is shown below.

$$P = KR \begin{bmatrix} \mathbf{I} & T \end{bmatrix} \tag{2.5}$$

where $R \in \mathbb{R}^{3x3}$ and $T \in \mathbb{R}^{3x1}$. R describes the roll, pitch, and yaw differences between the coordinate frames, and T describes the horizontal, vertical, and forward offsets between the camera center and the origin of the coordinate system.

2.2 Stereo Vision

This section will introduce stereo vision and will build on the information presented about projective geometry of a pinhole camera. Stereo vision uses multiple cameras in order to

estimate the position of an object in 3D. This is similar to how humans see.

2.2.1 Stereo Vision Fundamentals

The driving principle behind stereo vision is triangulation. Assume that there are two spatially separated pinhole cameras that have the same focal length, $f_L = f_R = f$, have the same principal points, $c_L = c_R$, and have parallel image planes. Note that f is in terms of pixels. Also, assume that everything viewed in one row of the left image can be seen in the same row of the right image. This configuration will be referred to as the cameras being frontal parallel and can be seen in Figure 2.2. B is the baseline of the stereo vision system and is the distance between the camera centers, x_L and x_R are the locations of the object in the left and right image planes respectively, and c_L and c_R are the principal points of each camera in pixel coordinates.

From Figure 2.2, the distance to the object, $\mathbf{X}$, can be determined by similar triangles. Equation 2.6 shows how to triangulate the 3D position of an object.

$$\frac{Z}{B} = \frac{f}{d} \quad \Rightarrow \quad Z = \frac{Bf}{d} \tag{2.6}$$

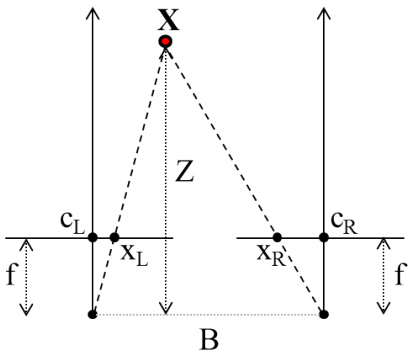


Figure 2.2: Stereo triangulation between two cameras

with

$$d = x_L - x_R \tag{2.7}$$

where d is the disparity of the object and Z is the distance to it. The disparity describes the difference in the locations of the object as seen in both images. It can be directly used to determine the distance to an object if two images are in the frontal parallel position [3]. However, in reality, this is not always the case. The two cameras oftentimes have rotations and translations between them causing them to be out of parallel. The next section will introduce the concept of epipolar geometry, which will allow us to rectify the images so that they are placed in the frontal parallel position.

The process of gathering depth information from disparities is also known as reprojection. It is called this because the points that have been projected onto the image plane are then projected back into 3D space. This allows multiple 2D images to gather depth information about the scene. Reprojection is not exact, and there are errors associated with it. They stem from the fact that the images are made up of a finite number of pixels. The position of an object is only known to within a 3D region of space that is created for a given disparity, leading to errors associated with the reprojection of the points. Figure 2.3 illustrates how these errors arise. Every object that is located within the green region of the figure, will be represented by the same pixel and will therefore have the same disparity. This is where the errors occur.

The error of greatest concern to this work is the distance error, ΔZ. This can also be thought of as the maximum achievable depth resolution. The maximum depth resolution can be determined by taking the derivative of equation 2.6 with respect to the disparity [4]. Solving equation 2.6 in terms of d and substituting into the derivative gives

$$\Delta Z = \frac{Z^2}{Bf}\Delta d \tag{2.8}$$

where ΔZ is the depth resolution, Z is the distance to the object, f is the focal length in

pixels, B is the baseline, and Δd is the smallest disparity possible in pixels. Equation 2.8 shows that the depth resolution decreases as the distance to the object increases. The depth resolution will play a role in future sections for justifying the need for an variational baseline stereo vision system. The errors in terms of the X and Y positions are greatest at the edges of the imagery, which will also be mentioned later.

2.2.2 Epipolar Geometry

Epipolar geometry is used to describe the rotations and translations between two pinhole cameras. The geometry gets its names from points known as epipoles. Epipoles are points on the image planes that are formed from the intersection of the image plane with a line connecting the camera centers, as shown in Figure 2.4. x_L and x_R are the locations of an

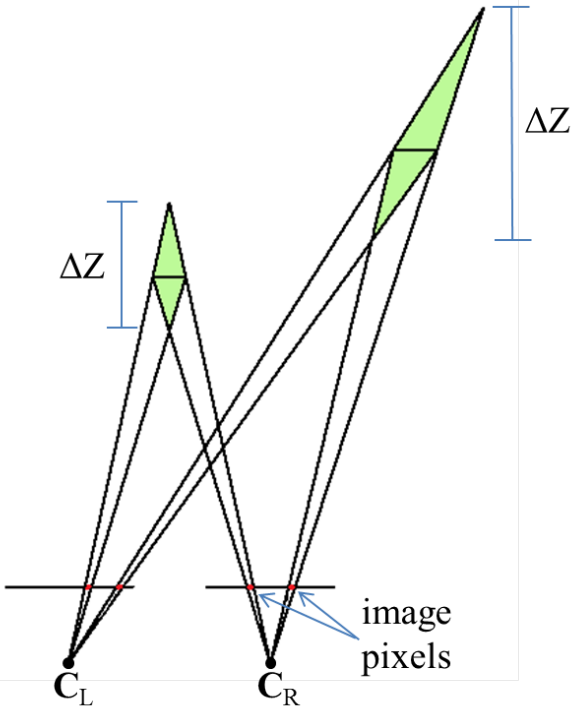


Figure 2.3: Illustrates the resolution achievable from stereo imagery. The pixels on the image plane are the same size and carve out the location that can contain the object(shown in green).

arbitrary point, $\mathbf{X}$, projected onto the image planes, e_L and e_R are the epipoles associated with the cameras, and $\mathbf{C}_L$ and $\mathbf{C}_R$ are the camera centers in 3D space.

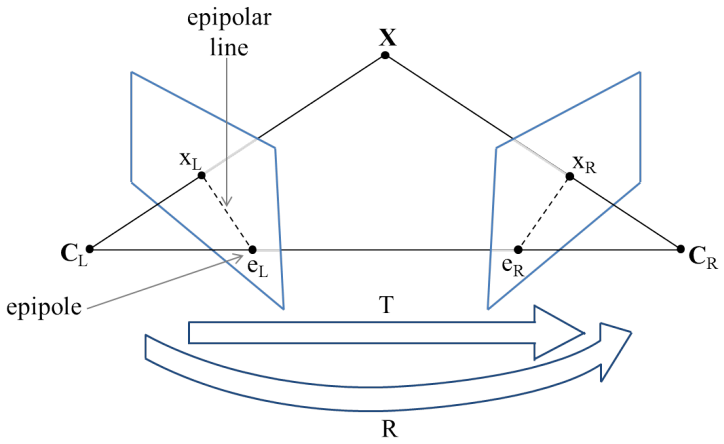


Figure 2.4: Epipolar geometry between two cameras

The line that passes through x_L and e_L and the corresponding line through x_R and e_L are epipolar lines. Everything on an epipolar line in the left image can be found on the corresponding epipolar line in the right image. This is referred to as the epipolar constraint. The epipolar constraint plays an important role in stereo vision, because stereo vision requires that the same objects are viewed in both images. This necessitates that the algorithm searches for matching locations in both images. Since objects are confined to the same epipolar lines, the search space is drastically reduced once the lines are found. This means that, theoretically, only one line needs to be searched for the matches instead of the entire image. Therefore, it becomes very important for timely stereo reconstructions.

To simplify the stereo algorithm further, it is advantageous to rectify the images such that the epipolar lines are horizontal. This can be done by letting the position of the epipoles go to infinity, which defines rotations that the two image planes must undergo. The intrinsic parameters of both cameras are recomputed to allow $f_L = f_R = f$ with f_x equaling f_y, and the principal points $c_L = c_R$. The method also rectifies the camera centers, $\mathbf{C}_L$ and $\mathbf{C}_R$, so that the only translation between them is the baseline distance. The origin of the stereo system is set equal to $\mathbf{C}_L$, leaving the right camera as the only one that has a translation

associated with it. The projection matrices for the two cameras in pixel coordinates become

$$P_L = \begin{bmatrix} f & 0 & c_x & 0 \\ 0 & f & c_y & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \tag{2.9}$$

$$P_R = \begin{bmatrix} f & 0 & c_x & -Bf \\ 0 & f & c_y & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \tag{2.10}$$

The end result of sending the epipoles to infinity is that the image planes are placed in the frontal parallel orientation. The images must undergo a transformation so that the location of each pixel would be found at the coordinates where it would have appeared had the images originally been captured in that orientation. This transformation is performed through a 2D mapping matrix known as a homography matrix. It is not important to know exactly how the homography matrix was found, but just that it rectifies the image so that it is in the frontal parallel position and the epipolar lines coincide. For a more in depth look at the mathmatics involved with both sending the epipoles to infinity and determining the homography, see [3], [5] and [6].

Figure 2.5 shows all of the steps that must occur in order to place two real cameras into the frontal parallel orientation. Notice that there is an additional undistortion step that must occur if the cameras are not pinhole cameras. This will be further discussed in Section 2.4.

Now that the images are in the frontal parallel position, depth information can be calculated from triangulation. All of the triangulation equations can be simplified into a linear system as shown in equation 2.11.

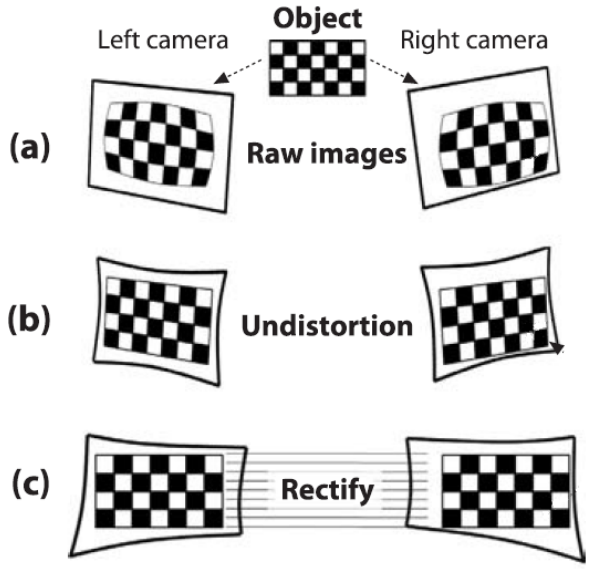


Figure 2.5: Steps that must be taken in order to get images in the frontal parallel orientation. Note that there is an undistortion step in the process which will be discussed in Section 2.4. Image courtesy of Bradski et al. [3] [used under fair use guidelines].

$$\mathbf{X} = Q \begin{bmatrix} x \\ y \\ d \\ 1 \end{bmatrix}$$

$$\mathbf{X} = \frac{B}{d} \begin{bmatrix} 1 & 0 & 0 & -c_x \\ 0 & 1 & 0 & -c_y \\ 0 & 0 & 0 & f \\ 0 & 0 & \frac{1}{B} & 0 \end{bmatrix} \begin{bmatrix} x \\ y \\ d \\ 1 \end{bmatrix} \tag{2.11}$$

where Q is referred to as the reprojection matrix. The reprojection matrix allows us to determine the 3D location of any point that is localized in both images.

2.2.3 Stereo Correspondence

Now that the images have been rectified with their epipolar lines aligned horizontally, features must be matched in between the images so that depth information can be gathered. These matches will be known as dense features and will define a 2D matrix known as a disparity map. A disparity map represents the disparities of individual pixels that have been localized in both images. The goal of stereo vision algorithms is to produce these disparity maps, because they can be used to directly determine the depth of an entire scene.

Many two camera stereo vision algorithms exist; many of these have been tested against what is known as the Middlebury benchmark. People working on computer vision at Middlebury college developed a website that provides 2D imagery along with ground truth disparity maps of various scenes. Stereo algorithms can be compared to the ground truth models in order to quantify how well they performed. With quantifiable data, the algorithms can be compared with one another to see which ones perform the best [7].

The algorithm that was chosen for stereo reconstructions in this project was the OpenCV adaptation of Hirschmuller's semi-global matching [8]. It was chosen because of its relatively good standing in the Middlebury benchmark, as well as the existence of an open source variant in OpenCV. The OpenCV variant is slightly different from the original and will be referred to as semi-global block matching (SGBM) [9]. The following section will discuss the basics of how SGBM works. This will be followed by descriptions of a few other top stereo vision algorithms. These may be good choices for future implementations of this work.

Semi-Global Block Matching

SGBM is a window based stereo vision technique that relies on an aggregated cost function calculated through a multi-directional search. The technique begins by reducing the resolution of the imagery and searching the reduced image for areas(windows) that, in a sense, have similar intensity profiles between the images. Matches are found based on a

technique developed in [10]. These matches then define an initial disparity map estimate for all of the points. The algorithm returns to the larger imagery and begins searching for matches again. If the match is not located where the estimated disparity map believed it to be, a cost is levied on it based on the magnitude of the difference between the locations. This is a surface continuity constraint and is important because, if it were not done, then discontinuities would not be recognized and the matching performance would suffer. There is also a cost placed on the matches based on the strength of the match. SGBM can then accrue a global cost function by moving a search window along different directions to further help improve the final disparity map, as shown in Figure 2.6. The position where the cost is the lowest is considered to be the true match, thereby allowing the creation of a disparity map. This disparity map then undergoes subpixel smoothing by way of quadratic interpolation. This makes the 3D reconstruction look smoother.

The OpenCV version of SGBM, which was just described, has a very similar framework to that described in [8], but there are a few significant differences. [8] uses a probabilistic technique, known as Mutual Information, to determine where matching occurs. SGBM, instead, uses the sub-pixel metric described in [10]. Another difference is that the Mutual Information technique determines matches on a pixel level where the OpenCV variant uses a window of pixels. Lastly, SGBM only searches in five different directions for the best cost, while [8] searches 16 different directions.

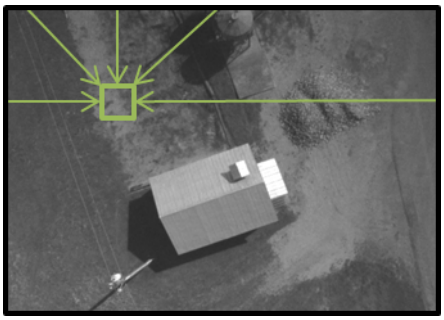


Figure 2.6: Multi-directional search pattern for SGBM. The green arrows represent the directions that the algorithm searches while the green box is a window containing multiple pixels.

Other Stereo Correspondence Techniques

The stereo vision algorithms that will be presented in this section constitute some of the top performers according to the Middlebury benchmark. These are algorithms that would in all likelihood outperform SGBM. The algorithms presented are also largely parallelizable, which make them capable of running on a graphics processing unit(GPU). This would allow them to be run extremely quickly and would be necessary if the variable length stereo boom is to operate in near real-time.

Yang et al [11] created a stereo correspondence technique that relies on color-weighting and belief propagation(BP). It begins by creating an initial correlation volume that is dependent on regions of similar colors within the images. These areas are assumed to be continuous and at roughly the same disparity levels. The algorithm then determines an initial estimate of disparity maps from both left-to-right and right-to-left by a statistical approach known as belief propagation. By searching the images in both directions, it removes some of the mismatches within the disparity maps. All of the pixels in the image are then examined to determine whether or not they are stable, unstable, or occluded based on the confidence with which they are localized between images. This confidence measure comes directly from the creation of both of the previously mentioned disparity maps. The entire process is iterated multiple times with the knowledge gained from the previous trials, which refines and better estimates the disparity maps. This technique is within the top ten stereo vision performers according to Middlebury [7].

The algorithm presented in [12] uses the concept of a cost function that incorporates both absolute differencing and census matching techniques to effectively find correspondences. The absolute differencing portion determines the difference of intensities in RGB colorspace. The difference should be small if it is a close match. The census portion of the code relies on a Hamming window. The Hamming window checks if a pixel is greater in intensity than its predecessor. If it is, it is a assigned the value one, and if it is not, it is assigned the value zero. This creates a type of structure of intensity distributions that can be matched

between images. The absolute differencing portion makes it easier to match areas that have repetitive structures, while the census portion works well on relatively textureless areas. The strength of the one is the weakness of the other, which allows them to work well together. The technique also employs an aggregated cost method that changes the window size for matching based on similar colors, which removes noise and matching ambiguities. This assumes that areas with same colors are at similar disparities. The technique also relies on a multi-directional search similar to the semi-global matching technique [8] to further increase the robustness. Lastly, it refines the disparity map based on a few previous techniques, such as sub-pixel interpolation. This algorithm is currently (as of June 2012) the top performer on Middlebury [7].

2.3 Sparse Feature Matching

Unlike the stereo correspondence methods, which attempt to find similarities between entire images, sparse feature matchers attempt to find highly recognizable regions within an image. These recognizable regions are known as sparse features because they do not encompass the entire image. By finding highly recognizable regions, these feature detectors can help to localize similar regions across multiple images. This ability makes sparse feature points useful for object recognition and image stitching. In this project, they will be used to see how well the epipolar lines have been aligned after the rectification process.

There are many different algorithms that localize sparse feature points. Some, such as the Harris Detector (described in [3]), Features from Accelerated Segment Test(FAST) [13], and Oriented-FAST Rotated BRIEF (ORB) [14], are corner detectors. They search the images for areas that have sharp intensity changes along multiple directions, which is indicative of a corner. The Scale Invariant Feature Transform(SIFT) [15] and Speeded-Up Robust Features(SURF) [16] are two feature point detectors that rely on convolutions of filters over

different scales of the imagery. SIFT is the feature detector that will be used later in this work, and although it is not a critical component, it will be further explained to abate any questions that may have arisen.

SIFT begins by convolving the image with many different Gaussian filters to create new images. These new images appear smoother because the Gaussian filter is a low-pass filter, and thus smooths out the high frequency changes in intensity. The new filtered images are then subtracted from the adjacent Gaussian images as shown in Figure 2.7. This process is known as the difference of Gaussians. Each pixel in the resulting images is then checked to determine which are local maxima and minima with respect to intensity. These locations are candidate feature point locations. The algorithm then ensures that the candidate point is a maximum or minimum with respect to the 8 pixels that surround it, as well as the 9 pixels from that location in the above and below scales. If it is, then it is considered to be a feature point. By ensuring that the feature is unique amongst different size images, it makes SIFT features scale invariant. These scale invariant features are then given a descriptor based on the gradients of the surrounding pixels. This descriptor allows the feature point to be recognizable amongst different images. The gradient directions surrounding the feature points are placed into a histogram. The histogram is shifted until the dominant direction

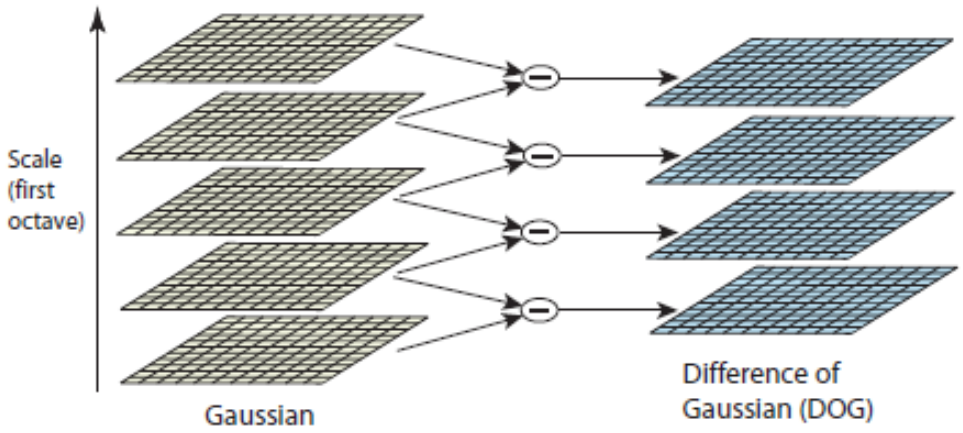


Figure 2.7: Difference of Gaussians used for SIFT feature detection for one scale-space of images. Image courtesy of Lowe [15] [used under fair use guidelines].

is in the first spot. The newly shifted histogram forms the descriptor which distinctively describes the feature point. This final shift makes SIFT features invariant to rotations [15].

2.4 Calibration

This section will present techniques that can be used to calibrate stereo cameras. It will begin with a discussion of the Camera Calibration Toolbox for Matlab, which will be used to gather the calibration results required for this work. The toolbox is an a priori calibration procedure and can not self-correct in middle of use. This will be fine if the boom can be calibrated before every use, but it is impractical in many scenarios. Other techniques have been developed that can self-calibrate and determine the calibration parameters during regular use. Self-calibration would be extremely useful; however, it will not be implemented here. Instead, a couple of self-calibration techniques will be presented to provide a basic understanding as to how they work.

2.4.1 Camera Calibration Toolbox for Matlab

A stereo vision system has many parameters of importance which include the intrinsics, extrinsics, and image distortions. Both the intrinsic and image distortion parameters describe each individual camera, while the camera extrinsics describe the relationship between multiple cameras. The camera intrinsics include parameters such as the focal lengths and principal points. The camera extrinsics are the rotations and translations between cameras. The intrinsic and extrinsic calibrations are necessary for the projective geometry representation of the cameras, while the distortion parameters arise from the fact that real cameras are not actually pinhole cameras. Instead, there is a finite opening known as the aperture associated with them. There are two distortions that need to be described and are known

as radial and tangential distortion. Radial distortion is caused by the shape of the lens, while the tangential distortion is an artifact of the imaging sensor being out of parallel with respect to the lens. These distortions will cause what should be straight lines in an image to appear bowed. Undistortion of the images can be performed by equation 2.12,

$$\begin{bmatrix} x_p \\ y_p \end{bmatrix} = \left(1 + k_1 r^2 + k_2 r^4 + k_3 r^6\right) \begin{bmatrix} x_d \\ y_d \end{bmatrix} + \begin{bmatrix} 2p_1 x_d y_d + p_2 \left(r^2 + 2x_d\right) \\ p_1 \left(r^2 + 2y_d^2\right) + 2p_2 x_d y_d \end{bmatrix} \tag{2.12}$$

where x_p and y_p are the undistorted image coordinates; k_1, k_2, k_3 are the radial distortion terms; p_1 and p_2 are the tangential distortion terms; x_d and y_d are the distorted image coordinates; and r is the magnitude in distorted image coordinates away from the principal point of the camera [1]. It is important to realize that the distortion terms are highly nonlinear. This will create some difficulties for the control algorithm which will be described in Chapter 5.

