

THE HENRY EDWIN SEVER GRADUATE SCHOOL  
MASTER OF SCIENCE DEGREE

THESIS ACCEPTANCE

(To be the first page of each copy of the thesis)

DATE: July 25, 2004

STUDENT'S NAME: Roman T. Stanchak

This student's thesis, entitled A 3D Laser Targeting System has been examined by the undersigned committee of five faculty members and has received full approval for acceptance in partial fulfillment of the requirements for the degree MASTER OF SCIENCE.

APPROVAL: \_\_\_\_\_ Chairman  
\_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_

Short Title: 3D Laser Targeting

Stanchak, M.Sc. 2005

WASHINGTON UNIVERSITY  
THE HENRY EDWIN SEVER GRADUATE SCHOOL  
DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

---

A 3D LASER TARGETING SYSTEM

by

Roman T. Stanchak, B.S.

Prepared under the direction of Professor Robert Pless

---

A thesis presented to the Henry Edwin Sever Graduate School of  
Washington University in partial fulfillment  
of the requirements for the degree of

MASTER OF SCIENCE

August 2005

Saint Louis, Missouri

WASHINGTON UNIVERSITY  
THE HENRY EDWIN SEVER GRADUATE SCHOOL  
DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

---

ABSTRACT

---

A 3D LASER TARGETING SYSTEM

by

Roman T. Stanchak

ADVISOR: Professor Robert Pless

---

August 2005

Saint Louis, Missouri

---

We present a 3D laser targeting system. The goal of the system is to aim a laser pointer at a specific point in the environment using observations from a stereo camera. Control systems of this type are typically closed-loop feedback architectures that require accurate knowledge of where the laser pointer is in the world. To avoid this detection step, we present two algorithms to solve for the direct transformation between a target point in 3D coordinates and the controls required to move the laser to that point. One algorithm solves for the transformation matrix directly using corresponding laser and 3D camera coordinates. The other utilizes the epipolar constraint between laser coordinates and 2D image coordinates to solve for the transformation. Although these algorithms are adapted from camera calibration literature, our application of them to this task is novel. In addition to these algorithms, we discuss the implementation of an automated calibration strategy designed to minimize operator intervention. Experimental results verify our approach and show the two algorithms comparable in accuracy. Our automated calibration strategy is shown to be comparable to a manual calibration.

# Contents

List of Tables . . . . .	iv
List of Figures . . . . .	v
List of Algorithms . . . . .	vi
Acknowledgments . . . . .	vii
<b>1 Introduction . . . . .</b>	<b>1</b>
<b>2 Related Work . . . . .</b>	<b>3</b>
<b>3 Background . . . . .</b>	<b>5</b>
3.1 Laser Model . . . . .	5
3.2 Camera . . . . .	6
3.3 Multiple Cameras . . . . .	7
<b>4 Laser Calibration . . . . .</b>	<b>10</b>
4.1 Direct Calibration . . . . .	10
4.2 Epipolar Calibration Method . . . . .	12
<b>5 Implementation . . . . .</b>	<b>14</b>
5.1 Laser Unit . . . . .	14
5.2 Stereo Camera . . . . .	17
5.3 Calibration . . . . .	17
5.3.1 Forward Method . . . . .	18
5.3.2 Inverse Method . . . . .	18
5.3.3 Red Dot Detection . . . . .	18
5.3.4 Point Selection . . . . .	19

<b>6</b>	<b>Experimental Results</b>	<b>21</b>
6.1	Calibration Procedure	21
6.2	Measurement Procedure	21
6.3	Results	22
6.3.1	Error w.r.t the number of calibration planes.	22
6.3.2	Error w.r.t. calibration grid size.	23
6.3.3	Error w.r.t. calibration point density.	24
6.3.4	Direct vs. Epipolar method.	25
6.3.5	Manual vs. Automatic correspondence	26
6.3.6	Error wrt. to depth of targeted point	27
6.3.7	Error wrt. to horizontal offset of targeted point	28
<b>7</b>	<b>Conclusion</b>	<b>29</b>
	<b>References</b>	<b>30</b>
	<b>Vita</b>	<b>34</b>