In order to ascertain all of the camera parameters, the Camera Calibration Toolbox for Matlab was used [1]. The toolbox uses slightly altered versions of the algorithms described in [6] and [17] in order to do this. [17] describes how to determine the intrinsic camera parameters along with the distortion coefficients. [6] provides an algorithm for gathering the extrinsic camera parameters. The Camera Calibration Toolbox for Matlab melds the two papers so that all of the camera parameters can be determined. It requires many images from different vantage points taken looking at a calibration rig with known size properties. The rig must be a checkerboard pattern because the algorithm needs to use the points along the checkerboard in order to find the camera parameters. An image of the rig that was used for the camera calibrations can be seen in Figure 2.8. It is important to note that the calibration images from one camera must undergo rotations with respect to other images from the same camera. Otherwise, if there is only a translation between the cameras, no new information about the camera parameters is gathered and this causes the calibration routine to fail [17]. This calibration routine is used throughout this work.

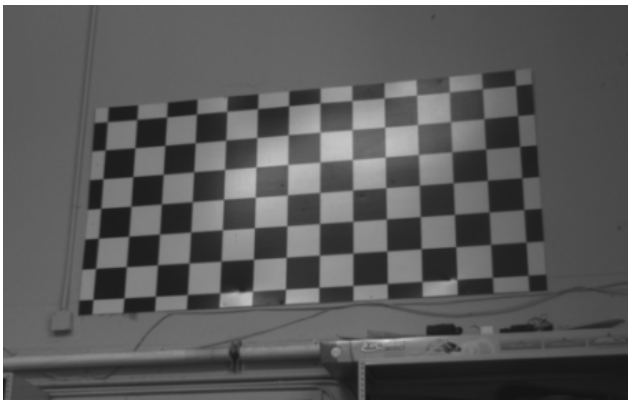


Figure 2.8: Calibration rig used for calibrating the stereo cameras

2.4.2 Self-Calibration Routines

One type of self-calibration technique determines the fundamental matrix from a set of point correspondences. The fundamental matrix forms the relationship between points in two uncalibrated cameras. The relationship is shown in equation 2.13.

$$x_{L,i}^T F x_{R,i} = 0 \tag{2.13}$$

where $x_{L,i}$ and $x_{R,i}$ are the same 3D point as viewed in both images, and F is the fundamental matrix. The fundamental matrix thus describes the epipolar geometry of the system and can be used to rectify the images. Calculating the fundamental matrix requires at least seven corresponding points across the images [2]. These points are oftentimes sparse feature points that have been matched between frames. This technique does not use any knowledge about the camera parameters a priori and determines them based on the point relationships. The seven-point technique is fairly simple to execute, but it does not calculate any terms related to distortion. The distortion parameters of the camera must therefore be determined beforehand.

[18] presents a five-point technique to find the essential matrix between two images. The essential matrix describes the rotations and translations between two calibrated cameras. It is the calibrated variant of the fundamental matrix. Determining the essential matrix

requires that the intrinsic parameters be known a priori, which would require a calibration. A calibration tool, such as the Camera Calibration Toolbox for Matlab, could be used to gather the intrinsic and distortion camera parameters. The essential matrix could then be used to rectify the objects into the frontal parallel position without the fear of problems stemming from lens distortions.

Another, more complicated technique, has been developed in [19]. Their technique relies on the epipolar geometry described by the fundamental matrix as well. However, they have added an extended Kalman filter in order to estimate what the parameters should be, based on previous estimates. This, in combination with a Gauss-Helmert model for optimization, is quite effective at arriving at the true calibration parameters. The optimization routine requires an initial estimate of the camera extrinsics that is accurate to within a few degrees. The initial estimate will allow the model to converge upon the true calibration parameters. SIFT feature points must be matched between the image pair so that the fundamental matrix can be computed. Improperly matched features will strongly influence the calculation of the fundamental matrix. [19] use a Least Median of Squares random sampling technique to remove these mismatches. The end result is a procedure that converges to the actual intrinsic and extrinsic calibration parameters. Note that this does not find the distortion parameters of the lenses, which means that they would need to be determined beforehand.

2.5 Imaging Hardware

For this project, two greyscale IEEE 1394.b (FireWire) Sony XCD-U100 cameras were used. The cameras capture individual images, not a video stream, with a resolution of 1600 by 1200 pixels. However, to reduce computation time and complexity, the resolution of the captured images was reduced to 800 by 600. The Sony cameras were chosen because of there availability. The cameras were used in conjunction with Kowa LM8JCM lenses with

a nominal focal length of 8 mm. This relatively small focal length allows imaging of a large field of view, which would be important in this application. The combination of both the cameras and lenses provide approximately 42.6° horizontal and 32.5° vertical fields of view.

The cameras were connected in a daisy-chain configuration. This means that the two cameras are connected together by a FireWire cable and then one of the cameras has an extension that plugs into a laptop. The extension running from the laptop to the cameras provides them with power and is also used to send commands in order to both configure and trigger the cameras. The cameras are triggered by way of a software signal. A signal is broadcast to all of the cameras, but only the camera with the corresponding address gathers the image. To trigger both cameras, two signals must be sent and received. This implies that the cameras can not be triggered precisely at the same time. This delay can be detrimental to stereo vision, especially if the stereo boom is being moved. Objects in the scene may have moved within that amount of delay time, which would hinder the depth estimates. It is uncertain how much of a delay exists, but it has arisen in previous tests with the cameras. This timing problem can be remedied by using hardware triggering. A hardware trigger would change the voltage on an input pin instead of sending a software signal, thereby allowing the image capture to occur simultaneously. However, that is outside of the scope of this project and therefore relegates the boom to a laboratory test rig.

2.6 Previous Works

There are many different stereo vision systems that have been developed. This section will describe a few works dealing with stereo vision systems that are similar to this variable baseline stereo boom.

Many stereo vision systems fit into the category of active stereo vision. Active stereo vision refers to systems that have the capability of adjusting the viewpoint of their cameras.

These systems oftentimes use cameras that can change their relative yaw, or vergence, angle. This is useful for allowing the system to focus on important regions within the field of view. In [20], [21], and [22], techniques have been developed to perform stereo reconstruction for cameras that have the capability of adjusting the vergence angle of the cameras. They use a mechanism capable of individually adjusting the yaw and tilt angles of the cameras. Similar rigs has been designed and constructed by [21] and [23].

In [24] and [25], they have found uses for stereo booms in order to aid in minimally invasive surgical procedures. A 5 degree-of-freedom stereo boom that could fit on a type of laparoscope is proposed and designed in [24]. It would be capable of adjusting its baseline, yaw of each camera independently, and a combined tilt of the cameras. Stereo vision could aid surgery by providing the surgeon with depth information about where surgical implement is located as well as where the tool needs to be. There is no indication that they have physically constructed the stereo vision system, and it instead appears to be strictly conceptual. A stereo vision system that fits on a laparoscope was constructed and tested in [25]. Their system has the capability of tracking the tip of a tool based on color and providing depth information about it. It does not have the capability of changing its baseline, but they mentioned future work to allow the system to automatically adjust its baseline based on the tool location.

Others have developed a stereo vision system that uses four cameras for vehicular navigation [26]. Two cameras that have relatively long focal lengths are separated by a distance of 22 cm. The other two cameras have a shorter focal length and are located at a baseline within the other two cameras. This system is useful for gathering depth information for both near and far distances at the same time. The interior cameras gather a lot of information about the terrain that is close while the outer ones gather more information about the objects that are further away. One drawback of a system such as this is that it requires four cameras instead of two. This could be financially costly.

Another stereo vision system incorporates a variable baseline in order to gather depth

information of an environment with a constant accuracy [27]. The entire map therefore ensures that the objects within the scene have a constant angle for triangulation, which is important for stereo correspondence. The algorithm begins by gathering images from different baselines. It then utilizes a plane sweeping technique to find matches between the images at different focal lengths and baselines. These matches ensure that the desired error and angle on the objects are maintained throughout the 3D reconstruction. Problems arise with this system because of the need for images gathered from different baselines. The images from multiple baselines can be gathered by either utilizing multiple cameras, or having cameras that can adjust the distance between them. If multiple cameras are used, the system becomes financially costly. Adjusting the baseline of the cameras also causes a problem because of the time that will occur between image capture. Everything in the scene must therefore be stationary with respect to the cameras. This is not practical for the applications of this thesis.

A variable baseline system has been developed in [28]. This system has the ability to gather depth information pertaining to an object of interest. It then automatically adjusts the width of the baseline so that the object is placed on the edge of the field of view. It is very similar to the work presented in this thesis, but it does not ensure that the stereo correspondence can be found.

Chapter 3

Boom Design

In this chapter, the mechanical design of the variable baseline stereo boom will be presented. It will begin with a calibration sensitivity analysis. This analysis will provide insight into the required manufacturing tolerances, that should ensure that good stereo reconstructions can be created from one calibration over the entire length of the boom. If only one calibration file is required, it would make the system simpler. The results of the analysis will be the driving force for the mechanical design process. The actual design will be presented in the following section. The chapter will conclude with a finite element analysis describing the structural integrity of the stereo boom.

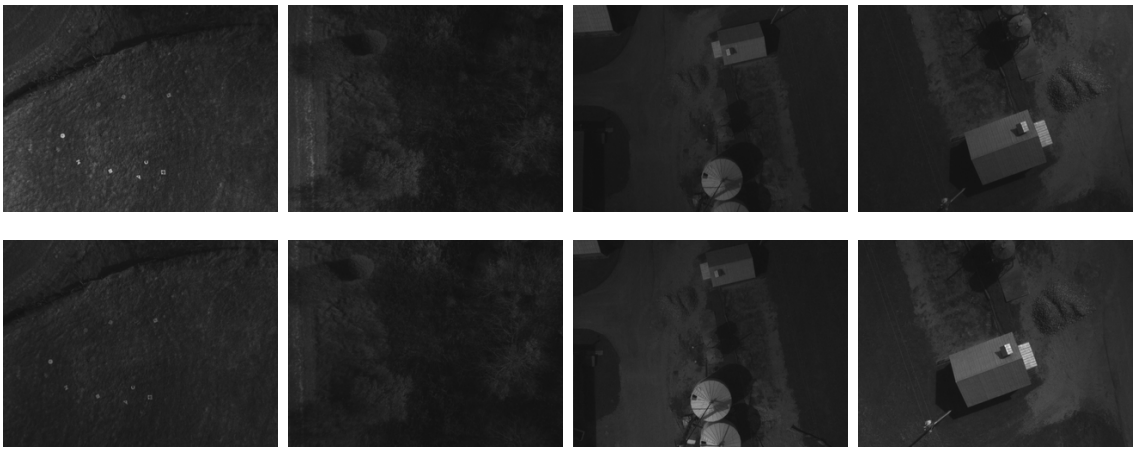
3.1 Calibration Sensitivity Analysis

This section will attempt to illustrate the effects that errors in the calibration can have on stereo reconstructions. It is especially important to analyze the allowable error associated with the calibration results for this particular system. This is due to the fact that there will be moving parts and the entire rig could be moving, which may cause relative motion between the cameras. Any relative motion will introduce errors in both the three-dimensional position

of points and the vertical offset between the rectified images. Many stereo correspondence algorithms search for matches along the horizontal epipolar lines. If there is a vertical offset, the algorithm may not be able to distinguish the same locations in both images, resulting in poor stereo reconstructions. Therefore, it is important to understand the influence that the calibration results have on stereo reconstructions.

3.1.1 Vertical Offset Error

As was previously explained, the vertical offset error is very important for stereo reconstructions. An analysis was performed using aerial imagery to determine how far the vertical alignments could deviate before the stereo reconstructions deteriorated. The imagery had been gathered from a different nadir-oriented Unmanned Systems Lab(USL) calibrated stereo boom flying above various terrains and over the course of many months. Most of the imagery was gathered at an altitude of 40 meters, but some was from 60 meters. Imagery from both altitudes were used in order to simulate objects that are different distances from the camera. Some of the imagery can be seen in Figure 3.1.



(a) (b) (c) (d)

Figure 3.1: Aerial imagery over different terrains with the left image on top and the right image on the bottom: a) is a relatively flat section of land, b) is a group of trees and shrubbery, c) are buildings from 60 meters, and d) is a building and silo.

The analysis was completed by running the SGBM algorithm on the image pairs at different vertical offsets [5]. How well the stereo algorithm performed was based on the number of points that were created. The larger the number of points, the more locations that SGBM could find in both images. The results were then normalized by dividing the number of points at each vertical offset by the largest number of points that were found for a given image, as shown in Figure 3.2. This determined the percentage of pixels that were found in comparison to the best case scenario with the vertical offset at zero. From the figure, it is evident that the vertical offset plays a critical role in the effectiveness of the SGBM algorithm. The number of points is reduced in a nearly quadratic fashion as the vertical offset is increased.

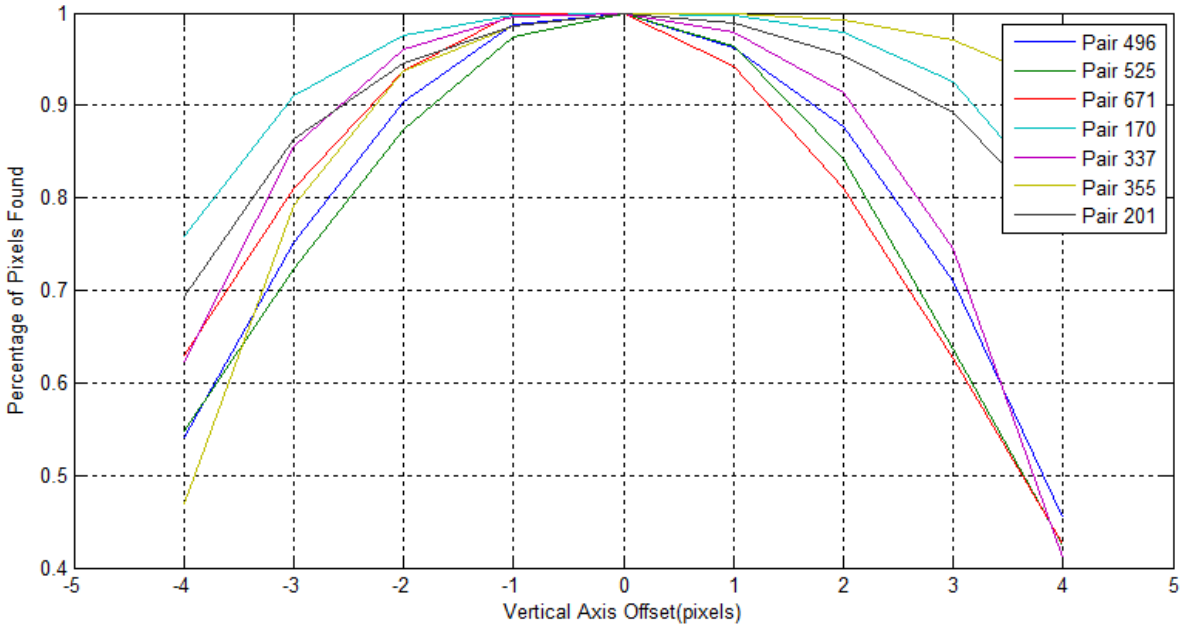
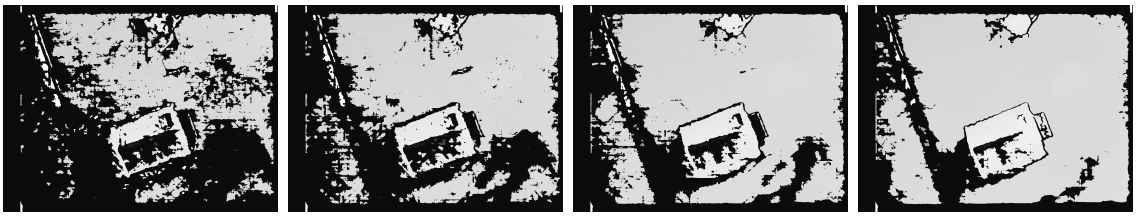


Figure 3.2: Effects of different vertical offsets on the SGBM algorithm

The performance of SGBM for all of the images appear to be good up to around 2 pixels of vertical offset irregardless of what the pair was viewing. All of the reconstructions retain greater than 80 % of their largest number of points when the offset is less than or equal to 2 pixels. Based on these results, it is assumed that the vertical offset induced by calibration

errors can be no more than 2 pixels. This value becomes important to the analysis which is performed in the following section. The disparity maps from Figure 3.1d are shown in Figure 3.3 for various amounts of vertical offset. The black regions in the disparity maps are representative of holes in the data set. This figure illustrates that the better the vertical alignment between an image pair, the greater the number of seemingly correctly localized points there are. This lends credence to the analysis procedure by showing that the number of points is a sound metric for determining the effectiveness of the system.



(a) (b) (c) (d)

Figure 3.3: Disparity maps shown at different vertical offsets in pixels: a) 4, b) 3, c) 2, and d) is 0.

It is important to realize that these results only describe the degradation of SGBM due to alignment errors for the specific SGBM parameters that were used. Other stereo vision algorithms may have difficulties at different values. This means that the stereo boom was designed with SGBM in mind.

3.1.2 Calibration Errors

Now that we know how precise the vertical resolution must be, a sensitivity analysis can be preformed in order to determine the allowable calibration errors. The analysis will determine the maximum roll, pitch, and yaw error limits necessary to keep the vertical alignment within 2 pixels along with maintaining a low distance error. In this work, roll is the rotation about the Z-axis, pitch is rotation about the X-axis, and yaw is the rotation about the Y-axis. Figure 3.4 shows the camera coordinates along with the world coordinates.

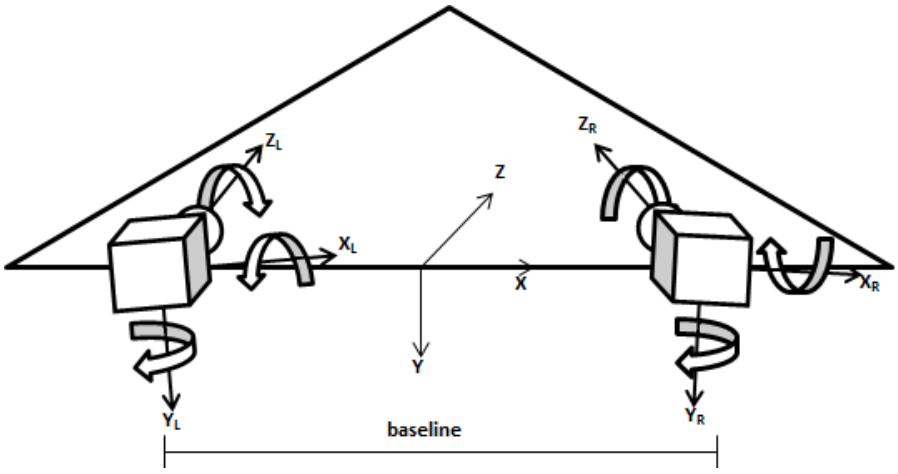


Figure 3.4: Camera coordinates and rotations

Thao Dang et al. [19] have performed an analysis where they derive the theoretical sensitivities for all of the parameters that affect stereo vision, which includes the roll, pitch, yaw, baseline, image center, and focal length errors. The analysis presented in this section will utilize their derivations for errors associated with the roll, pitch, yaw, and baseline. The focal length and image center errors are intrinsic parameters and are assumed to be constant, and are not analyzed since they are not important for mechanical design purposes.

The analysis presented in [19] assumes that the cameras have been calibrated such that the cameras are parallel and are perpendicular to the stereo baseline. This is a valid assumption because the cameras should not deviate greatly from the initial calibration. Another assumption that is made is that the x and y focal lengths are the same. This is a valid assumption because the rectification process will make the focal lengths the same. To obtain the sensitivity equations, the projective pinhole camera matrix is differentiated with respect to each parameter and the first term of the Taylor series expansion is used, which makes the analysis valid for small angles. This is an accurate approach so long as the rotations between cameras are small.

The sensitivity equations rely on normalized camera coordinates, which means that the pixel locations within an image are referenced assuming that the focal length is 1. The nor-

malized camera coordinates can be found by multiplying the inverse of the camera intrinsic matrix by the pixel coordinate locations which leads to

$$\tilde{\mathbf{x}}_{\mathbf{L}} = \begin{bmatrix} \frac{x_L - c_x}{f} \\ \frac{y_L - c_y}{f} \\ 1 \end{bmatrix} \tag{3.1}$$

where $\tilde{\mathbf{x}}_{\mathbf{L}}$ is the normalized coordinates $[\tilde{x}_L\ \tilde{y}_L\ 1]^T$, $[x_L\ y_L\ 1]^T$ are the locations in pixel coordinates, f is the focal length in pixels, and $[c_x\ c_y]^T$ is the principal point in pixel coordinates.

The following table shows the equations that were derived in [19]. This section will not attempt to derive the equations, but will instead only present them. See [19] for the detailed derivations. Each equation was derived assuming that there was no error in the other parameters. Therefore, these equations do not address coupling between themselves, which may indeed play a role.

Table 3.1: Sensitivity Equations for Calibration Errors

Error Parameter	**Vertical Offset**	**3D Reconstruction Error**
Yaw (Ψ)	$\Delta\Psi_L \approx \frac{\Delta y}{f\tilde{x}_L\tilde{y}_L}$	$\Delta\Psi \approx \frac{-B\Delta Z}{Z^2(1+\tilde{x}_L)^2}$
Pitch (Φ)	$\Delta\Phi_L \approx \frac{-\Delta y}{f(1+\tilde{y}_L^2)}$	$\Delta\Phi_L \approx \frac{B\Delta Z}{Z^2\tilde{x}_L\tilde{y}_L}$
Roll (Θ)	$\Delta\Theta_L \approx \frac{\Delta y}{x_L - c_x}$	$\Delta\Theta_L \approx \frac{B\Delta Z}{Z^2\tilde{y}_L}$
Baseline (B)	$\Delta y \approx 0$	$\Delta B \approx \frac{B\Delta Z}{Z}$

B is the baseline between cameras, Z is the distance to the cameras, ΔZ is the distance error to the point of interest, and Δy is the vertical offset error. These parameters must be chosen in order to solve for the allowable roll, pitch, yaw, and baseline errors. The allowable vertical offset error was set at 2 pixels. This follows the analysis that was performed in the previous section and should ensure images that can be reconstructed well. The distance to the object, Z, was set to 394 inches ($\approx$ 10 m). This distance was chosen for a few reasons. One reason is that a typical excavator has a maximum reach of around 472 inches($\approx$ 12

m) [29]. So, 394 inches would cover most of the operating range of the tool. Another reason was that at 394 meters objects become smaller and more difficult to detect with the current lenses. The error in 3D reconstruction was set equal to 3.94 inches($\approx$ 0.10 m). This was chosen to ensure that the error is at most $1/100^{th}$ of the total distance. The baseline was set at 27 inches($\approx$ 0.685 m). The rationale behind the baseline width will be discussed in the following section. The analysis was performed using 800x600 resolution imagery captured with an 8 mm lenses, which leads to a focal length of around 943 pixels. The focal length was determined in terms of pixels through calibrating the cameras with the Camera Calibration Toolbox for Matlab [1]. Table 3.2 shows the maximum allowable errors for these different parameters.

Table 3.2: Maximum allowable calibration errors for 800x600 resolution imagery, a distance 394 inch with a 3.94 inch allowable 3D location error, a focal length of 943 pixels, a vertical offset error of 2 pixels, and a baseline of 27 inches.

Error Parameter	**Vertical Offset (Center)**	**Vertical Offset (Edge)**	**3D Error (Center)**	**3D Error (Edge)**
Yaw (Ψ) (deg)	∞	1.332	0.039	0.024
Pitch (Φ) (deg)	0.122	0.259	∞	0.430
Roll (Θ) (deg)	0.382	0.229	∞	0.125
Baseline (b) (in)	∞	∞	0.270	0.270

Table 3.3: Allowable movement of cameras

Error Parameter	**Max Angle (deg)**	**Max Movement (in)**
Yaw (Ψ)	0.024	0.001
Pitch (Φ)	0.122	0.005
Roll (Θ)	0.125	0.005

The maximum allowable errors in the calibration are extremely small for each parameter. Table 3.3 puts this into perspective by showing how much the camera can move. This is the amount that the end of the lens can be angled with respect to its original position. This means that the design must be extremely rigid to ensure that the stereo vision algorithm works with just one calibration. Note that the maximum angle for pitch would be halved if full resolution imagery was used.

3.2 Mechanical Design

This section will present the mechanical design of the stereo boom and will attempt to justify decisions made throughout the design process. This section will begin by discussing the specifications that governed the design of the boom, followed by a description of the actual design. Analysis of the boom will be performed by way of finite element analysis (FEA) in order to provide insight into the strength of the design.

3.2.1 Specifications

There are a large number of design considerations for a variable length stereo boom. These include the minimum and maximum baseline, the overall size of the structure, the weight of the boom, and where to place the cameras. One of the goals of this project was to create a boom that had a large variational baseline while still being compact and lightweight. The maximum baseline needed to be large enough to gather accurate distance measurements of objects nearly 394 inches away and yet small enough to fit through a large doorway. The large baseline is important for construction type applications, as was previously mentioned, but could be a hindrance if a ground vehicle needed to traverse indoors. By incorporating a large variational baseline, objects that are very near to the boom could still be seen. Also, the weight is of some concern. It is important to keep the boom as lightweight as possible because otherwise it would become unwieldy and hard to move. The design should be kept beneath 10 lbs. Above that weight, it may become difficult for a ground robot to handle.

In order to determine the most suitable baseline distances, the maximum possible depth resolution was analyzed. A relationship for parallel cameras was developed in Chapter 2 and is shown again in equation 3.2,

$$\Delta Z = \frac{Z^2}{fB}\Delta d \tag{3.2}$$

where ΔZ is the resolution, Z is the distance to the object, f is the focal length in pixels, B is the baseline, and Δd is the smallest disparity possible in pixels.

An analysis was performed through equation 3.2 for a variety of baselines, focal lengths, and distances. Δd is set to 1 pixel in order to determine the greatest possible resolution. Two focal lengths were used in different trials for both full resolution imagery, 1908 pixels, and half resolution imagery, 943 pixels. These focal lengths were determined through calibration on full-size 1600x1200 resolution, and half-size 800x600 resolution imagery collected from the Sony XCDU100 cameras. The baselines of the analysis were varied between 4.7 and 27.0 inches. The largest baseline was chosen to be 27.0 inches because this would be able to fit through a large doorway. The doorway of interest was 33.5 inches wide. 4.7 inches was the size of the smallest baseline and was chosen because it is close to the width between human eyes. The distance to the points was varied from 8 to 720 inches.

For each of the different combinations, the depth resolution was found along with the amount of overlap in images. The overlap is important because stereo triangulation requires the same points to be localized in both images. If there is insufficient overlap between images, stereo vision will only produce a small amount of 3D information. It could also require more computation time because the disparity increases as depth decreases, which requires the computer to search for matching points over a larger range. The stereo vision process will become exponentially longer. The computational issue could be resolved by adaptively changing the initial position that the algorithm looks for matches. However, this will make the algorithm more difficult to tune for best results, and it would be easier to get large amounts of overlap. The fractional overlap can be calculated by equation 3.3 assuming that the cameras are frontal parallel.

$$\text{Overlap} = \frac{2Z\tan\frac{\theta}{2} - B}{2Z\tan\frac{\theta}{2}} \tag{3.3}$$

where θ is the field of view, which is 42.6° for the camera and lens combination. Tables 3.4

Table 3.4: Depth Resolution from Stereo Vision for 4.7 inch Baseline

Distance(in)	**Fractional Overlap**	**Max Accuracy Half Resolution (in)**	**Max Accuracy Full Resolution (in)**
16	0.623	0.06	0.03
120	0.950	3.25	1.61
240	0.975	13.00	6.42
394	0.985	35.03	17.31
540	0.989	65.79	32.52
720	0.992	116.96	57.81

Table 3.5: Depth Resolution from Stereo Vision for 27 inch Baseline

Distance(in)	**Fractional Overlap**	**Max Accuracy Half Resolution (in)**	**Max Accuracy Full Resolution (in)**
16	0.000	—	—
120	0.711	0.57	0.28
240	0.856	2.26	1.12
394	0.912	6.10	3.01
540	0.936	11.45	5.66
720	0.952	20.36	10.06

and 3.5 show some of the results from the analysis and illustrates the importance of having a large variational baseline.

The results of the analysis indicate that a baseline of 27 inches produces outputs which are 5.7 times more accurate than those from a baseline of 4.7 inches. This is a very substantial improvement in resolution. It is also important to realize that as the cameras approach the object of interest, the amount of overlap between images decreases. The lower the overlap, the more centered the object of interest must be. If the baseline was kept at 27 inches, the object would need to be 69.25 inches away to maintain an overlap of 50%. The 4.7 inch baseline requires a distance of 12.05 inches for the same amount of overlap. Realizing the importance of both the baseline and overlap, it is clearly advantageous to have a boom with a large variational baseline. Table 3.6 shows the requirements for the design.

Table 3.6: Requirements of the Boom

Specification	**Range**
Min baseline (in)	≈ 4.7
Max baseline (in)	≈ 27.0
Weight (lbs)	< 10

3.2.2 Boom Design

Now that the requirements have been defined both in terms of calibration error allowances and the overall size of the stereo boom, the design process can begin. The first step is to decide what kind of structure the cameras should be enclosed in. The cameras are relatively long, which made the task somewhat difficult. They are $2\frac{3}{8}$ inches long without the cabling. The cabling adds a minimum of $2\frac{1}{8}$ inches. This makes the cameras effectively $4\frac{1}{2}$ inches long without the lenses attached.

There are a few potential solutions to this problem. One option is to make cases for the cameras to rest in. This would provide ample protection for the cameras themselves. However, it leaves the possibility of something hitting the camera cases and causing the cameras to become misaligned. Encasing the cameras would also cause issues with weight. There would be a large amount of additional material that would provide no structural benefits, and thus require more material to strengthen the overall structure. Another possible option is to use tubing. Tubing would be able to provide protection for the cameras while also adding structural integrity to the design. However, tubing large enough to fit the entire cameras would become heavy and difficult to handle.

In the end, tubing was chosen as the base structure because it has a large strength to weight ratio. The idea was to have the cameras inside the tubing while the lenses and cabling were outside. This in turn would keep the weight of the structure to a minimum while still providing some protection to the cameras. Circular tubing, as opposed to square tubing, was chosen because of its strength characteristics as well as size availabilities. Square tubing was only available in large thicknesses and would provide a lot of strength to the rig, but

would be very heavy.

The next step was to choose the tubing material. There was tubing available in Aluminum, Steel, and PVC. PVC was almost immediately discounted because of its very low modulus of elasticity. The low modulus elasticity meant that the structure would bend more under the same loading as given by Hooke's law. This would make it difficult to keep the structure within the allowable calibration tolerances. Aluminum and steel both have much higher modulus of elasticities. The question then became weight. Aluminum has a density of around 0.098 $\frac{\text{lbs}}{\text{in}^3}$ while steel has a density of nearly 0.284 $\frac{\text{lbs}}{\text{in}^3}$ [30]. The density of steel is nearly three times that of aluminum. This means that the same structure would be that amount heavier. Of course, the tube thickness could be reduced to maintain the strength and reduce weight. However, the loading on the structure is expected to be minuscule, which would mean that the tube thickness could also be small. Tubing is readily available in only certain sizes, and therefore aluminum was chosen because it is lightweight and still fairly strong. The dimensions of the tubing were not determined at this time because of its dependence on all of the components that will be placed within the tube.

The next concern pertained to moving the cameras. The cameras needed to be able to traverse the entire range of the baseline. This would require a linear bearing for the cameras to rest on. The bearing must be long enough, must allow the cameras to move relatively easily, and must constrain the cameras from moving in any direction except along the bearing. To meet these requirements, low-profile Igus NK-02-40 linear bearings with no floating bearing directions were chosen. These bearings are comprised of two parts, an aluminum rail and a movable plastic bearing as shown in Figure 3.5. The plastic bearings fit within the aluminum rail and slide along the length of the aluminum rail. Each camera will be mounted on an independent plastic bearings so that they could move separately. These bearings have no floating bearing directions, which simply means that the bearing movements are restricted to solely along the rail. This is important for reducing the amount of angular deflections that the cameras experience. The low-profile bearing was chosen to reduce the space that the rail must occupy, while simultaneously increasing the pitch stability. The

pitch stability increases because of the very wide horizontal base making it more resistant to pitch.

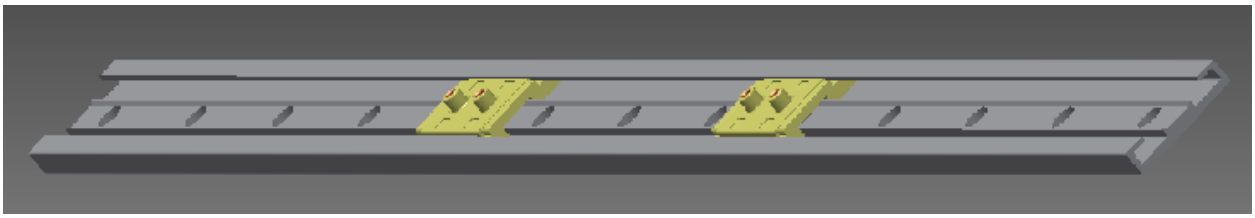


Figure 3.5: Aluminum rail with the two independent NK-02-40 Igus Bearings

At this point, the cameras can move freely along the length of the baseline, but there still is not a mechanism to precisely position them. This requires some sort of actuation, whether it be from a motor or a linear actuator. A linear actuator would be simplest because it could precisely position the cameras without any other mechanical system. However, linear actuators are bulky and tend not to have enough travel. They also introduce problems because they are oftentimes nearly twice as long fully extended, as when they are fully contracted. This would make it difficult to achieve both very short, and very long baselines simultaneously. Motors, on the other hand, are rotary devices. They do not have any problems with travel because they can continuously spin, but they would require an additional mechanism to transfer rotary movement into a linear displacement. The decision was made to use motors as actuators because of their essentially infinite placement capabilities, as well as their size.

It then became important to determine the number of motors that would be used. Two motors would allow the cameras to move independently. This would be useful for situations when the object of interest is near the edge of one of the images. It would allow the one camera to move inwards, while the other one stays stationary. This would be important for situations where the stereo boom is rigidly mounted on an unmovable object, such as a building. However, the applications that have been discussed will have the capability to rotate the stereo boom by either using a rotatable platform, or by moving the vehicle in order to change the viewpoint. This would negate the advantage of using two motors. Therefore,

only one motor will be necessary. The motor will be a stepper motor because it will allow infinite motion and can precisely move a discrete number of steps, thus allowing the cameras to be positioned without any sort of positional measurement of the current camera location.

A gear, which will be attached to the shaft of the motor, will drive two gearing racks that will be mounted to the cameras. The gearing racks will be located on either side of the gear in order to move both cameras at the same time. One camera will be pulled, while the other is being pushed. An illustration of the mechanism is shown in Figure 3.6. The free ends of the racks will move atop of a middle support and will slide past the other cameras allowing them to move over the entire baseline. To prevent the racks from being forced away from the gear as it turns, two rack positioners are placed on the edge of the rack opposing the gear. These will keep the rack in contact with the gear at all times and will ensure that no gear slipping occurs. If the gear slips, it would become difficult to position the cameras without some sort of feedback. This would be very detrimental for stereo triangulation. At this time, the tubing size was finalized. The boom would be constructed from 3.5 inch outer diameter 6061-T6 aluminum tubing with a wall thickness of 0.069 inches.

The racks that were chosen are 20° pressure angle module 1 nylon racks and were used because of their lightweight characteristics. The gear that was chosen is a 0.787 inch (2.0 cm) diameter 0.197 inch(0.5 cm) bore size aluminum gear with a locking screw. The outer diameter was chosen to ensure that a small motor had enough torque to move the cameras, while keeping the discretization of movements much smaller than the allowable baseline error, and yet large enough to be relatively quick. Using the equation for arc length, as shown in equation 3.4, and a typical discretization of a stepper motor, 1.8 °, the distance per step was calculated to be 0.012 inches. This value only represents half of the change in baseline distance because the cameras move simultaneously. The total distance per step then becomes 0.024 inches, which is 10.9 times smaller than the allowable 0.270 inch baseline error as shown in Table 3.2.

$$s = r\theta \tag{3.4}$$

With the actuation mechanism designed, the required amount of torque that the stepper motor needs to move the cameras can be calculated. The motor torque can be related to a linear force because the radius of the gear is known. This linear force is the force that is required to overcome the friction created by the bearings. The force to overcome static friction was calculated by assuming the boom is on a level surface and becomes

$$F_{\text{fr}} = \mu F_{\text{cam1}} + \mu F_{\text{cam2}} \tag{3.5}$$

where μ is the coefficient of friction and F_{cam1} and F_{cam2} are the weights of the cameras. The coefficient of friction for the plastic that comprises the bearings when in contact with aluminum was 0.12, as provided by Igus [31]. The weight for each camera was estimated to be a pound. Each camera weighs only 0.49 lbs a piece. However, there are additional components that will add mass to the system. These include the mounts that hold the cameras, the bearings that the camera mounts rest on, and the weight of the rack. The estimated force of friction was found to be 0.24 lbs. Since there are a lot of parameters that

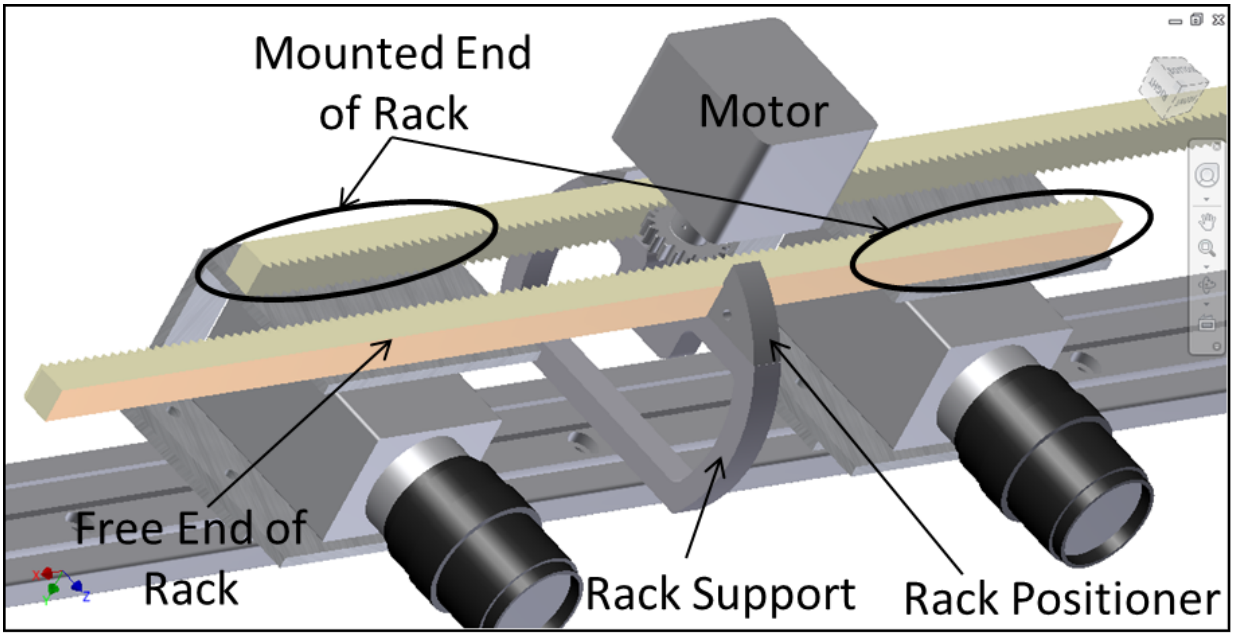


Figure 3.6: Mechanism for camera actuation

are unknown, this force was multiplied by a safety factor of 1.3 leading to 0.31 lbs of resisting force, which equates to 0.25 in-lbs of torque. The motor must be able to output at least that much torque. A Sanyo Denki Model 103H546-0440 stepper motor was chosen because it has a maximum torque output of 1.3 in-lbs and has a step angle of 1.8°.

This allowed the design to be finalized and can be seen in Figures 3.7 and 3.8. Figure 3.7 mentions a few components that have not been introduced thus far as well as showing the final machined boom. The components which are shown more clearly there are the motor mount, the camera mount, and the limit switch mount.

The motor mount is comprised of three parts. They include a flat plate of aluminum and two aluminum side supports that have flat tops and curved underbellies. The flat plate holds the stepper motor in place. The motor had mounting screw holes on the bottom, and therefore required some kind of flat mount to hold them. The two shaped pieces of aluminum are there to attach the plate to the boom structure. They secure the motor mount by way of screw holes through the tubing.

The camera mount is constructed from two parts. The main structure is a $\frac{1}{8}$ inch thick 2x2 inch aluminum C-channel. This piece has a large slot in the vertical section of the channel to allow the camera cabling to escape out the rear of the tube. The C-channel also has many

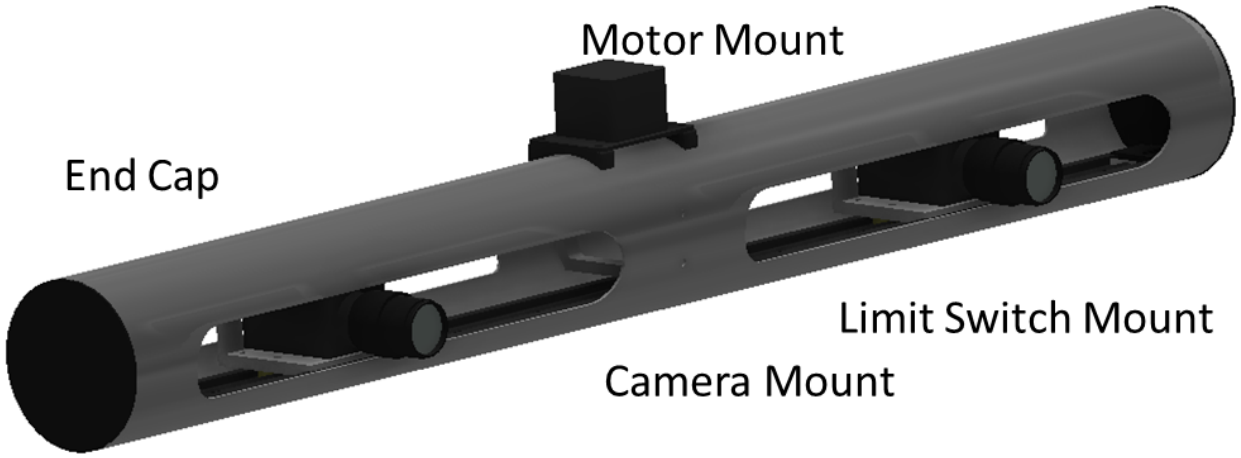


Figure 3.7: Final boom design

holes in the bottom. Two of the holes are there in order to place the camera mounts on the bearings. The bearings have two cylindrical protrusions that fit tightly into the these holes, but epoxy was used as well in order to secure them. These holes merely attach the C-channel to the bearing and do not permanently affix the cameras to the mount, which allows them to be changed if need be. The second part of the mount is a flat aluminum plate that attaches to the camera and is then bolted to the C-channel. This was necessary for this size of C-channel because the cameras are longer than the C-channel and only two of four of the camera mounting holes could be attached to it. Larger C-channels could have been purchased, but the 2x2 inch version was readily available. The mounting plate feature also allows different cameras to be incorporated so long as the cameras can fit within the boom. If different cameras were desired, a new mounting plate must be created to hold the the cameras in place. The only restriction of this design is that the cameras have mounting holes on the bottom side.

The other mount, that was referenced in Figure 3.7, was the limit switch mount and can be more easily seen in Figure 3.9. The limit switch is a necessity since the cameras will be controlled without any feedback. Every time the system is turned off, the global positions of the cameras are lost, making it impossible to accurately know their baseline. The limit switch will remedy the problem by allowing the system to reset itself every time that it is powered on. At startup, the cameras will be driven outward until the limit switch is triggered. The triggering of the limit switch will stop the cameras, and make that position the new zero position of the cameras in the global reference frame. A limit switch is specifically designed for just such a task. They are triggered by a consistent amount of force and will make the cameras have the same zero. Limit switches are used in applications such as 3D printers and CAD machines allowing them to re-zero themselves. The mount for the limit switch is comprised of 3 interlocking pieces of ABS plastic. The first is the end cap, and it fits tightly into the end of the boom. It has a slot cut into it in order to hold the mounting platform. The mounting platform is shaped in such a way that it fits snugly within that slot. There is also a slot in that platform as well, so that another support can be added. All three

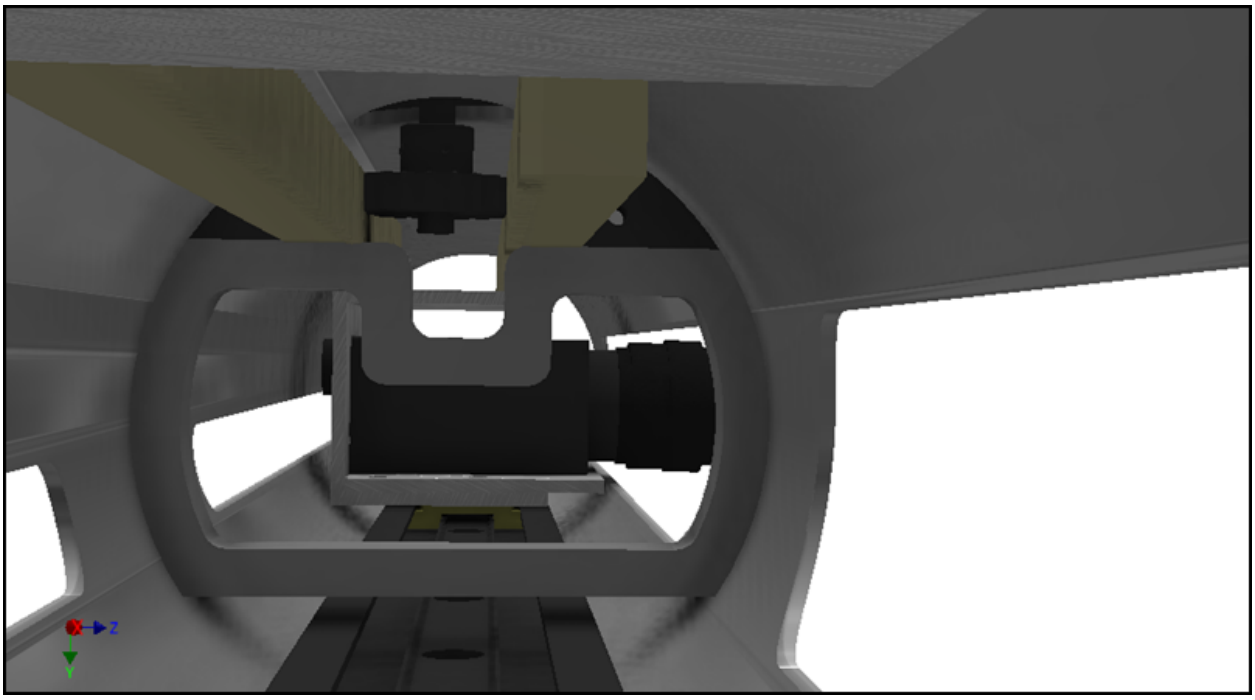


Figure 3.8: Final design from inside the boom

components can be seen in Figure 3.9.

The remaining parts, that can not be seen in the images, are the rail supports. There are five rail supports that are spaced equally along the length of the rail and are aligned with the holes in the rail. The supports have a half moon shape on the bottom and a flat top, and are approximately $\frac{3}{8}$ of an inch wide. The rail rests upon the flat portion of the supports while the half moon side rests upon the inner face of the tubing. This ensures that the rail is not supported directly by the tubing. The supports have holes drilled vertically through them. The rail, rail supports, and the tubing are then all bolted together.

The outer tubing required a lot of machining so that all of the components could fit within. This included two large slots in the front of the boom. These slots allow the lenses to protrude outside of the boom. They are 12.5 inches long, 1.5 inches high, and are 3.5 inches apart. There are also slots in the back of the tubing to allow the cabling from the cameras to exit the boom. They are the same length as the slots in the front, but are only

Figure 3.9: Limit switch mount and end cap

around half an inch high. These two sets of slots allow the cameras to move the entire length of the baseline. The tube was cut to 32 inches long to ensure that there is still some support on the outside of the structure, while allowing it to fit through the door frame. There are also a lot of holes on the boom. Most of the holes are bolt holes and are fairly small, but one hole on the top of the boom is large enough to fit the gear through. The gear hole and slots can be seen in Figure 3.8.

3.2.3 Finite Element Analysis

A Finite Element Analysis(FEA) of the design will be performed in this section. The analysis will attempt to quantify the structural integrity of the stereo boom. This will include a structural and thermal loading analysis to illustrate the deflections under a variety of loading conditions, but will exclude a modal analysis due to its heavy dependence on the mount. It is very important to perform the FEA because the allowable calibration errors are diminutive, and so the slightest deflections could introduce problems to the stereo reconstructions. To reiterate, this FEA is trying to illustrate whether or not one stereo calibration file can be used for stereo reconstructions at every possible baseline. All of the FEA analysis will be performed using Abaqus CAE 6.10-2.

Model

In order to perform an FEA of a structure, it is often times useful to simplify the model. This includes removing non-critical components and holes. Simplification allows the FEA to run more quickly because larger mesh sizes can be used. The important aspect of simplification is to ensure that critical load bearing members are not removed. This requires an intuition into where the loads will propagate through the structure. Features, such as small holes, can often be removed from the analysis as well, but can only be removed if they will not see very large stresses. If there is sufficient stress near the holes, the holes will act as stress concentrators and increase the stresses at that point, thereby requiring that they remain in the model. The smaller the hole, the more concentrated the stress, which makes it important to ensure that they will not see much stress.

All of the small bolt holes were removed from this analysis because the amount of loading that the boom will experience is rather small. It is less than three pounds of total load. This loading comes from the cameras and motor, along with their associated mounts. The gearing hole on top of the structure and the long slots were the only holes that were kept in the model. The stereo boom also has many non-critical components that were removed from the model. These include the camera mounts, cameras, bearings, motor mounts, and so on. All that remained was the outer tubing, the rail, the middle support structure, and the rail supports. These parts were not removed because they were believed to play a critical role in the stiffness of the overall structure and would therefore affect the deflections.

Of the parts that remained, the rail structure and rail supports were further simplified. The rail was simplified because it had a very intricate design and was difficult to mesh. Both the original and simplified profiles of the rail are shown in Figure 3.10. As you can see, the original part had different thicknesses on the different sections of the rail, but these were all assigned a uniform thickness in the simplified version. The holes that ran along the length of the rail were also removed in order to further simplify the part. The simplification of the rail was believed to be adequate because the material that was removed would have increased

the stiffness of the rail, while the holes would have lowered it. More material was removed from the part in critical areas than the amount that was added by filling in the holes, making it potentially less stiff than the original. It is better for the model to be less stiff than the physical piece because it will deflect more, which will provide a more conservative estimate of the deflections. If the allowable deflections are not surpassed with a less stiff part, then there is greater confidence that the real part will not surpass them as well. The rail supports were also simplified. A very slight bit of material was added to the top of the parts, and the hole through the middle was filled. The simplified supports will be a little bit stiffer than the original because of the additional material. However, the part was already very stiff and will not deflect much either way.

Figure 3.10: Original compared with simplified versions of the rail and rail support: a) original rail, b) simplified rail, c) original rail support, and d) simplified rail support.

Meshing

Arguably the most critical step of the FEA process is developing good meshes for the parts. An FEA mesh divides the model of interest into smaller finite elements. These elements allow the deflections to be estimated along the length of the structure. Since continuous parts are being represented by a finite number of elements, the solution from the FEA will differ from the real solution. It follows that as the element size decreases, the FEA solution generally converges to the real solution [32]. The elements should not be made too small however, because the computational cost would drastically increase for a minimal amount of gain and

other issues could arise.

The tube and rail were meshed using continuum shell elements. Shell elements are very thin elements and are suitable for parts that have a much greater length than thickness. They are often times used in sheet metal applications for the car industry [32]. Since both the rail and the tube are much longer than they are thick, they are good candidates for shell elements. These two parts were meshed separately so that different thicknesses could be assigned to their shells. The shells of the tube were set to a thickness of 0.069 inches while those of the rail were set to 0.059 inches.

The rail supports and middle support were meshed with hexagonal brick elements. Brick elements are useful for volumetric structures and are appropriate if parts have an appreciable thickness. Another volumetric element that could have been chosen was the tetrahedral element. Brick elements were chosen instead of tetrahedral elements, because they are less stiff than tetrahedral elements and will therefore more accurately predict the boom deflections [32].

The result of each mesh was checked in order to ensure there were not any elements with substantial aspect ratios. If the elements have large aspect ratios, they will be misshapen and will change the stiffness characteristics of the mesh, which would in turn affect the amount of induced deflection. This check is a simple way to determine if a mesh will work. None of the parts of the mesh had any elements with aspect ratios greater than 10, which is a common value for checking the aspect ratios of elements. The results of the mesh can be seen in Figure 3.11.

Verification

Since FEA can only provide estimates of the deflections and is affected by the element sizes and types, it is important to verify that the model will provide results that are similar to those expected from theory. The easiest way to do this is to work with a simplified version

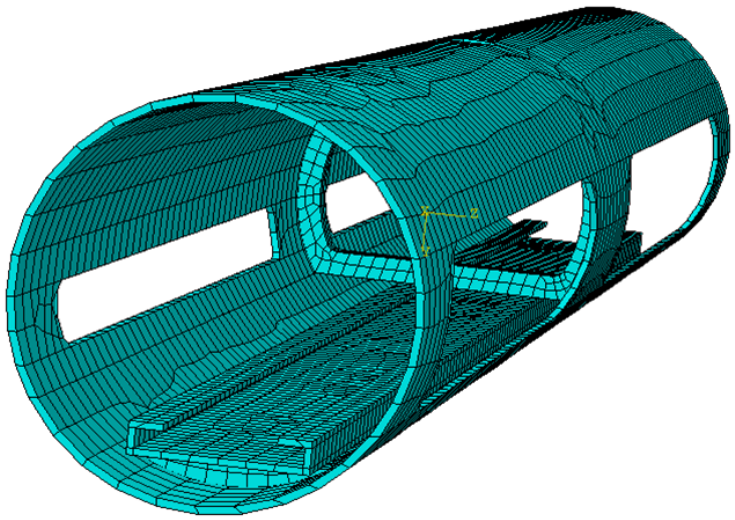


Figure 3.11: Mesh of the stereo boom assembly

of the model. A verification of a simpler model provides confidence that the element types and sizes will be the proper stiffness, thereby increasing the confidence in the results from the more complex model. The model used for this verification is a beam that is the same size and thickness as the tube, but with all of the holes removed. Removing the holes from the model significantly simplifies the theoretical calculations.

The loading applied to the beam for the theoretical deflection calculations is shown in Figure 3.12. The weight of the cameras are loaded at either end of the boom while the weight of the motor is loaded in the middle. In order to further simplify the model, the weight of the tube was neglected in this analysis. The supports were chosen with one pin support and one rolling support in order to have a deterministic force balance, meaning that all of the unknown forces could be determined by force and moment equations.

The unknown forces at the supports were determined through the force and moment balances as shown in the equations below. The sum of the moments was taken about point

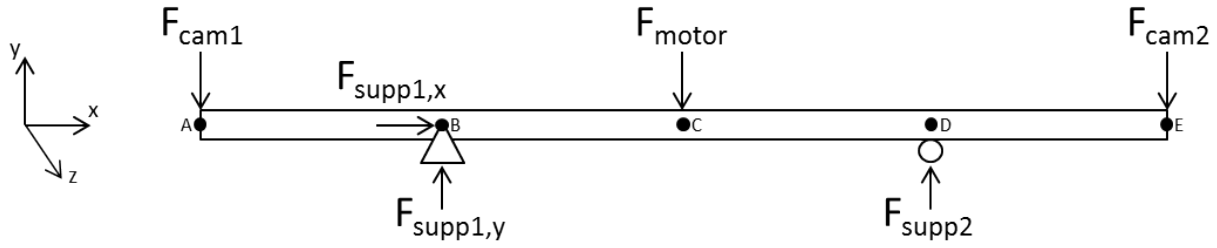


Figure 3.12: Loading of the tube for verifying FEA mesh

B.

$$\sum M_B = 0 \Rightarrow F_{cam1}(d_{AB}) - F_{motor}(d_{BC}) + F_{supp2}(d_{BD}) - F_{cam2}(d_{BE}) = 0 \tag{3.6}$$

$$\sum F_y = 0 \Rightarrow -F_{cam1} + F_{supp1,y} - F_{motor} + F_{supp2} - F_{cam2} = 0 \tag{3.7}$$

$$\sum F_x = 0 \Rightarrow F_{supp1,x} = 0 \tag{3.8}$$

where F is the force of a given object and d is the distance between two points. The force for each cameras was 1 lb, that from the motor was 0.55 lbs, d_{AB} was 7.86 inches, d_{BC} was 8.14 inches, d_{BD} was 16.28 inches, and d_{BE} was 24.14 inches. Simultaneously solving the equations leaves a total of 1.275 lbs at each support.

Now that all of the forces have been calculated, our goal was to determine the deflection of the beam. This was performed by utilizing singularity functions. Singularity functions are functions that can be activated at certain lengths along a beam allowing integration at singularities, such as point loads. These functions are inactive if the value contained within them is negative, and are active when it is positive. This is especially useful when integration along the length of a beam is required, like in the case of determining the deflection of a beam. Different types of loading requires different singularity functions. For instance, a moment acting at a point, a, has a the singularity function $\langle x-a\rangle^{-2}$, while a force acting at the same location is described by $\langle x-a\rangle^{-1}$. There are different singularity functions for other loadings, but it is not essential to present them here. The only one that will be required for this analysis is that for concentrated forces. It is important to note that these singularity

functions have somewhat peculiar integration properties. If the singularity function is raised to a negative power, the integral is simply the singularity function raised to the base power plus one. The singularity function is not multiplied by a constant. However, if the singularity function is raised to a positive power, the integral follows normal integration rules [33].

The first step in solving for the beam deflection is to determine the load intensity function. For this loading, it becomes

$$\begin{aligned} q &= -F_{cam1}\langle x\rangle^{-1} + F_{supp}\langle x-d_{AB}\rangle^{-1} - F_{mot}\langle x-d_{AC}\rangle^{-1} \\ &\quad + F_{supp}\langle x-d_{AD}\rangle^{-1} - F_{cam2}\langle x-d_{AE}\rangle^{-1} \\ &= -\langle x\rangle^{-1} + 1.275\langle x-7.86\rangle^{-1} - 0.55\langle x-16\rangle^{-1} + 1.275\langle x-24.14\rangle^{-1} - \langle x-32\rangle^{-1} \end{aligned} \tag{3.9}$$

Since the beam does not extend past F_{cam2}, that term is never active and can therefore be removed from the load intensity function. Integrating this function twice provides the moments in the beam as shown in equation 3.10.

$$M = -\langle x\rangle^{-1} + 1.275\langle x-7.86\rangle^{-1} - 0.55\langle x-16\rangle^{-1} + 1.275\langle x-24.14\rangle^{-1} \tag{3.10}$$

Assuming small deflection angles, the vertical deflection, y, at any location along a beam, x, is given by

$$EI\frac{d^2y}{dx^2} = M \tag{3.11}$$

where E is the modulus of elasticity and I is the second area moment of inertia [33]. The modulus of elasticity is a material property and is 10,000,000 psi for aluminum 6061-T6. The second area moment of inertia is related to the cross section of the beam and is equal to 1.094 in^4 for the tube. Integrating twice and setting $y = 0$ at both $x = 7.86$ and $x = 24.14$

and solving for the integration constants leaves

$$y = \frac{1}{EI}(-0.1667\langle x\rangle^3 + 0.2125\langle x-7.86\rangle^3 - 0.0917\langle x-16\rangle^3 + 0.2125\langle x-24.14\rangle^3 + 85.76x - 593.1490) \tag{3.12}$$

which is equal to -5.417×10^{-5} inches at $x = 0$ and $x = 32$. This value will be compared with the FEA model to ensure that they are similar.

As was previously mentioned, the FEA model used for this verification step is a model that is solely the tube with no holes cut in it. The tube model was meshed using the same size shell elements as those in the full mesh and was loaded similarly to the theoretical model. The forces of the cameras at the end of the beam were simulated by 1 lb friction loads, while the motor was made into three equally spaced point loads along the center of the beam. The supports were placed halfway up the tube on the sides, because the theoretical bending model assumes that the bending occurred about the neutral axis of the system, which is located halfway up the tube. Upon running the FEA simulation, the y displacements at the edges were probed at the neutral axis. The values were all very similar and had a magnitude of around -4.757×10^{-5} which is 13.87% different from the theoretical calculation. This is fairly close to the theoretical calculations and provides confidence that the original mesh will be able to accurately estimate the deflections of the boom.

Mechanical Loading

The next step is to simulate the loads that are expected on the boom and to find the largest deflections in the local region around the cameras. This will provide insight into whether or not the boom is stiff enough to use one calibration file. There are two separate loading cases that will be analyzed. One of the load cases places the cameras at the largest baseline while the other puts them at the smallest, as shown in Figure 3.13. These two load cases illustrate the worst case scenarios. The load from the cameras was applied as

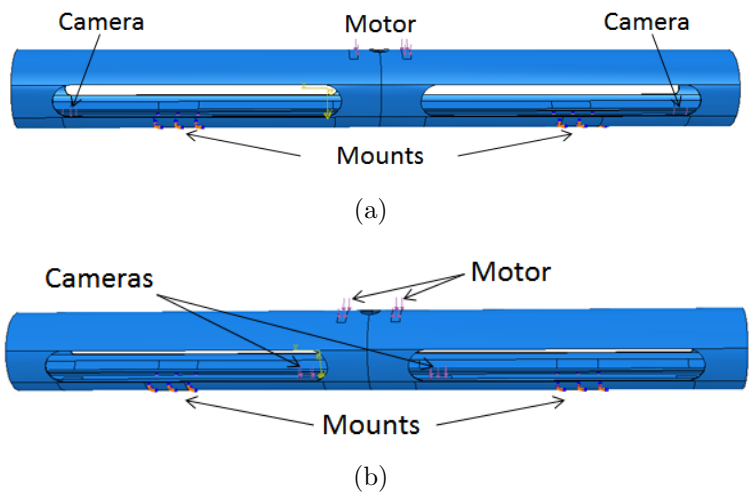


Figure 3.13: Load conditions on the a) fully extended boom, and b) fully contracted boom

a constant pressure located on the top of the rail where the bottom of the bearings would rest. The weight of the motor was applied as pressure loads at the top of the boom at the locations where the motor mounts sit. A gravity loading was also applied to the model so that the final deflections will include the weight of the rest of the parts. The FEA model was supported by two fully constrained mounts, which would represent a rigid mount welded onto the boom. The model was analyzed at 68°F, which becomes important later.

The FEA was run and the deflections at the point of camera loading were recorded. The results of the FEA simulation with the cameras at full baseline can be seen in Figure 3.14. These deflections determined the difference in the roll, pitch, and yaw between the cameras at their outermost and innermost positions. The equations for the angular differences are derived assuming that the boom deflects symmetrically about the middle of the boom, which is true for these loading cases. Equation 3.13 shows the yaw angular difference, equation

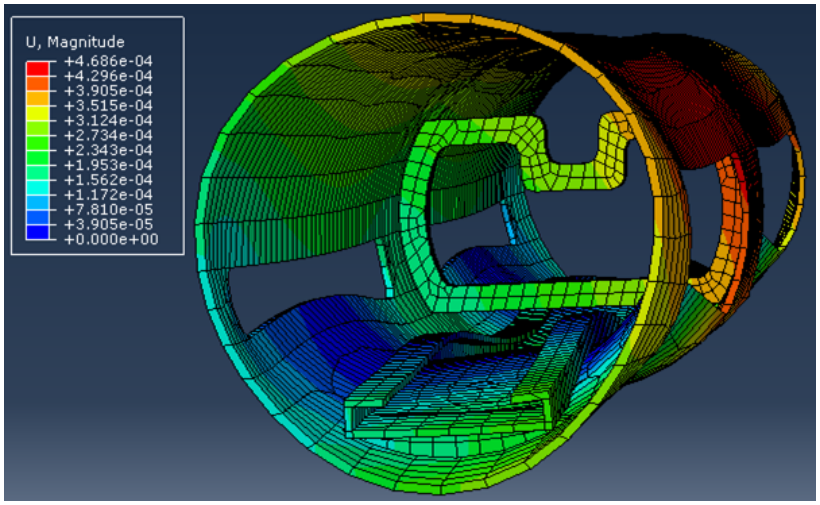
3.14 shows the angular difference for pitch, and equation 3.15 shows that for roll.

$$\Delta\Psi = 2\left(\sin^{-1}\left(\frac{\Delta U_z}{\Delta U_x}\right)_{out} - \sin^{-1}\left(\frac{\Delta U_z}{\Delta U_x}\right)_{in}\right) \tag{3.13}$$

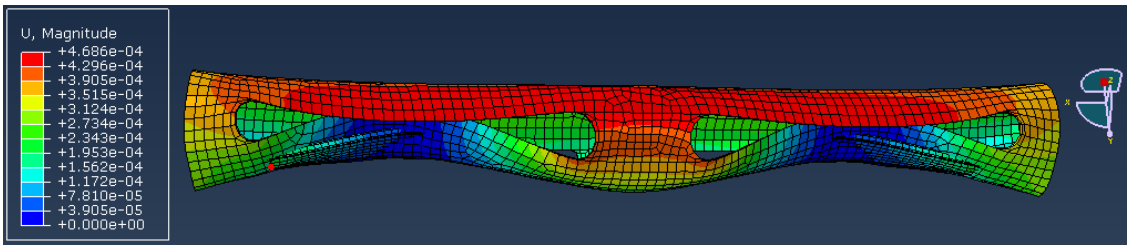
$$\Delta\Phi = \sin^{-1}\left(\frac{U_{y,front} - U_{y,back}}{d_{FB}}\right)_{out} - \sin^{-1}\left(\frac{U_{y,front} - U_{y,back}}{d_{FB}}\right)_{in} \tag{3.14}$$

$$\Delta\Theta = 2\left(\sin^{-1}\left(\frac{\Delta U_y}{\Delta U_x}\right)_{out} - \sin^{-1}\left(\frac{\Delta U_y}{\Delta U_x}\right)_{in}\right) \tag{3.15}$$

where U_y is the vertical deflection, ΔU_y is the difference in the vertical deflections for either the front or back of the load area, ΔU_z is the difference in the forward deflections, ΔU_x is the distance along the length rail, and d_{FB} is the distance from the front of the camera load area to the back of it along the rail. The subscripts, out and in, signify which loading condition



(a)



(b)

Figure 3.14: FEA representation of deflected stereo boom from the side, a) and the front, b)

the deflections were gathered from, while the subscripts, front and back, describe the position where they were measured. Figure 3.15 depicts the angles in order to aid with understanding how the equations were derived. They depict the angles for one loading condition.

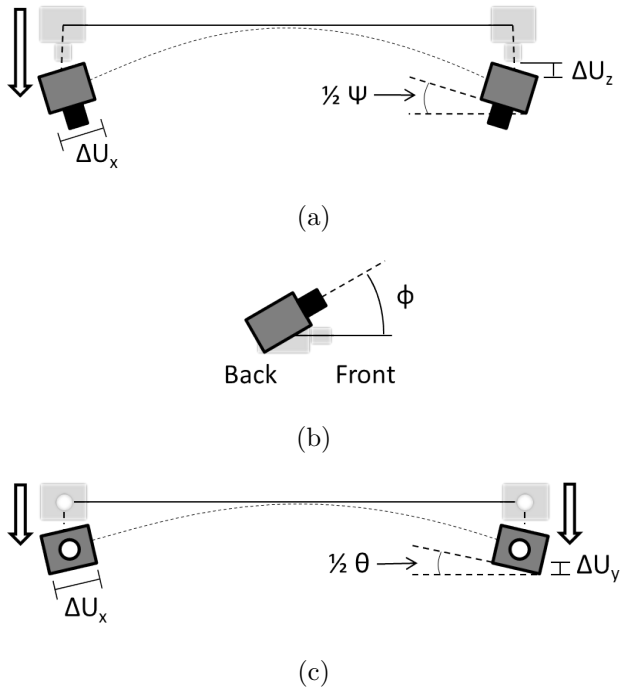


Figure 3.15: Angular deflections for the different rotations with a) yaw, b) pitch, and c) roll

The analysis resulted in angular deflection differences of 6.00E-04° in yaw, 3.66E-02° in roll, and 1.36E-03° in pitch between the two load cases. Comparing these results with those from Table 3.3, it is evident that the stereo boom should be stiff enough to use only one stereo calibration file at room temperature.

Thermal loading

Since the boom is expected to be used in outdoor environments, it must be able to work at a variety of different temperatures. Temperature variations cause materials to shrink and expand. This expansion and shrinking induces strains in the materials which could create stresses in constrained and non-uniform structures, such as the boom. The FEA that was

Table 3.7: Angular deflections at different temperatures for fully constrained boom. Deflections that are larger than the maximum allowable deflections are shown in red.

Temperature	Parameter	Max Deflection (deg)	Allowable Deflection (deg)
0°F	Yaw(Ψ)	1.49 E-02	2.36 E-02
	Pitch(Φ)	3.35 E-02	1.22 E-01
	Roll(Θ)	3.91 E-01	1.25 E-01
100 ° F	Yaw(Ψ)	6.12 E-03	2.36 E-02
	Pitch(Φ)	1.39 E-02	1.22 E-01
	Roll(Θ)	1.86 E-01	1.25 E-01

presented previously was performed at room temperature, and this section will use the same procedure to analyze the boom at 0°F and 100°F. These temperatures are estimations of the maximum and minimum temperatures that a typical work environment would see.

In order to understand the effect different temperatures will have, the thermal expansion coefficient of the material must be known. The thermal expansion coefficient describes how much a material will deform at a given temperature. Oftentimes, the thermal expansion of a material is nonlinear and so the thermal coefficients are valid for a range of temperatures, but are different outside that range. The thermal expansion of Aluminum 6061-T6 is $13.1\frac{\mu in}{in°F}$ from 68 to 212°F [30]. This means that at 0°F the aluminum has different properties. However, it was difficult to locate data for aluminum below this temperature range. Therefore, the same thermal expansion coefficient was used for both cases. This should be acceptable because the thermal expansion coefficient tends to increase with increasing temperatures [34], which means that this thermal coefficient should overestimate the strains.

The FEA was performed at the two different temperatures and the deflections were once again measured. These deflections were used to find the roll, pitch, and yaw discrepancies between the different camera locations. The results can be seen in Table 3.7. According to the FEA results, the boom would not function with just one stereo calibration file at either of the temperatures.

These results make it seem as though the fully constrained mounting system would not work properly. They restrict the expansion of the boom too much. In order to reduce the

Table 3.8: Angular deflections at different temperatures with pin and slider supports. Deflections that are larger than the maximum allowable deflections are shown in red.

Temperature	Parameter	Max Deflection (deg)	Allowable Deflection (deg)
0°F	Yaw(Ψ)	2.83 E-03	2.36 E-02
	Pitch(Φ)	2.81 E-03	1.22 E-01
	Roll(Θ)	7.67 E-02	1.25 E-01
100 ° F	Yaw(Ψ)	1.60 E-03	2.36 E-02
	Pitch(Φ)	2.26 E-03	1.22 E-01
	Roll(Θ)	6.76 E-02	1.25 E-01

temperature induced strains, the mounting system was changed from fully fixed to a pin and slider configuration. This configuration could represent a vibration isolation system that will allow the boom supports to move slightly, thereby relieving some of the strain. The FEA analysis was performed again and the results are shown in Table 3.8. The new mounting system made the resulting FEA asymmetric, which created some issues regarding the angular differences between cameras. One side experiences slightly more deflections than the other. This meant that the orientation needed to be gathered from both sides of the boom and the results were summed to determine the total differences. The results suggest that the boom would only need one calibration to perform the stereo reconstructions if a pin and slider mounts are used. Note that the allowable pitch term is more than double the observed deflections, which means that the full resolution images should work with this boom as well. The design of the mounts will not be presented in this thesis, but it is recommended that one be created with vibration isolation so that thermal expansion can occur.

3.2.4 Final Boom

Having verified that the mechanical design will function with one stereo calibration file, the boom was constructed in an attempt to satisfy the design. The tube structure was manufactured by the Mechanical Engineering Machine Shop at Virginia Tech, while the rest of the components were fabricated at the USL. All of the parts were assembled and the finished boom can be seen in Figure 3.16. The stereo boom has a maximum baseline of

26.7 inches and a minimum baseline of 5.7 inches. These values are similar to the specified values of the design and only differ because of the camera cabling. The complete boom, with everything assembled, weighs in at 5.35 lbs. This does not include the batteries that run the cameras and motor, nor does it include the microcontroller and motor controller that position the cameras. All told, the system is expected to weigh somewhere near the 10 lb goal, but it is uncertain exactly how much it will weigh. The final weight will be heavily influenced by the electronics. They could be reduced to a single battery and printed circuit board, thereby saving weight. The electronic development will not be performed in this thesis because this device is going to only operate as an initial prototype.

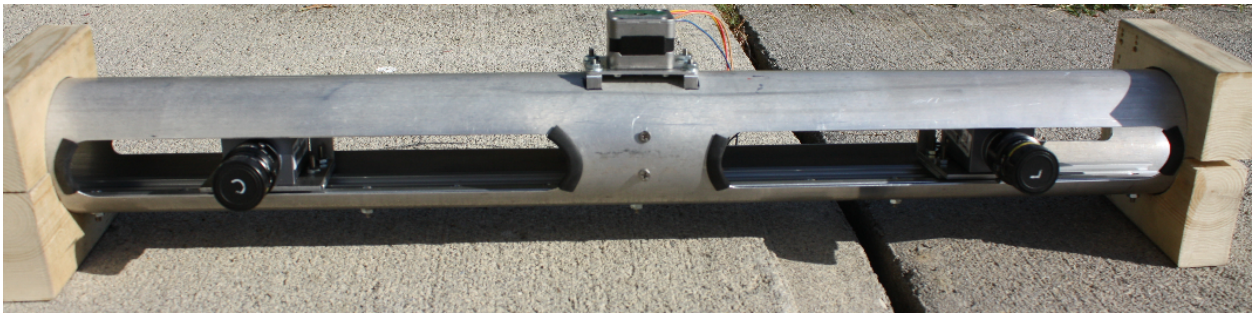


Figure 3.16: Completed stereo boom

At this point, it is important to determine whether or not the weight estimates of the camera structures were correct. These weight estimates were used as the loadings in the FEA models. Verifying that the real weights were similar to the estimates would help to conclude whether or not the FEA models were accurate in terms of calculating the deflections. The cameras, mount, rack, and bearing weighed in at 16.04 ounces, which is very close to the 1 lb that was estimated at the onset. This further supports the results of the FEA.

A few issues arose upon the completion of the boom pertaining to the Igus bearings. They created a lot more friction than was estimated. This had the potential to require a more powerful motor. To quantify just how much greater the force was than was expected, a simple test was run. The entirety of the boom was placed vertically on a scale and the scale was zeroed. The cameras were pushed down the track and both cameras together

required a total of nearly 3.75 lbs of force, which is a little over 12 times greater force than was anticipated. The torque required to move the cameras equated to 3.02 in-lbs, which is somewhat close to the 3.75 in-lb limit of the stepper motor. To reduce the required torque, a silicon lubricant was applied on the interior of the bearings. Silicon lubricant was chosen because of its relatively inert properties and because it does not readily attract dirt. The lubricant was able to cut the required amount of torque in half.

Another problem that presented itself during construction, was that the nylon racks were bowed significantly. They were bowed so extremely that it was difficult to move them within the confines of the boom. It was decided to add stiffeners to the racks in an attempt to straighten them out. Aluminum sheet metal was bent in a 90 channel and was epoxied to each rack. The stiffness of the sheet metal was sufficient to render the racks straight. However, being that the boom was very confined, the addition of the aluminum caused the racks to rub along the interior of the boom. This created a larger than expected torque on the bearings, which caused them to wear down strangely. In future implementations, it is recommended that the racks are either straightened before construction, so that the aluminum would not be necessary, or that the racks are made from a stronger material that would already be straight.

Chapter 4

Stereo Vision

This chapter will present and discuss the stereo vision that is performed between the cameras. It will begin with a look at whether the boom, in its manufactured state, would be able to use one calibration file. Unfortunately, it can not. This required a further look into the calibration by analyzing the results from calibrations with the cameras at different locations along the length of the boom.

4.1 One Calibration File Results

The adjustable baseline stereo boom was designed in order to create SGBM reconstructions at different lengths along the boom, while only using one calibration file. This would simplify the system immensely. As was presented in Section 3.1.1, the vertical offset between the left an right images can not exceed 2 pixels. This section will present results that determine whether or not the offset between the left and right images gathered from different points along the boom is greater than 2 pixels. If it is, then one calibration file can not be used for reconstructions.

In order to perform the test, the stereo boom was calibrated with the cameras halfway

in between the extremes of the baseline. Calibrating the boom at the middle would provide the best possible chance for creating good stereo reconstructions with only one calibration file. This calibration consisted of both the intrinsic and extrinsic parameters between the two cameras.

Five image pairs were then gathered at the middle position, the fully contracted position, and the fully expanded position. The cameras were zeroed before moving to the different locations in order to make sure they were at the proper location. For both the fully expanded and fully contracted locations, the translation in between the cameras was changed in the calibration file, so that it matched the physical distance between the cameras at that point. This would represent the case were the translation between the cameras was the only calibration parameter to change for different points along the boom. None of the other parameters could be changed because there was not any information available about them once the cameras are moved. The translations between the cameras at the maximum and minimum baselines, however, are known.

All of the images were then undistorted and rectified into the frontal parallel position. The resulting images were then searched for SIFT feature points so that the vertical offset between the images could be determined. The SIFT features were matched between the corresponding left and right images and the incorrect matches were removed from the data set. The difference between their y positions were determined for all of the correctly matched feature points, and the average offset was recorded. If the rectification worked properly, the average vertical offset should be zero. This procedure was repeated with each of the five images from all three locations and the results are presented in Table 4.1. As is evident, the vertical offset is around zero for all of the middle images, while the others have mean offsets of greater magnitude than 2 pixels.

Statistical procedures were run on the results to see whether or not they were statistically significant. A Tukey-Kramer multiple comparison test was performed on the data in order to make the determination for the observed differences in the vertical offsets [35] [36]. With

Table 4.1: Vertical offsets at different positions while using only one calibration file

Trial	**Min Baseline**	**Middle**	**Max Baseline**
1	-2.769	0.321	2.185
2	-2.337	-0.279	2.145
3	-1.847	-0.268	2.500
4	-3.195	0.145	2.283
5	-2.555	0.265	2.112
Mean	-2.541	0.037	2.245
Std. Dev.	0.501	0.290	0.156

95% confidence, the vertical offsets are all statistically different from one another. Student t-distribution one-sided hypothesis tests were used in addition to conclude with 95% confidence, that the vertical offsets at the interior and exterior positions are greater than two pixels [36]. Since the means are greater than 2 pixels, it appears as if the stereo boom can not be operated with one calibration file. The difference in vertical offset can be seen in Figures 4.1 and 4.2, which show some of the epipolar lines in images from both the middle and maximum baselines, respectively. The epipolar lines should pass through the same locations in both images of a pair. For the images gathered at the middle they do, but at the other locations they do not.

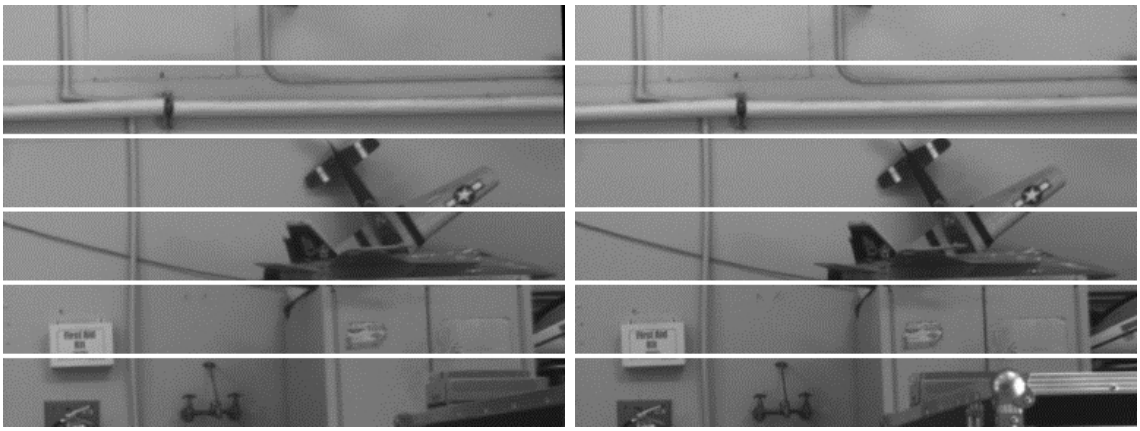


Figure 4.1: Epipolar lines for imagery from the middle baseline with the middle calibration. The white lines represent some epipolar lines and are there to help see the vertical offsets. Images small portions of the full size images.

In order to make sure that the results are correct and not just an artifact of a poor

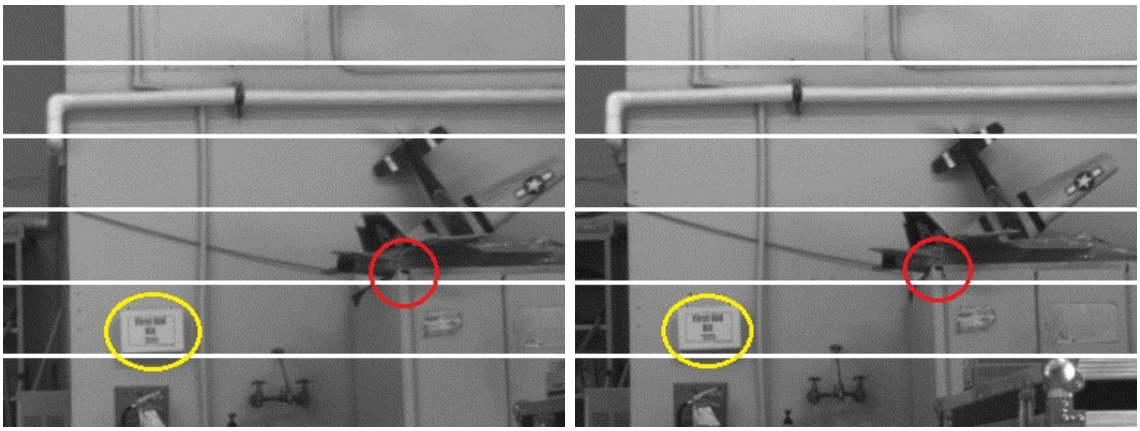


Figure 4.2: Epipolar lines for imagery from the maximum baseline with the middle calibration. The white lines represent some epipolar lines and are there to help see the vertical offsets. Notice how the objects within the images shift vertically. Images small portions of the full size images

Table 4.2: Vertical offsets at different positions using the calibration results for that location

Trial	**Min Baseline**	**Max Baseline**
1	-0.129	-0.380
2	0.369	-0.021
3	-0.017	0.145
4	-0.164	-0.021
5	-0.151	0.013
Mean	-0.019	-0.053
Std. Dev.	0.224	0.195

calibration, the images at the minimum and maximum locations must be put through the same test with their respective calibrations. The cameras were calibrated at both of these baselines and the test was repeated. The results of the tests are displayed in Table 4.2. The vertical offsets at both positions are very close to zero, as was expected. Since each location has minimal vertical offsets with their respective calibrations, it strengthens the conclusion that solely one calibration file can not be used.

It is uncertain as to where the offset stems from, but there are a few possible culprits. One explanation is that coupling between the rotations created a much larger vertical offset than was expected. Another possible reason could be that the bearings are not perfect. They are

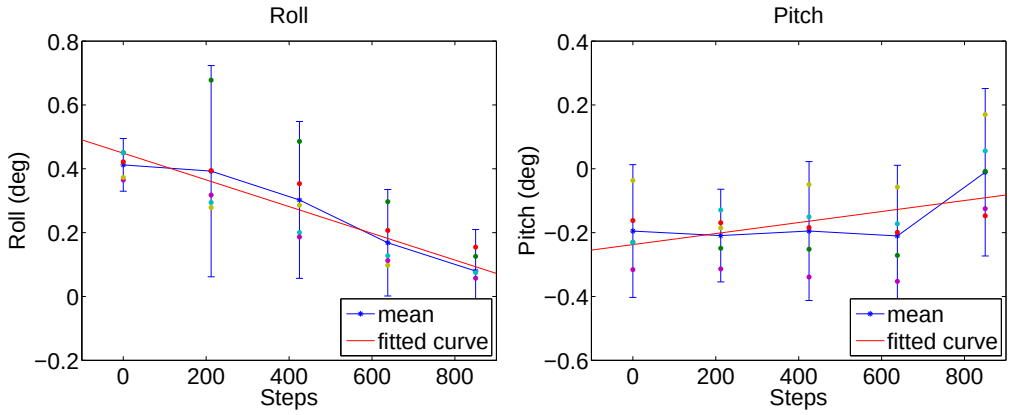
nice and tight bearings, however, there is still some amount of play associated with them. Otherwise, they would not move. This little bit of slack could introduce enough movement to cause the vertical offsets to differ.

4.2 Calibration Relationship

Since a single calibration file would not be able to produce very good stereo reconstructions, it becomes interesting to look at the relationship between the calibrations. If all of the components define a clear relationship with respect to position, it may be possible to place the cameras anywhere along the length of the boom and still create quality stereo reconstructions. This could be achieved by interpolating between the calibration results. The relationship would be most useful if it were linear.

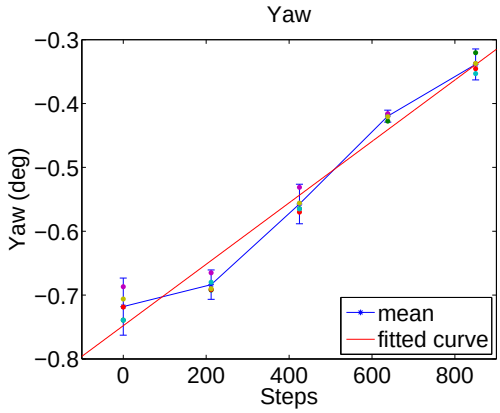
In an attempt to characterize the relationship, the boom was calibrated at five different equally spaced positions along the boom. The routine was repeated until five calibration results existed at each position. They occurred on five different days and after using the boom. The importance of performing multiple calibrations is to determine if the calibration process is at all consistent. The trials should all be similar because not much should change in between them. They have been plotted and are shown in Figure 4.3.

These plots show the results of the calibration routines for each individual trial in different colors, as well as showing the mean values at each position with error bars in blue. The error bars are plus and minus two standard deviations away from the mean of the data at that position. Two standard deviations was chosen because, assuming the results are Gaussian, which should be the case, two standard deviations would encompass 95% of the points [36]. The means for the trials are not the population means and are therefore subject to change, but it does help in understanding what values should be expected. Linear regressions were performed for each parameter to determine if the parameter changed linearly along the length of the boom and are shown in red. The results suggest that certain parameters are linear,

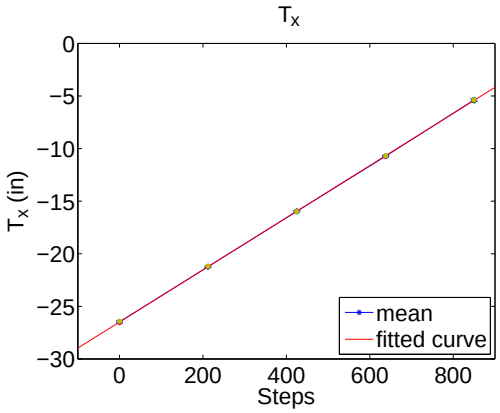


(a) Roll: linear fit $R^2 = 0.95$

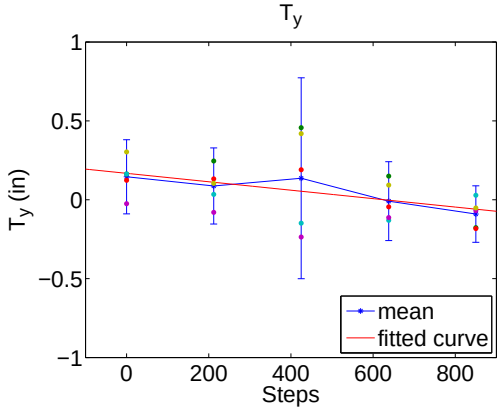
(b) Pitch: linear fit $R^2 = 0.46$



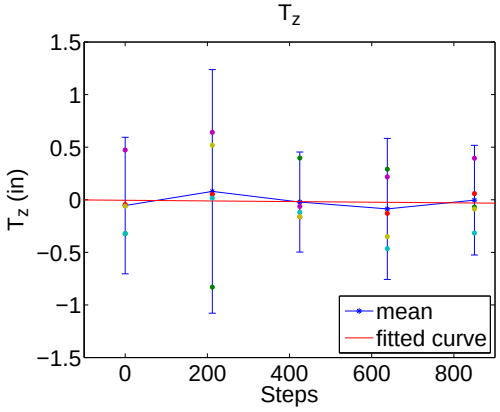
(c) Yaw: linear fit $R^2 = 0.97$



(d) T_x: linear fit $R^2 = 1.00$



(e) T_y: linear fit $R^2 = 0.79$



(f) T_z: linear fit $R^2 = 0.03$

Figure 4.3: Graphs of the different calibration results. The blue line is the mean of the results from each position with error bars of a total of 4 standard deviations. The red line is the linear best fit line through the means.

i.e., roll, pitch, T_x, T_z, and T_y. The yaw is not. These are based heavily on the coefficient of performance value, R^2, which is the estimation strength of the linear regression. A high R^2 value (close to 1) implies a useful relationship, while a low one (close to 0) implies that it would be better to use the mean of all the positions as an estimate [36]. Although T_z and pitch have low R^2 values, it appears that it would be alright to use the mean value as an estimate for all of them. Yaw may appear to have a linear relationship, but the linear regression does not pass through all of the error bars. This implies that it is unlikely for yaw to be linear along the length of the boom and that another relationship most likely exists. That being said, the R^2 value is fairly high for yaw which implies that the linear relationship would do well as an estimation tool.

Even though some of the results are nonlinear, it does not appear that they are extremely nonlinear. There are only five data points, which means that it is possible for higher frequency nonlinearities to be aliased [37]. Even if that is the case, the results still appear as if they could be used to linearly interpolate between two of the calibration locations, because the nonlinear equation will approach linear as the discretization increases. The positioning system that will be described in Chapter 5 will utilize the multiple calibrations.

4.3 Stereo Results

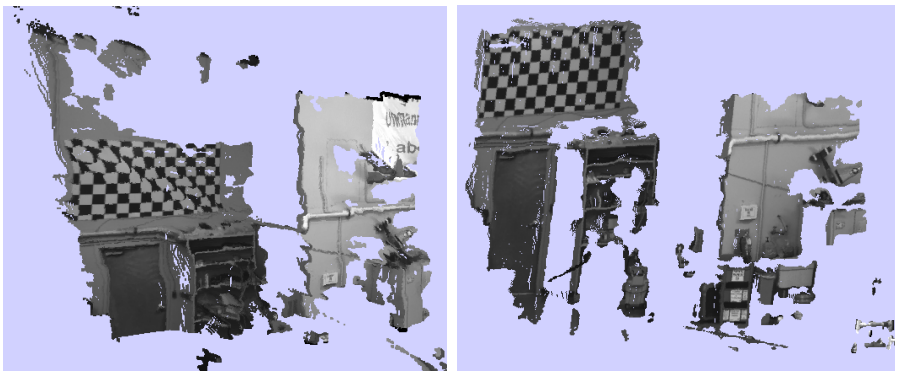
This section will show some results of the stereo reconstructions. It is mainly used to illustrate what stereo reconstructions look like from using SGBM at different baselines and calibration files. An example of a reconstruction of a person close to the cameras, possibly too close, for the minimum baseline can be seen in Figure 4.4. The reconstruction did a fairly good job, but failed to pick up the areas that were at steeper angles with respect to the cameras. This phenomenon will be discussed further in the following chapter.

Figure 4.5 shows stereo reconstructions from images gathered from roughly 25 feet away and for both the minimum and maximum baselines. The figure attempts to illustrate that

Figure 4.4: Stereo reconstruction of the author. I was a little spaced out that day.

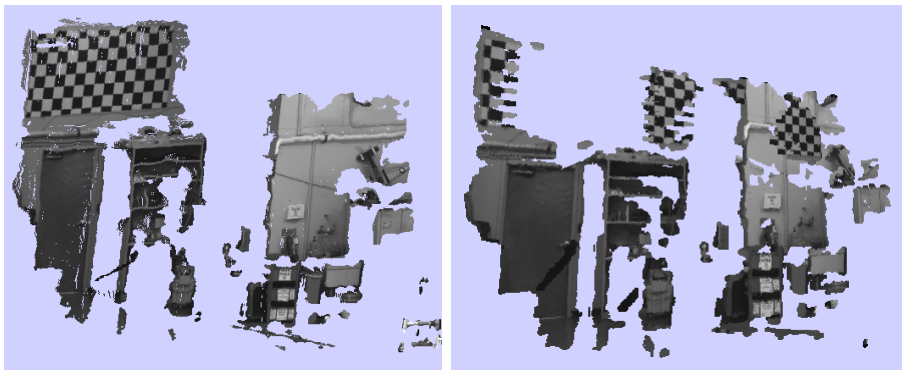
the reconstructions have a lot better resolution with the cameras at the maximum baseline than at the minimum. Also, there are holes in the reconstructions in the textureless regions of the imagery such as on the walls.

To illustrate the importance of using the appropriate calibration file, stereo reconstructions from the maximum baseline are created using the maximum baseline calibration file and also the middle baseline calibration. The results are shown in Figure 4.6. The reconstruction from the maximum baseline with the middle calibration had significant mismatches that even resulted in the calibration rig being estimated at three separate locations, while the other one appears accurate.



(a) (b)

Figure 4.5: Stereo reconstruction from a) the minimum baseline and b) maximum baseline



(a) (b)

Figure 4.6: Stereo reconstruction from maximum baseline with a) the correct calibration and b) the middle calibration

Chapter 5

Camera Positioning

The variable baseline stereo boom allows the cameras to be moved to a large variety of different positions. However, the system becomes less useful without a method to automatically place the cameras given the location of an object of interest. This chapter will present the hardware and technique that has been implemented in order to properly place the cameras for effective stereo reconstructions. It assumes that the system has some sort of object recognition in order to localize an object of interest within the frontal parallel rectified imagery, and provide the location of the object's bounding box. The technique will rely heavily on experimental data pertaining to SGBM reconstructions for an object at different ranges from the cameras. The chapter will begin with an overview of the camera positioning hardware. The next section will further describe the metric for positioning the cameras based on the capabilities of the SGBM stereo vision algorithm. This will be followed by a description and derivation of the control scheme used for moving the baseline in both continuous and discrete fashions. The chapter will conclude with the results obtained from implementing the control technique on the stereo boom hardware.

5.1 Hardware

The cameras must be able to be repeatably positioned in a large variety of different spots along the length of the boom. Otherwise, the calibration results would not work and would require uncalibrated stereo vision techniques, thereby making stereo vision more difficult. Repeatable positioning can be achieved with a microcontroller, a stepper motor driver, and a limit switch.

The microcontroller receives position commands from an algorithm that is running on a laptop computer, by way of serial communication over USB. The positioning algorithm sends the microcontroller a number of steps that the motor must rotate, as well as the direction that it needs to turn. The microcontroller then parses that information, and sends a pulse-width modulated signal with a 50% duty cycle to the stepper motor driver along with the direction that the system must move. The stepper motor driver interprets the input signal and moves the motor based on the number of low-to-high transitions in the signal.

The microcontroller that was chosen for this task was the Arduino Duemilanove. The Arduino Duemilanove is a user-friendly microcontroller that can easily interpret serial commands. The Easydriver v4.4 stepper motor driver was used to move the stepper motor. It is a cheap stepper motor driver that can provide enough current to control the motor. However, it does have some limitations. The Easydriver v4.4 is a printed circuit board with a large chip on the middle of it, which provides current to the stepper motor. The problem is that the chip gets extremely hot ($\approx 220^\circ$ F), and stops holding the stepper motor at stall torque. This causes the Easydriver v4.4 to occasionally twitch and move the cameras. Since there is no positional feedback from the cameras, any unwanted twitches would harm the stereo estimations. This will be alright in a testing environment, because the chip can be put to sleep, but it would cause problems in a real world scenario were there are large amounts of vibrations. An aluminum heat sink was added to the chip to try to dissipate the heat faster. The twitches have not occurred since. However, a different motor driver chip may be advantageous. A possible replacement is the SN754410 Quadruple Half-H Driver.

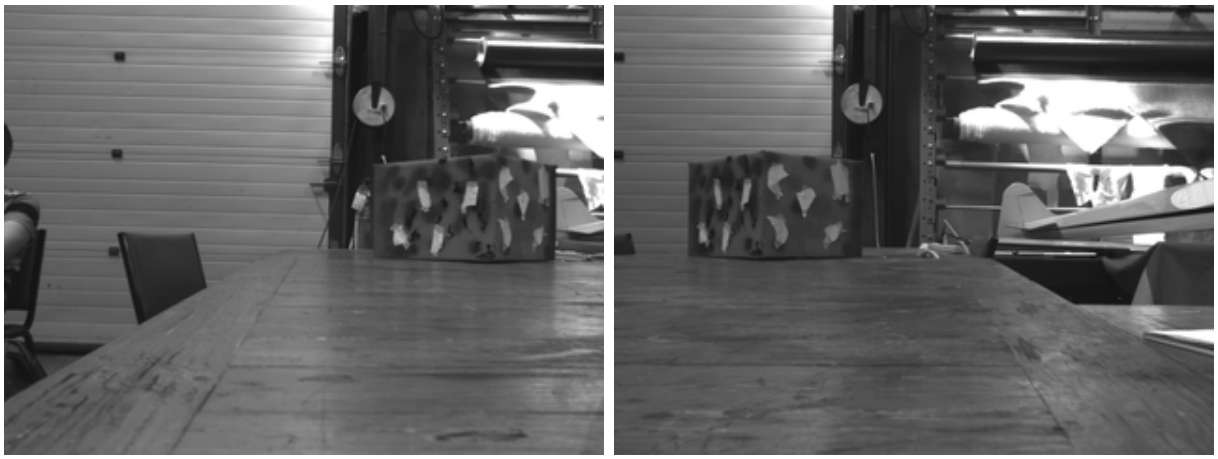
The last part of the positioning system was the limit switch. As was described in Section 3.2.2, the limit switch was used to zero the cameras at power up. Zeroing the cameras places them at a known global position. Therefore, recording the number of steps will allow the global position to be known at all times, assuming that the stepper motor does not have any problems. The position of the camera is stored on both the laptop and the microcontroller by recording the number of steps that have been moved, and which direction they were moved. This information is vital to the stereo boom system because there is no positional feedback. Without it, it would be uncertain where the boom is located.

5.2 Stereo Vision Metric

The positioning algorithm of the variable length stereo boom draws heavily on the effectiveness of the SGBM stereo vision technique. Stereo vision requires that objects are visible in both cameras, and window based stereo vision techniques, such as SGBM, also necessitate that objects look similar in both images. If either of these requirements are not met, those regions of the imagery will not be reconstructed in 3D. These problems become especially evident as an object approaches the cameras. Imagine looking at a box with one of the corners of the box facing you. Whenever the box is far away, both eyes can each see two sides of the box. However, as the distance between the box and your eyes decreases, one side of the box shrinks in one eye's field of view, while the other one grows. Until eventually, only one side of the box is visible in each eye. The same object can be seen in both eyes, but to each eye, it would appear as two different objects. This phenomenon is experienced by the stereo cameras and makes it difficult to estimate the position of the object in 3D, causing the stereo vision techniques to fail. Fortunately, the problem can be overcome by shrinking the distance between the cameras, thereby ensuring that each camera has a more similar vantage point of the object of interest.

An experiment was devised in an attempt to quantify the point at which the SGBM

algorithm would begin to fail. The experiment was comprised of the stereo boom at full extension and a box that was painted in a type of camouflage. The painting scheme had white areas, black areas, and areas with colors in between. SGBM operates on greyscale images by attempting to localize distinctive regions in terms of similar intensity distributions in both images. The paint job contained scattered portions of different intensities in order create distinctive textures for the algorithm. Failures would then occur because of the different vantage points, and not because of a poorly textured surface, making it more simple to pinpoint where the system would fail. A picture of the box can be seen in Figure 5.1. The figure illustrates the problems that occur when the object is relatively close to the cameras. The box is at a point where different sides of the box are more clearly visible in one image than in the other.



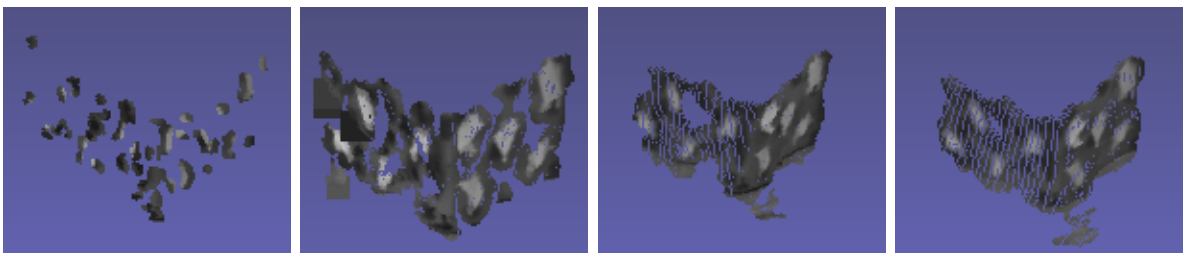
(a) (b)

Figure 5.1: Images captured from box experiment with a) as the left image and b) as the right image. The box can be seen near the middle the images.

To conduct the experiment, the fully expanded stereo boom was placed at the end of a very long table. The box was situated midway in between the baseline of the cameras, such that the sides were at an angle of 45° offset from the image plane. This would represent a medium difficulty and yet realistic object, in terms of shape, to observe. Medium difficulty is meant to describe the ease with which stereo vision can reconstruct the object. Since the box edges are orthogonal, they form a 90° angle in their interior. If this angle were increased,

stereo vision would be more likely to work and would be termed easier. At an angle of 180°, the object would be a flat plane and would be in its easiest configuration. Contrarily, if the angle were decreased, the box would become more difficult to construct until, at 0°, it would be impossible because the same sides would never be visible in both cameras. The box, in its experimental setup, will help quantify the degradation of stereo vision for a medium difficulty object, and will form the metric for the positioning technique. The box was placed at different distances away from approximate location of the cameras' image plane. The distances ranged from 46 inches to 200 inches in increments of 12 inches, which created 13 equally spaced box locations. Stereo imagery was gathered at each of these locations and stereo reconstructions were performed.

The reconstruction from the box between 46 and 92 inches away produced very poor results. There was a large amount of holes in those reconstructions with the size of the holes decreasing as the box moved outward. At the locations between 104 and 128 inches, the reconstructions had some holes and gaps, but the surfaces of the box were mostly reconstructed. Finally, at 140 inches, a complete reconstruction of the box occurred. Every distance beyond 140 inches produced full stereo reconstructions, which means that the minimum distance to achieve a full reconstruction of the box is approximately 140 inches. Some of these stereo vision results can be seen in Figure 5.2. This figure illustrates that as the distance increases, the reconstructions continually improve.



(a) (b) (c) (d)

Figure 5.2: Stereo reconstructions of the box experiment gathered at different distances with a) at 46, b) at 92, c) at 128, and d) at 140 inches.

The cameras and the box form a triangle, as shown in Figure 5.3, which relates the box position to those of the cameras. The size of the triangle depends directly on the distance to the box and the baseline. Since both the distance and the baseline are known, the triangle is completely defined. The angle, θ, in the figure will be referred to as the convergence angle henceforth. The convergence angle is that formed by the difference in vantage points of the two cameras, and can be related to the aforementioned experiment. For a given baseline, the convergence angle increases as the object approaches, and decreases as it recedes. Therefore, the minimum effective experimental distance defines the maximum allowable convergence angle that can create full stereo reconstructions for a medium difficulty object, such as a box at a 45 degree angle. The maximum convergence allowable angle, as calculated from the experiment, is 10.9 degrees. This angle will play a pivotal role in the control scheme.

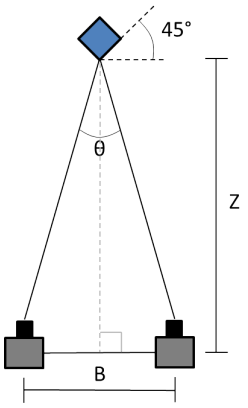


Figure 5.3: Convergence angle, (θ), as shown for the medium difficulty box

5.3 Positioning

This section will present the control scheme for positioning the cameras. A description of the fundamentals of the control framework will be presented first, and will be followed by a description of the control scheme for moving along a continuous baseline. A continuous baseline control method is especially useful if one calibration file is capable of being used, or if a self-calibration technique is being implemented. Otherwise, the continuous method

may create problems with the stereo reconstructions. Given difficulties that have presented themselves by using a single calibration file, this project will rely on multiple calibration files at different spots along the boom, which requires a discrete control scheme. The discrete control technique will be described after the continuous portion.

5.3.1 Overview

As was previously mentioned, the positioning mechanism will rely on the convergence angle formed between the cameras and a localized object. The system will attempt to position the cameras such that the maximum convergence angle of 10.9 degrees is never surpassed, while still having the widest possible baseline, thereby increasing the likelihood of producing a full stereo reconstruction of the object. The baseline should be kept at its widest possible position in order to achieve the greatest depth accuracy. In order to perform this task, it is first necessary to understand what shape will describe the maximum convergence angle. It turns out to be a circle that passes through the camera centers and is known as the theoretical horopter, or the Vieth-Muller circle, and is commonly referred to when dealing with human vision. In terms of terminology, it is a slight misnomer to call the shape the Vieth-Muller circle, because it assumes that the eyes, in this case cameras, are not frontal parallel. Instead, it describes the locations that require the eyes to keep a constant relative angle between them [38] [39]. This stems from the fact that the eyes have one focal point, while the rest of the image is, in a sense, out of focus, which requires the eyes to adjust their angles in order to fixate on one point. This equates to the fixation point, or object, moving positions within the field of view of the frontal parallel cameras. It follows that the proof of a constant convergence angle remains the same regardless of the orientation of the cameras. Despite the slight misnomer, this shape will be referred to as the Vieth-Muller circle for the remainder of the document, and can be seen in Figure 5.4. A full derivation of the Vieth-Muller circle can be found in Appendix A.

This circle describes the closest allowable locations that the object can be situated at

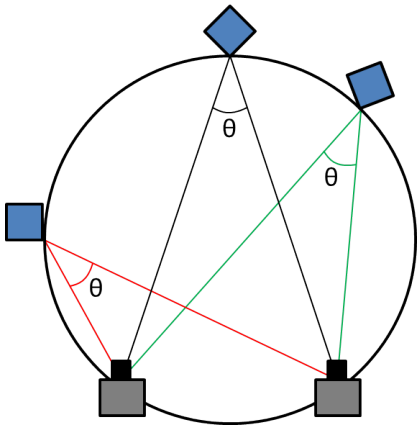


Figure 5.4: Shows the Vieth-Muller Circle with a medium difficulty object at various positions along the circumference of the circle

in order to still achieve a convergence angle of no more than the allowable 10.9 degrees for a given baseline. This should ensure that a medium difficulty object will be reconstructed, which is equivalent to rotating the box along the horopter, as shown in Figure 5.4. The size of the circle is directly dependent on the width of the baseline. As the baseline increases, so to does the radius of the Vieth-Muller circle, and the opposite is also true. This means that the baseline should be increased for better depth resolution until the position of the object is on the circle, while also keeping the object within the field of view of the cameras. This would provide a reconstruction with high depth accuracy, while still having a good chance to fully reconstruct the object of interest. This concept is the main principle behind the positioning algorithm.

5.3.2 Continuous Baseline

The control scheme that will be discussed in this section is that for a continuous baseline. What is meant by a continuous baseline, is one in which the cameras can be positioned at any location along the operational range of the boom. It is not truly continuous because of the discretization of the stepper motor, but it can approach continuous as the resolution of the stepper motor increases. This control technique would require that the cameras are

calibrated at every possible baseline. Otherwise, the stereo vision techniques might fail, or might be inaccurate. This technique could therefore be used so long as the boom is calibrated at every possible baseline, or a self-calibration technique is being employed.

The algorithm begins by gathering images from both of the cameras. Based on the location of the cameras, the images are rectified into the frontal parallel orientation. The baseline determines which calibration results must be used to properly perform this rectification. These newly rectified images could then be searched for instances of the object of interest. As will be discussed in the results portion of this chapter, the objects are currently localized by a user at every instance in time. Depending on whether the object is located in one image, both images, or neither image determines what steps the control algorithm must take.

For the case where the object is located in both images, the algorithm can gather spatial information about the object's position. The distance to the object is calculated by finding a common point within the bounding box of the object, and gathering the disparity of the point between the images. The distance information about that point is found through equation 5.1, which comes directly from the reprojection matrix.

$$Z = \frac{Bf}{d} \tag{5.1}$$

where Z is the distance from the cameras, B is the current baseline, f is the focal length, and d is the disparity.

Using the distance information, the global x-position, x_g, can be determined with respect to the two cameras, which also comes from the reprojection matrix. The positive x direction is defined as the direction from the left camera towards the right one. The zero position of the axis is located midway in between the cameras. This leads to

$$x_g = \frac{-B}{2} - \frac{Z\left(c_x - x_l\right)}{f} \tag{5.2}$$

where c_x is the camera center along the x-direction of the image, and x_l is the position of

the object point in the left image.

The position information is then used to calculate the baseline that would place that point on a Vieth-Muller circle, and is shown in equation 5.3. θ in the equation is the required convergence angle.

$$B_{horopter} = -4\left|\frac{1}{\sin\theta}\right|\left(Z\sqrt{\frac{-1}{4}(\sin^2\theta - 1)} - \frac{1}{2}\sqrt{x_g \sin^2\theta + Z^2}\right) \tag{5.3}$$

The calculated baseline, $B_{horopter}$, is not the only baseline calculation that must be considered. The possible physical baselines that can be obtained from the stereo boom must also be considered along with the maximum allowable baseline, B_{FOV}, that keeps the object in the field of view of both cameras. If the object was not kept within the field of view of both cameras, stereo reconstruction of the object would be impossible. This is not a trivial task and is very difficult to do robustly, because the system must somehow know whenever the entire object is in the field of view. Figure 5.5 illustrates this difficulty. The cameras must be brought together until the entire bounding box is visible in the image. If the bounding box coincides with the edge of the image, it is still unknown as to whether the entire object is visible. The bounding box must therefore have space between it and the side of the image. The image rectification and undistortion present additional problems. They are that the pixel locations of the bounding box can not simply be kept within so many pixels of the edge of the image. This could be done if it were possible to solve for the location of pixels in the undistorted images, but this task is not easy and may be impossible because the undistortion equations are highly nonlinear.

To attempt to solve these problems, it was decided to estimate the global positions of the edges of the bounding box. The global left and right positions of the bounding box are found by using equation 5.2, with the Z that was calculated for the point of interest along with the minimum and maximum image locations of the bounding box. By using the same Z, the system assumes that the object is flat. This is not always the case, and can therefore cause additional problems. However, it works more effectively than using the disparity at

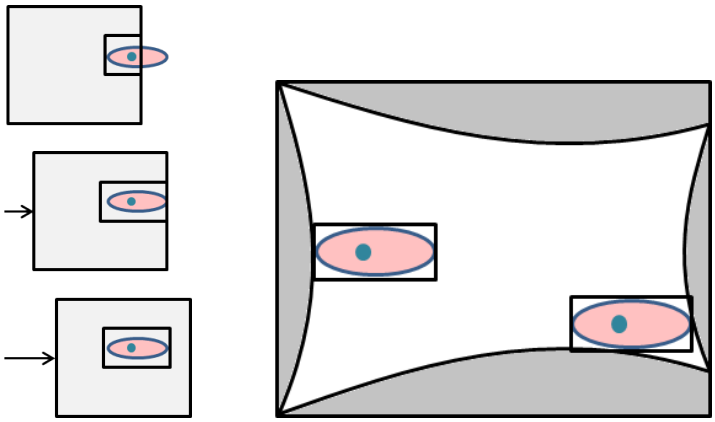


Figure 5.5: Problems introduced by the object bounding box. The object is represented in red and the bounding box in black around the object. a) shows the problems along the edge and b) shows those from rectification and undistortion

the edge of the bounding box, which is heavily influenced by partial bounding boxes and oddly shaped objects. The global positions were determined for the edges of the box. If x_g was positive, the maximum pixel location in the left image becomes important, while the minimum pixel location in the right image becomes important if the opposite is true. The global position is then compared with the field of view at that distance assuming that the actual cameras are frontal parallel. If the x position is within a small percentage from the edge of the field of view, the baseline is shrunk in an attempt to find the entire object bounding box. This technique is not perfect and has problems because of the frontal parallel assumption and lens distortions. Although the technique is not perfect, it does a fine job and allows B_{FOV} to be determined.

Now that $B_{horopter}$ and B_{FOV} are known, they are compared with one another. Which ever one is smaller, is the baseline that the cameras are moved to, such that they do not overreach the minimum and maximum physical baselines. This is what happens if the object of interest can be located in both images.

For situations where the object can only be found in one of the images, the control algorithm attempts to bring the cameras closer together until the object can be found in

both. It is impossible to determine the size of the object without knowing something about the environment, or the physical size of the object. This also means that the global position can not be known. Therefore, an iterative technique must be used. The process works by understanding where the bounding box is located. If it is far enough away from the edge of the image, the cameras are moved inwards. Otherwise the cameras remain in their current position. This is an attempt to keep the object localized within at least one of the images. It would not provide useful depth information to do so, but may be useful in other facets, like turning a vehicle, for instance.

In the last scenario, where the object is not localized in either of the images, the cameras are expanded to their maximum baseline in the hopes of catching a glimpse of the mythical beast. One potential problem with this technique is that there is a blind spot at a location close to the boom and within the cameras. This may cause a collision with the object if the boom is moving towards it. However, this is an unlikely situation and so expanding the boom appears to be the best option. The entire algorithm is summarized in Algorithm 1.

Simulations have been run on the control scheme to see how well it works with perfect knowledge of the global positions. The algorithm was provided with the locations of the object in global space. It determined where the object would be located in the image, which allowed the algorithm to determine what baseline the cameras should be situated at. The algorithm was tested with the object moving relative to cameras. The results of an object moving forward and horizontally can be seen in Figure 5.6. Figure 5.6a depicts the object moving from right to left. At the onset, the object can only be seen from the right camera, which requires that the baseline be shrunk until it is seen in both. The cameras move inward further to try and capture the entire object, and then tries to keep it within the field of view of the cameras. Whenever the object approaches the middle of the cameras, the object is then placed so that it lies on the horopter. Figure 5.6b is the object moving directly towards the camera. The baseline of the cameras is continuously adjusted to keep the object on the horopter. As it gets very close, the cameras are situated so that the object is still visible in both images. These simulations illustrate that the control scheme will work for

Algorithm 1 Continuous baseline positioning algorithm

```
Gather the images;
Rectify images based on the baseline;
Localize object;
if Object is localized in both images then
    Calculate Z and x_g                                  {Determines object's global position}
    Calculate B_horopter                                 {Finds the baseline to put object on the horopter}
    Calculate B_FOV                                      {Largest baseline to keep object in FOV}
    if B_horopter > B_FOV then
        if B_FOV > B_max then
            Move cameras to B_max                        {Maximum physical baseline}
        else if B_FOV < B_min then
            Move cameras to B_min                        {Minimum physical baseline}
        else
            move cameras to B_FOV
        end if
    else if B_horopter < B_FOV then
        if B_horopter > B_max then
            Move cameras to B_max                        {Maximum physical baseline}
        else if B_horopter < B_min then
            Move cameras to B_min                        {Minimum physical baseline}
        else
            move cameras to B_horopter
        end if
    end if
else if Object localized in one image then
    if Object is sufficiently far from the edge of the image then
        Move the cameras inwards
    end if
else if Object not localized then
    Move the cameras to B_max                            {Attempt to localize it in one image}
end if
```

the continuous baseline in theory. However, the system will only work as well as the object recognition and position estimates are performed. These will be discussed further in the results section.

5.3.3 Discrete Baseline

The discrete baseline control algorithm only differs from the continuous system in the situation where the object is localized in both images. Instead of placing the cameras anywhere along the length of the stereo boom, it must place it at one of the five positions that the boom was previously calibrated at. These positions are known a priori. Therefore, the boom is placed at the widest baseline such that the actual baseline is always less than or equal to the calculated baseline. This ensures that the object will retain the proper convergence angle.

The case whenever the object is only localized within one image introduces an interesting

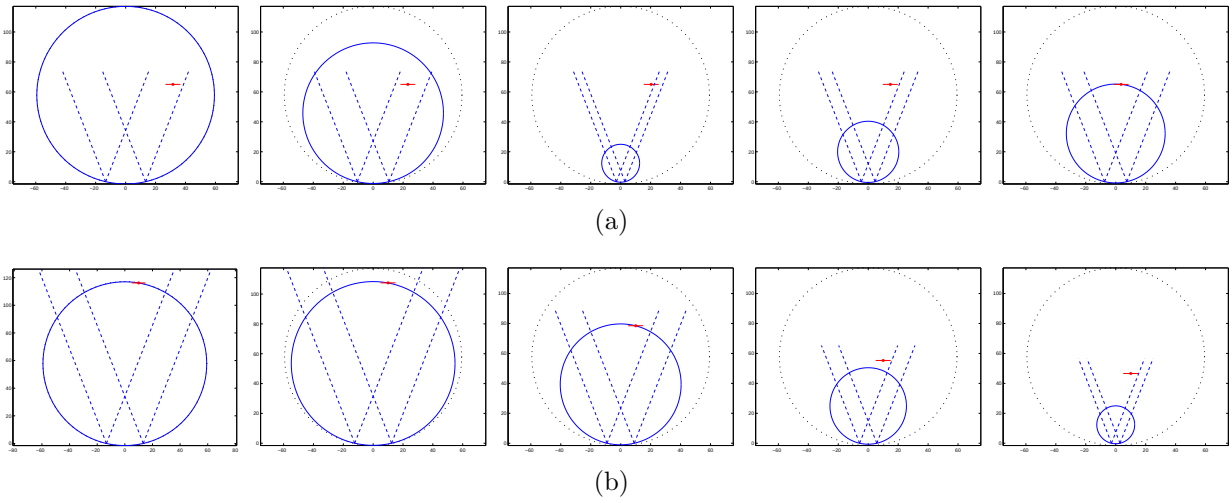


Figure 5.6: Simulations of the continuous baseline control algorithm with an object moving a) horizontally from right to left and b) towards the cameras. The red horizontal line is the object with a distinguishable point shown by a red dot. The black dashed circle is the Vieth-Muller circle from the maximum baseline and the blue circle is the current one based on the baseline. The field of view of the cameras are shown as blue dashed triangles.

scenario. That is that the boom is not necessarily positioned at one of the five calibrated positions when it is first seen in both images. This, of course, could be solved by shrinking to the next calibrated baseline position. The control algorithm can do this because it would represent the position that the boom would have been sent to any ways. The increment of shrinking is simply larger in this case than in the continuous one.

Simulations were performed to test the discrete baseline control algorithm with horizontally and forward moving objects. The set up was the same as that described in the continuous section, and the results are shown in Figure 5.7. Figure 5.7a depicts the horizontally moving object. Notice how the stereo boom stays at the same baseline through the middle section of the boom. This is because the next largest baseline would place the object within the horopter, which would result in a larger than allowable convergence angle. The approaching object is shown in Figure 5.7b. The most important thing to realize is that the object was not visible in the last image of the figure, which caused the cameras to move to their maximum baseline in order to localize it in one of their the fields of view. This algorithm works in

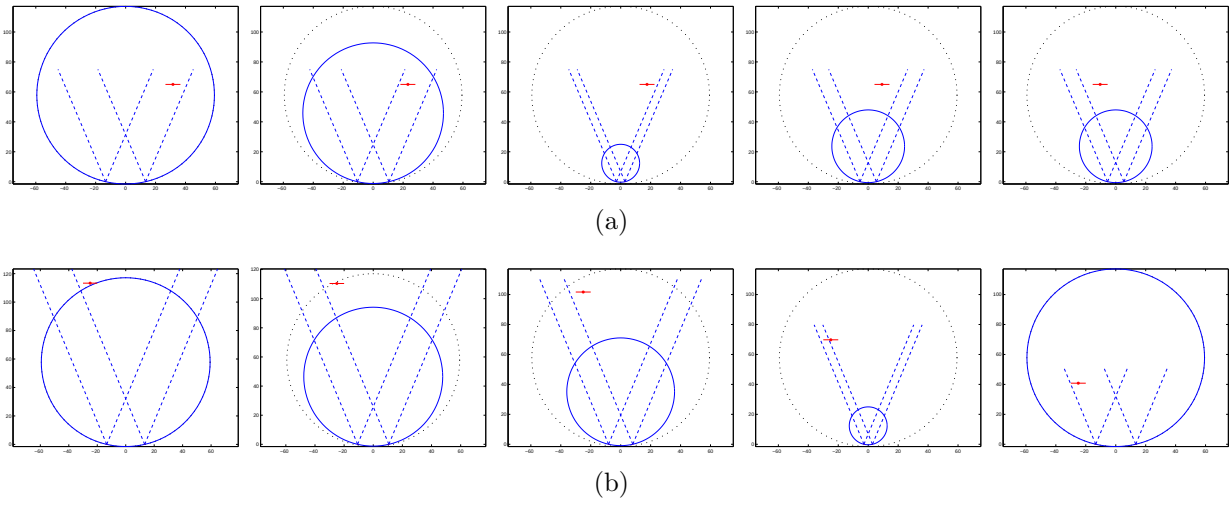


Figure 5.7: Simulations of the discrete baseline control algorithm with an object moving a) horizontally from right to left and b) towards the cameras. The red horizontal line is the object with a distinguishable point shown by a red dot. The black dashed circle is the Vieth-Muller circle from the maximum baseline and the blue circle is the current one based on the baseline. The field of view of the cameras are shown as blue dashed triangles.

theory, but again is only as effective as the position estimate and object recognition are.

5.4 Results

As was shown in the previous sections, the control algorithm functions within a theoretical framework. However, it is uncertain how well they will work on the actual stereo boom. This section will attempt to address this problem by discussing the current setup and user interface of the boom, along with results from accuracy testing of the position estimates.

5.4.1 User Interface

The current system has many components that all must be connected and powered in order for the system to work. It begins with the Arduino microcontroller. The USB port on the microcontroller must be connected to the computer that is operating the software via a USB cable. The computer will then power the device and be able to communicate with it. The cameras must also be attached to the computer by a FireWire cable. The cameras receive their power over the FireWire cable. So, the computer must have a 9-pin FireWire connection, or a powering hub must be used. In this work, the power was supplied to the cameras by an external FireWire hub. A 4-pin FireWire cable runs from the FireWire hub to the computer, which allows images to be gathered.

The control algorithm begins by ensuring that the Arduino and the cameras are connected. If either one is not, the program will exit with an error specifying that the components must be connected. Once the program verifies that everything is attached and properly communicating, it will prompt the user for an exposure setting. The values between 1 and 1000 are valid exposure settings. If a negative value is inputted, the cameras will be set to the auto-exposure mode. The cameras are then given the command to zero themselves. They drive outward until the limit switch is hit.

At this point, images are gathered and presented to the user. The user is prompted to place a bounding box around the object of interest. This is done by left-clicking and dragging a box around the object. If the object is not visible in an image, the user can right-click on the image. The box and visibility can be continually adjusted until any key is struck. For instances where the object was not detected at all, the cameras are sent to their maximum baseline and images are regathered. The baseline will be shrunk if the object was detected in only one image. If the object is visible in both frames, the program will prompt the user to click on a distinctive point within the bounding box, that is visible from both cameras. This will determine the depth to the object, and then the controller will move the cameras according to the estimate of the global position of the object.

5.4.2 Position Estimation Accuracy

The control algorithm is heavily dependent on the accuracy of the global position estimates. Therefore, it is important to take a look at how well they are estimated. This was performed by placing the camouflaged box at different x and z distances away from the cameras. The box was placed so that the face was parallel to the image planes, and an 'X' was placed on the box to provide a distinguishable point in both images. 'X' marks the spot, that the user must click to gather the distance estimate. This shape made it easier to pick the same spot repeatably. The position was estimated with the box at $x_g = 0$, and also at the edge of the field of view of the cameras. The centerline that represented $x_g = 0$ was aligned perpendicular to the stereo boom, and was marked by a string. This test would demonstrate the effect that object location has on the position estimates. The z distance to the box was varied through every 12 inch interval between 84 and 144 inches. The minimum distance was chosen because the object gets harder to see at the maximum baseline if it gets any closer. 144 inches was set as the maximum distance because this is a little bit outside of the largest horopter. For objects further away than this, the baseline would be the maximum, unless the object is near the edge of one of the images. Since small baselines may be used to

estimate large distances, it is also important to see how well the position is estimated with different baselines. Each box location was estimated with the cameras at different baselines to see how well they performed.

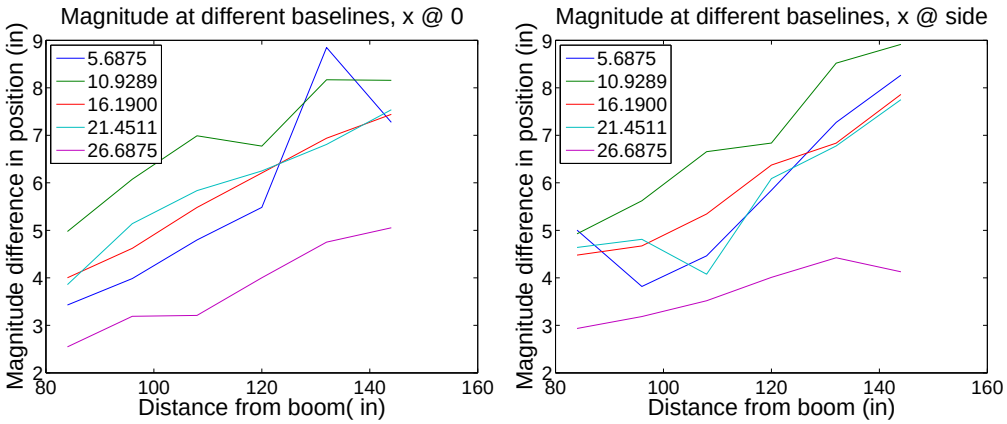
The results were gathered and the magnitude difference between the estimated position and the experimental position was calculated. The magnitude difference included the difference in both the x and z directions, and was used as an estimation metric because the x position depends directly on the z position. Therefore, if the z estimate is off, the x estimate will also be off, which means that the magnitude can better describe the accuracy of the estimate. The estimated position that was gathered from the experiment will be represented as z_{est} and x_{est}.

The results have been plotted in Figure 5.8. Figure 5.8a depicts the magnitude difference for the five calibrated baselines with the box at the middle location. There does not appear to be any relationship between the baseline and the amount of error. If the error would have decreased with an increasing baseline, it would of been thought that there was a relationship. However, this is not the case. The difference at the minimal baseline is smaller than some of those at the larger baselines. This is opposite from what was expected, because the larger baselines should result in a smaller error. The distance resolution increases with the baseline, so the error should be reduced. The maximum baseline supports this notion since its magnitude errors are well below those of the other baselines, but none of the others really do. It is believed that these results stem from poor calibrations at those positions. More tests with different calibrations would need to be performed in order to determine whether this was true. Unfortunately, time did not permit this. These tests are very time consuming because the rig must be calibrated at each of the baselines, the box must be imaged at all of the locations, and the user must select the object, all of which add time.

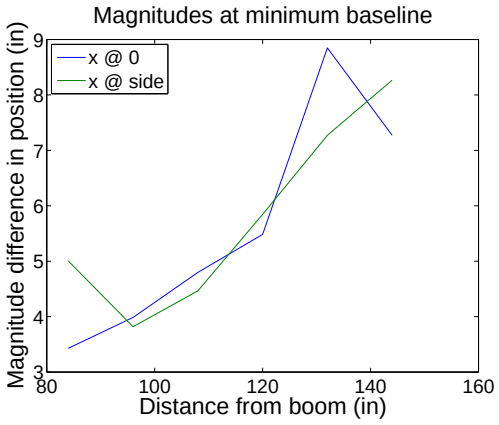
Comparing the results from the middle and side, Figures 5.8a and b, it appears that the position does not influence the error significantly. This result was expected, because the Z error is not dependent on x or y. Figure 5.8c shows this comparison distinctly for the mini-

mum baseline. Notice that the magnitude difference at both x locations are approximately the same.

One interesting thing that presented itself in the data was that there was a fairly consistent offset in x_{est} across baselines for tests at the same experimental z distance, z_{exp}. This suggests that the experimental centerline was not the actual centerline and was instead offset by some angle. It therefore becomes a little more informative to normalize the z_{est} based on the experimental x positions and not the estimated ones. This would, essentially, align the



(a) (b)



(c)

Figure 5.8: Magnitude difference between the actual and estimated positions of the box at the calibration locations. a) is the differences at $x_g = 0$, b) are those at the sides of the images, and c) is a comparison of differences at the middle and the side for the minimum baseline.

centerline so that the experimental and observed centerlines coincided. There are errors associated with x_{est}, which makes this technique imperfect, but it will provide some insight. The normalized z distance, z_{norm}, can be determined through the Pythagorean theorem because the experimental magnitude and x position are known. Figure 5.9 illustrates the geometry of the experiment, which may help to alleviate any confusion. With the data normalized, the amount of error in the z direction could be represented as the number of pixels that the disparity may have been incorrect by, in order to cause that amount of z error. The expected resolution is known at the different z_{exp}. Therefore, the number of pixels can be calculated by dividing z_{norm} by the expected resolution at that distance. The average deviation from all distances at different baselines are shown in Table 5.1 in terms of number of pixels.

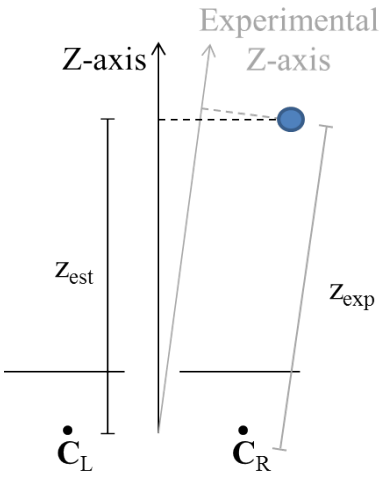


Figure 5.9: Geometry of the experiment for determining the position estimation accuracy

The results at $x_g = 0$ all have a difference that is less than two pixels. This is acceptable because the data relied on user input, which was not perfect and could have been off. Two pixels seems like that would be a reasonable amount that the disparity could change based on user error. Some of the images for the object at the side have resulted in greater than two pixels of error. This implies that something else is occurring.

It is not very definitive as to what caused these discrepancies, but there are a few

possibilities. It could be that the undistortion of the images has skewed the pixels enough to cause the two representations of the object to appear differently. Distortion is most prevalent at the edges of the images, which means that those pixels require the most undistortion and are therefore most different from their observed state. Another possible explanation is that the objects at these distances created such a great convergence angle that the objects did not appear the same. This is a distinct possibility because the problems seem to have occurred mainly at the larger baselines, where the difference is most pronounced. Also, for the most part, the error decreases as z_{exp} increases, which would lend to further corroborating this claim. The orientation of the box at these convergence angles would cause it to appear more different at the sides, and result in the user selecting two different locations. The convergence angle does not, however, explain the pixel error from the side at the minimum baseline, which is nearly double the error from that of $x_g = 0$. Lastly, the error in the x direction may be increasing the pixel error. The x position is not known as precisely at the sides of the baseline (recall Figure 2.3 on page 10) as in the middle. This would result in the magnitude being more incorrect at the sides. It is believed that all of these play a minor role in producing the observed error.

Table 5.1: Differences in magnitude

Baseline	**Average z_{norm} over Experimental Magnitude, $x_g = 0$**	**Average Number of Pixels**
5.69	1.018	0.762
10.93	1.008	0.804
16.19	0.995	0.673
21.45	0.998	0.932
26.69	0.995	1.387
Baseline	**Average z_{norm} over Experimental Magnitude, side**	**Average Number of Pixels**
5.69	1.028	1.435
10.93	1.000	0.866
16.19	0.996	0.908
21.45	0.989	2.327
26.69	0.987	3.115

Although there are some unknown errors occurring with the position estimates, for the most part, they are pretty accurate. There do not appear to be any extremely wrong estimates, which suggests that the discrete baseline variant could be used effectively. The small errors that are occurring will not affect the positioning of the cameras greatly, unless the object is very near to the edge of a horopter. However, errors of this nature may make the continuous baseline system a little spastic. If the position estimates are not the same over time, the cameras will continue to move in an oscillatory manner. This would cause unwanted wear on the system.

The continuous baseline algorithm would perform better if additional steps were taken. Increasing the resolution of the images to the full resolution of the cameras would improve the positioning estimates by approximately double. Since there are only 850 possible baselines, the increase in the resolution could potentially reduce the error of one measurement to below that of its nearest neighbors. This would help to ensure that the boom stays at the proper baseline, making it less apt to moving erroneously. Another possible solution is to perform some sort of low pass filtering. In other words, if the boom keeps moving the cameras quickly back and forth for no apparent reason, the filter could force it stay at one of those positions.

5.4.3 Object Recognition Problems

In order to make the stereo boom adjust fully autonomously, there must be some sort of object recognition to provide information about an objects whereabouts. Although the object recognition is currently being performed by the user, it is important to illustrate where some of the systems shortcomings lie, so that they could be fixed. Both the continuous and discrete baseline control algorithms, assume that the objects are recognized perfectly, which is not realistic. Object recognition is prone to many different errors, and these would affect the algorithms. These include false positives, when the wrong object is localized; false negatives, when the object is unable to be localized, although it is within the scene; and improper bounding box locations, correctly finding the object, but not recognizing all of

it. These would wreak havoc on the control algorithm. The false positives would cause the the boom to gather improper distance information about the object. False negatives would cause the system to move either outward or inward, which may cause the object to be lost from sight. Similarly, the improper bounding box locations may result in the object leaving the field of view.

It may be possible to reduce some of these problems by adding filtering. A Kalman filter could be used to keep track of the previous estimates of the object's location in 3D space, and estimate where it should appear in the future. This could greatly alleviate some of these problems by not relying solely on the object recognition. This would allow the object detection technique to occasionally fail, while still properly positioning the cameras.

5.4.4 Implementation

Since the position estimates are fairly accurate, the system was tested using the discrete baseline control algorithm. The system worked properly. The user was able to localize the camouflaged box in the images, and then the cameras were adjusted accordingly. The box was moved towards and away from the boom, and then horizontally across the field of view, mimicking the tests performed in the theoretical section. The boom worked as expected.

Chapter 6

Conclusions

This thesis attempted to uniquely overcome one shortcoming pertaining to conventional stereo vision systems. These conventional stereo systems are limited by the fact that they have a fixed baseline. A large baseline can provide better depth estimates of an environment, but fails to capture objects whenever they get too close to the cameras. This limitation can be overcome by utilizing a smaller baseline. However, the smaller baseline can not gather as accurate depth information. Therefore, an automatically adjusting variable baseline system has been developed.

The variable baseline stereo boom is comprised of two cameras that are mounted on linear bearings. These bearings allow the cameras to move freely along the length of the boom, thereby adjusting the baseline. A gearing rack was attached to the each of the cameras. This allowed a stepper motor, in conjunction with a pinion gear, to simultaneously position both cameras at a large array of different baselines. The cameras are capable of traversing baselines between 4.7 and 26.7 inches. The entire system was placed within a 3.5 inch aluminum tube to both protect the cameras, and to provide some structural stability. By utilizing aluminum tubing, it also kept the system lightweight. The entire structure weighs a total of approximately 5.4 pounds, not including some electrical components.

The stereo boom was designed in order to use one stereo calibration result for all of the different baselines. A sensitivity analysis determined the allowable calibration errors in terms of roll, pitch, yaw, and baseline between cameras that would make that possible. They were minuscule. The sensitivity analysis used uncoupled equations in order to calculate these parameters. Since the errors were so small, a finite element analysis (FEA) was required to determine whether or not the design would exceed them. The analysis was performed first at 68°F with two fixed mounts, and the errors were not surpassed. The results were rerun at 0°F and 100°F, because these are possible temperatures for an operating environment. Unfortunately, strains induced by the temperature loading caused the boom to deflect greater than the allowable amount. This required that the mounts were changed to one pinned and one sliding joint, which could represent a vibration isolation system. The FEA was re-administered with the new configuration and the boom met the deflection requirements. The boom was then manufactured to the design specifications.

Although the boom should theoretically operate with one calibration, it is important to determine if it could physically. In order to do this, the boom was calibrated at the middle baseline. This calibration was then used to rectify images from the middle, maximum, and minimum baselines into the frontal parallel position. SIFT features were matched between image pairs in order to determine whether the allowable vertical offset between images had been maintained. It had not, which meant that one calibration file could not be used over the entire length of the boom. This could be an artifact of the uncoupled sensitivity equations, the manufacturing tolerances, or play in the bearings. The cameras were then calibrated at five equally spaced locations along the length of the stereo boom, and the relationships between calibrations were observed. It appears that linear interpolation could be performed between the calibration results to determine the extrinsics at all positions along the boom.

This analysis was followed by a derivation and explanation of the positioning control scheme, which is capable of automatically adjusting the baseline based on the location of an object. The positioning algorithm depends heavily on the failures of semi-global block matching(SGBM). SGBM requires that the same objects be localized in both scenes. How-

ever, as an object approaches the cameras, the the object looks more and more dissimilar between the images. Eventually, the SGBM technique fails to find correspondences between the stereo image pairs and can not create 3D reconstructions. The point of failure defines an angle between the cameras and the object of interest, which is known as the maximum convergence angle. The maximum convergence angle carves out a circle that passes through the camera centers, known as the Vieth-Muller circle. The control scheme ensures that the object of interest never enters the Vieth-Muller circle, which should allow full stereo reconstructions to be created.

The algorithm begins by having a user localize an object of interest in the images. If it is located in both images, the 3D position of the object is calculated through triangulation. The baselines that would place the object on the circle, and that which would keep the object within the field of view of both cameras are then calculated. The cameras are placed at the smaller of the two baselines so that the maximum convergence angle is not exceeded, and that the object can still be seen. If the object is localized in one of the images, the cameras are brought closer together in an attempt to make the object visible in both. The case were it is not found in either, places the cameras at the maximum baseline so that the object can hopefully be localized in one. There are two variants of the control algorithm, continuous and discrete. The continuous variant assumes that either linear interpolation between calibrations or self-calibration is being performed. It places the cameras at any location along the length of the boom. Contrarily, the discrete version places the cameras at one of the five calibrated baselines. The baseline which is chosen is the largest baseline that does not exceed those that were calculated earlier.

Both of the algorithms rely on the position estimates of the the object of interest. Therefore, it becomes important to determine how accurate the estimates are. The positions were estimated at different distances from the camera, and offsets from the middle of the baseline. This test was run at the five different calibrated baselines. The results suggested that neither the offset, nor baseline affects the estimation error. This was expected for the offset, but not for the baseline. It is believed that the results for different baselines deviated

from the expectation because of errors in the calibration procedure. More tests would need to be performed in order to determine this concretely.

In the end, the position estimates for the system were adequate. They were good enough to operate the boom with the discrete control algorithm, but not necessarily with the continuous one. The end result of all the work was a stereo vision system that could automatically adjust its baseline based on position estimates of an object. This alleviates some of the shortcomings of conventional stereo cameras, and allows objects that are both near and far to be accurately localized in 3D.

Chapter 7

Future Works

Although the automatically positioning variable baseline stereo boom works fairly well, there are still areas that could use improvement. This section will introduce some ideas that would make future implementations even better.

7.1 Design Improvements

The mechanical design could have been improved in various ways, some of which have already been presented and will not be shown again. Instead, only additional improvements will be discussed in this section.

One change that would be simple and yet very useful, would be to add a hard plastic cover above the lenses. This would serve two purposes. It would be able to shade the lenses, reducing the effects of glare on the images, and also provide some additional protection. The covers could be mounted directly above the front slots in the boom.

Another improvement would be a plastic sheath that fits snugly around the cameras, and fills the slots. This would keep out dirt from the interior of the system. On a construction site, there may be lots of particulate that could get within the current open design, and

could cause the bearings to seize up.

To help further prevent seizing, a more powerful stepper motor is recommended. The current one can adjust the cameras, however, it is operating at the edge of its limitation. If dirt would happen to get into the system, it might inhibit the motor from moving the cameras, which would be catastrophic.

As a further safety measure, it may be wise to add an additional limit switch to the interior of the boom, or a positional feedback sensor. These would stop the cameras from contracting too much, which may occur if the motor skips steps or the gearing slips. In its current state, the boom has the ability to loose track of its global position and to cause harm to the cameras.

7.2 Object Detection

In the current framework, a user must localize the object in every frame. It would be better if the software would localize the object automatically. This would make the boom completely autonomous. There are different ways that this could be done. The one that may be the most useful would follow the framework described by Kalal et. al. in [40]. This system first prompts the user to select the object of interest in an image. It then performs many affine transformations on the object to estimate what the object looks like in other orientations. The algorithm uses a binary classifier and bootstrapping technique to estimate the position of the object in subsequent frames. If the object is detected and the vantage point of the object is unique, then the new orientation is learned and added to the library of what the object looks like. This allows the algorithm to learn and more effectively detect objects the longer it runs. The algorithm is known as Tracking-Learning-Detection approach, and also as the Predator algorithm. This system would allow the user to select an object that they are interested in, and have the boom adjust its baseline automatically. The Predator algorithm can operate in real time, which would make the entire process faster.

7.3 Stereo Vision

SGBM is currently being used to create 3D terrain maps. However, it is not the most effective, nor is it the fastest, stereo correspondence algorithm. Therefore, it would be advantageous to use one of the better algorithms that were presented in Chapter 2. These would run faster and would create better maps of the environment. It is important to realize that if the stereo algorithm changes, it affects many aspects of this work. These include the sensitivity analysis and the convergence angle. Those experiments would have to be rerun in order to determine what they are for the new stereo algorithm.

Another possible improvement is to allow the stereo maps to be additive. Some technique, possibly an iterative closest point(ICP) technique, could match 3D information between the stereo reconstructions to create one global map of the environment. This would be preferred over having many unrelated point clouds, but presents an interesting problem. The 3D information gathered from the short baselines have lower accuracy than that from the larger baselines. This may make the maps look strange. It may be useful to use a technique similar to that in [41] to better estimate the environment. They use probabilistic techniques to estimate the positions of 3D points that are both near and far simultaneously.

7.4 Calibration

Lastly, something should be done to improve the stereo calibrations. A self-calibration technique would help with this, but would be even more computationally costly. The object recognition, stereo vision, and ICP will need most of the processing. It may be better to, instead, understand the relationship between different calibrations along the length of the boom more completely. This would allow a priori calibrations to be used, and would keep the computational cost to a minimum.

Bibliography

[1] J. Y. Bouquet. Camera Calibration Toolbox for Matlab. California Institute of Technology. (2012, May) Software. [Online]. Available: http://www.vision.caltech.edu/bouguetj/calib$_$doc/

[2] R. Hartley and A. Zisserman, *Multiple View Geometry in computer vision*, 2nd ed. Cambridge University Press, 2003.

[3] G. Bradski and A. Kaehler, *Learning OpenCV*. O'Rielly Media, Inc., 2008.

[4] L. Heng, G. H. Lee, F. Fraundorfer, and M. Pollefeys, "Real-time photo-realistic 3D mapping for micro aerial vehicles," *IEEE International Conference on Intelligent Robots and Systems*, pp. 4012–4019, 2011.

[5] OpenCV Software and Documentation. Willow Garage. (2012, June) Software. [Online]. Available: http://sourceforge.net/projects/opencvlibrary/files/

[6] Z. Zhang, "A Flexible New Technique for Camera Calibration," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, pp. 1330–1334.

[7] D. Scharstein and R. Szeliski. The Middlebury Computer Vision Pages. Middlebury College. (2012, May). [Online]. Available: http://vision.middlebury.edu/

[8] H. Hirschmüller, "Accurate and Efficient Stereo Processing by Semi-Global Matching and Mutual Information," *IEEE Conference on Computer Vision and Pattern Recognition*, vol. 22, June 2005.

[9] J. Gassaway, "Local Bundling of Disparity Maps for Improved Dense 3D Visual Reconstruction," Master of Science Thesis, Mechanical Engineering Dept. of Virginia Tech, Blacksburg, VA, 2011.

[10] S. Birchfield and C. Tomasi, "A pixel dissimilarity measure that is insensitive to image sampling," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 1998.

[11] Q. Yang, L. Wang, R. Yang, H. Stewnius, and D. Nistr, "Stereo matching with color-weighted correlation, hierarchical belief propagation, and occlusion handling." *IEEE*

Transactions on Pattern Analysis and Machine Intelligence, vol. 31, no. 3, pp. 492–504, 2009.

[12] X. Mei, X. Sun, M. Zhou, S. Jiao, H. Wang, and X. Zhang, "On building an accurate stereo matching system on graphics hardware," in *Computer Vision Workshops (ICCV Workshops), 2011 IEEE International Conference on*, Nov 2011, pp. 467 –474.

[13] E. Rosten and T. Drummond, "Machine learning for high-speed corner detection," *In European Conference on Computer Vision*, 2006.

[14] E. Rublee, V. Rabaud, K. Konolige, and G. Bradski, "ORB: an efficient alternative to SIFT or SURF," *Computer Vision (ICCV), 2011 IEEE International Conference*, November 2011.

[15] D. G. Lowe, "Distinctive Image Features from Scale-Invariant Keypoints," *International Journal of Computer Vision*, vol. 60, pp. 91–110, 2004.

[16] H. Bay, T. Tuytelaars, and L. V. Gool, "Surf: Speeded up robust features," in *In ECCV*, 2006, pp. 404–417.

[17] J. Heikkila and O. Silven, "A four-step camera calibration procedure with implicit image correction," in *Computer Vision and Pattern Recognition, 1997. Proceedings., 1997 IEEE Computer Society Conference on*, June 1997, pp. 1106 –1112.

[18] D. Nister, "An efficient solution to the five-point relative pose problem," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 26, no. 6, pp. 756 –770, June 2004.

[19] T. Dang, C. Hoffmann, and C. Stiller, "Continuous stereo self-calibration by camera parameter tracking," *IEEE Transactions on Image Processing*, vol. 18, no. 7, pp. 1536–1550, 2009.

[20] N. Ahuja and A. Abbott, "Active stereo: integrating disparity, vergence, focus, aperture and calibration for surface estimation," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 15, no. 10, pp. 1007 –1029, October 1993.

[21] W. Klarquist and A. Bovik, "FOVEA: a foveated vergent active stereo vision system for dynamic three-dimensional scene recovery," *Robotics and Automation, IEEE Transactions on*, vol. 14, no. 5, pp. 755 –770, October 1998.

[22] S. Das and N. Ahuja, "Performance Analysis of Stereo, Vergence, and Focus as Depth Cues for Active Vision," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 17, no. 12, pp. 1213–1219, December 1995.

[23] D. W. Murray, F. Du, P. F. McLauchlan, I. D. Reid, P. M. Sharkey, and M. Brady, "Active vision," A. Blake and A. Yuille, Eds. Cambridge, MA, USA: MIT Press, 1993, pp. 155–172.

[24] A. Miller, P. Allen, and D. Fowler, "In-vivo stereoscopic imaging system with 5 degrees-of-freedom for minimal access surgery," *Studies in health technology and informatics*, pp. 234 – 240, 2004.

[25] T. Hu, P. Allen, T. Nadkarni, N. Hogle, and D. Fowler, "Insertable stereoscopic 3D surgical imaging device with pan and tilt," in *2nd IEEE RAS EMBS International Conference on*, October 2008, pp. 311 –316.

[26] F. Rovira-Ms, Q. Wang, and Q. Zhang, "Bifocal Stereoscopic Vision for Intelligent Vehicles," *International Journal of Vehicular Technology*, 2009.

[27] D. Gallup, J.-M. Frahm, P. Mordohai, and M. Pollefeys, "Variable baseline/resolution stereo," in *Computer Vision and Pattern Recognition*. IEEE Computer Society, 2008.

[28] Y. Nakabo, T. Mukai, Y. Hattori, Y. Takeuchi, and N. Ohnishi, "Variable Baseline Stereo Tracking Vision System Using High-Speed Linear Slider." in *International Conference on Robotics and Automation*. IEEE, April 2005, pp. 1567–1572.

[29] Everything about Equipment. Ritchie Specs. (2012, March) Software. [Online]. Available: http://www.ritchiespecs.com/specification?type$=&$category$=$Hydraulic$+$Excavator$&$make$=$Caterpillar$&$model$=$345C+L$&$modelid$=$92300

[30] Online Materials Information Resource. MatWeb. (2012, March). [Online]. Available: http://www.matweb.com/

[31] Plastic Bearings. Igus. (2010) Sales Catalog. pp 20.7.

[32] D. L. Logan, *A First Course in the Finite Element Method*, 4th ed. O'Rielly Media, Inc., 2007, pp. 367,507,515.

[33] R. G. Budynas and J. K. Nisbett, *Shigley's Mechanical Engineering Design*, 8th ed. McGraw-Hill Companies, Inc., 2008.

[34] W. F. Riley, L. D. Sturges, and D. H. Morris, *Statics and Mechanics of Materials*, 2nd ed. John Wiley & Sons, Inc., 2002, p. 125.

[35] JMP 9.0. JMP. Statistical Software.

[36] R. L. Ott and M. Longnecker, *An Introduction to Statistical Methods and Data Analysis*, 6th ed. Brooks/Cole, Cengage Learning, 2010, ch. 5,9,11.

[37] T. G. Beckwith, R. D. Marigold, and J. H. Lienhard, *Mechanical Measurements*, 6th ed. Pearson Education, Inc., 2007, p. 128.

[38] B. Jähne and H. Haußecker, *Computer Vision and Applications: A Guide for Students and Practitioners*. Academic Press, 2000, ch. 11.

[39] I. P. Howard and B. J. Rogers, *Binocular Vision and Stereopsis*. Oxford University Press, 1995, ch. 2.

[40] Z. Kalal, J. Matas, and K. Mikolajczyk, "P-N learning: Bootstrapping binary classifiers by structural constraints," in *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*, June 2010, pp. 49 –56.

[41] Y.-C. Lim, C.-H. Lee, S. Kwon, and W.-Y. Jung, "Distance estimation algorithm for both long and short ranges based on stereo vision system," in *Intelligent Vehicles Symposium, IEEE*, June 2008, pp. 841 –846.

Appendix A

Vieth-Muller Circle Proof

This section will prove that the shape inscribed by a constant convergence angle upon an object can only be a circle that passes through both camera centers. The proof will begin by assuming nothing about the shape other than it maintains a constant convergence angle. Since the shape will account for the object at different locations, the position of the object will not always be located directly in the middle of the baseline. That means that the distance to the object must change in order to keep the convergence angle the same. It will create a shape that will ensure that the angle be kept. Figure A.1 shows the angles of the triangle for the object at a different position.

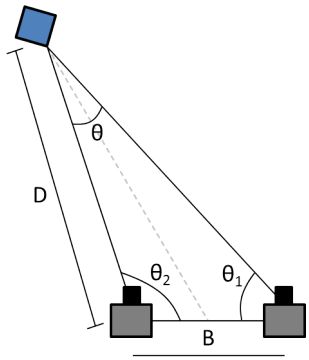


Figure A.1: Convergence angle on an object that is on the side

From the image and the law of sines, we know that

$$\frac{\sin\theta_1}{D} = \frac{\sin\theta}{B} \quad \Rightarrow \quad D = \left(\frac{B}{\sin\theta}\right)\sin\theta_1 \tag{A.1}$$

where D is the length of the side opposite from θ_1, B is the baseline distance which will be a constant for the sake of determining the shape, and θ is the convergence angle which is also a constant. Therefore, taking the derivative of D with respect to θ_1 gives

$$\frac{dD}{d\theta_1} = \left(\frac{B}{\sin\theta}\right)\cos\theta_1 \tag{A.2}$$

Setting equation A.2 equal to zero and solving for θ_1 provides the angle at which D is maximum. The maximum length of D is located at a θ_1 of 90 degrees. The same steps can be performed with θ_2, as well, and proves that the shape must be symmetric about the middle of the baseline. The maximum D for both angles can be found by substituting 90° into equation A.1, and it turns out that they are the same. Since both of the lengths for θ_1 and θ_2 are at 90 degrees and lengths are the same, they form the same baseline at the top. This means that the shape must also be symmetric about a horizontal axis that passes through the intersection of both maximum lengths. This intersection point will be referred to as the center point and half of the maximum distance will be referred to as r. So,

$$r = \frac{B}{2\sin\theta} \tag{A.3}$$

It is still uncertain what the full shape is, but it appears that it would be a circle. In order to prove that it is indeed a circle, the problem must be formulated in terms of distances from the center point to the edge of the object. Figure A.2 shows the geometry and nomenclature that are used in proving the shape. The point, C, is the center point of the shape, which means that l_{EC} and l_{FC} are equal to r. The variable, l, refers to the length of an edge of the triangle while the subscript describes which edge the length corresponds to. This means that l_{EC} is the length of the edge between point E and C. Similarly, angles are denoted

by $\angle$, and triangles are denoted by $\triangle$. From similar triangles, we know that half of $\angle ECF$ must be equal to the convergence angle, θ.

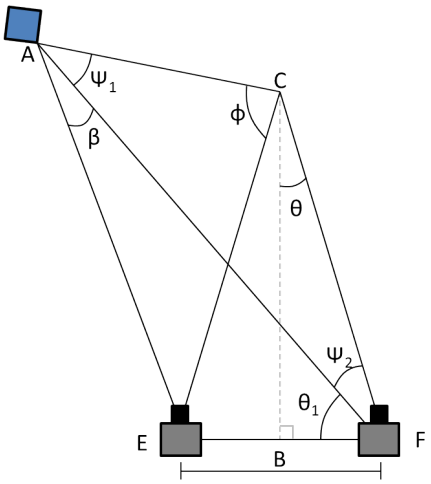


Figure A.2: Angle nomenclature for the object at different locations

The proof for the circular shape assumed that l_{AC} is equal to r in an attempt to prove that

$$\text{If: } l_{AC} = r \quad \Rightarrow \quad \beta = \theta \tag{A.4}$$

which will show that the convergence angle remains constant so long as the shape is a circle. The proof begins by taking the law of cosines for $\triangle ACE$ about ϕ, which gives

$$\begin{aligned} l_{AE}^2 &= l_{AC}^2 + l_{CE}^2 - 2 l_{AC} l_{CE} \cos\phi \\ l_{AE}^2 &= r^2 + r^2 - 2r^2 \cos\phi \\ l_{AE} &= r\sqrt{2\left(1 - \cos\phi\right)} \end{aligned} \tag{A.5}$$

Observing that l_{AC} and l_{FC} are the same, which was assumed to be true, the law of sines proves that both Ψ_1 and Ψ_2 are the same. A relationship can be formed between the angles comprising $\triangle ACF$. All of the angles of the triangle must sum to 180 degrees. Letting $\Psi_1 = \Psi_2 = \Psi$ leads to

$$\Psi = 90 - \frac{\phi + 2\theta}{2} \tag{A.6}$$

θ_1 can be determined by creating a right triangle between C, F, and the middle of the baseline.

$$\begin{aligned} \theta_1 &= 90 - \theta - \phi \\ \theta_1 &= \frac{\phi}{2} \end{aligned} \tag{A.7}$$

The law of sines of the $\triangle AEF$ about θ_1 and β gives

$$\begin{aligned} \frac{\sin\beta}{l_{EF}} &= \frac{\sin\phi/2}{\sqrt{2r^2\left(1-\cos\phi\right)}} \\ l_{EF} &= \frac{r\sin\beta\sqrt{2\left(1-\cos\phi\right)}}{\sin\phi/2} \end{aligned} \tag{A.8}$$

Since l_{EF} is equal to the baseline, we know that $l_{EF} = 2r\sin\theta$. Substituting into equation A.8, leads to

$$\begin{aligned} 2r\sin\theta &= \frac{r\sin\beta\sqrt{2\left(1-\cos\phi\right)}}{\sin\phi/2} \\ 2r\sin\theta\sin\phi/2 &= r\sin\beta\sqrt{2\left(1-\cos\phi\right)} \\ 2\sin^2\theta\sin^2\phi/2 &= \sin^2\beta\left(1-\cos\phi\right) \end{aligned} \tag{A.9}$$

Using the half-angle formula leaves

$$\begin{aligned} \sin^2\theta\left(1-\cos\phi\right) &= \sin^2\beta\left(1-\cos\phi\right) \\ \sin^2\theta &= \sin^2\beta \end{aligned} \tag{A.10}$$

which can only be true for $\beta = \theta$. Therefore, a circle passing through the camera centers will ensure that a constant convergence angle is maintained. We know that $D = l_{AE}$, which indicates that the shape must be unique. Since it has been proven that a circle works, it requires that the only shape that keeps a constant convergence angle be a circle. This circle will be referred to as the Vieth-Muller circle, and is shown in Figure 5.4 (pg 75).