# List of Tables

4.1 Comparison of requirements for laser calibration methods . . . . . 13

# List of Figures

1.1	Laser Targeting System . . . . .	1
3.1	Laser coordinate system . . . . .	5
3.2	Pinhole perspective projection camera model . . . . .	6
3.3	Multiple cameras . . . . .	8
3.4	Epipolar constraint . . . . .	9
5.1	Polytec Laser with Tyzx Stereo Camera . . . . .	14
5.2	Diagram of laser mirror layout. . . . .	15
5.3	Distortion in laser control. . . . .	16
5.4	Tyzx stereo camera housing. . . . .	17
5.5	Laser artifacts in camera image . . . . .	19
5.6	Corner detection result for stereo image pair. . . . .	20
6.1	Average error vs. the number of calibration planes. . . . .	22
6.2	Average error vs. size of the calibration grid. . . . .	23
6.3	Average error vs. the number of points per calibration plane . . . . .	24
6.4	Average error for Direct vs. Epipolar calibration methods . . . . .	25
6.5	Average error for Manual vs. Automatic point correspondence. . . . .	26
6.6	Average error vs. depth of targeted point . . . . .	27
6.7	Average error vs. horizontal offset of targeted point . . . . .	28

# List of Algorithms

1	Iterative Laser Targeting Algorithm . . . . .	18
2	Detect Red Laser Dot in Image . . . . .	20

# Acknowledgments

Thanks to: my advisors Robert Pless and Bill Smart; Michael Dixon for the red dot detection code; and my wife Rachel for help with the mundane task of data collection. This research was supported by the Technical Support Working Group (TSWG).

Roman T. Stanchak

*Washington University in Saint Louis*  
*August 2005*

# Chapter 1

## Introduction

The goal of this work is the automatic positioning of a laser pointer in the environment using a stereo camera (Figure 1.1). LASERs (Light Amplification by Stimulated Emission of Radiation) have seen a variety of uses in autonomous systems. They have been used for measurement purposes – vibration, distance, velocity. They also have been used as a guide for human action, such as in surgery or industrial applications. Enabling fine control of the laser pointer in such applications is an important goal.

A necessary aspect of our solution is the stereo camera. In a one camera system, there is an inherent ambiguity between depth and scale that requires assumptions to be made about the environment. With a two camera system, this ambiguity is resolved, and the assumptions relaxed.

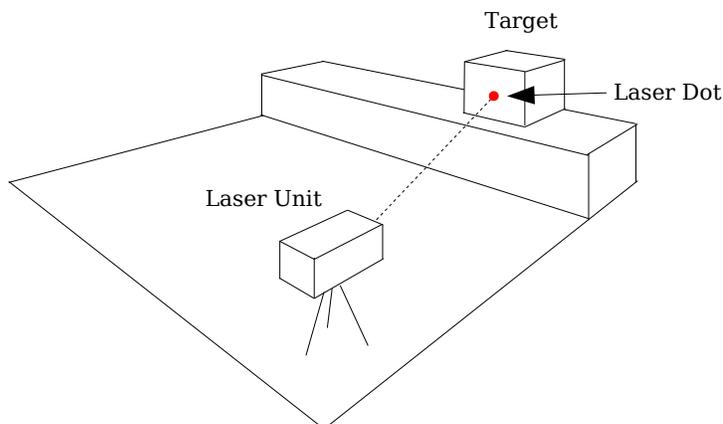


Figure 1.1: Laser Targeting System

We present two algorithms to solve for the transformation between image coordinates and laser controls in order to automatically position the laser pointer in the environment. Although these techniques are adapted from the camera calibration literature, their application in this setting is novel.

The rest of this document is structured as follows: Chapter 2 discusses related work; Chapter 3 presents background information relevant to the understanding of our calibration approach; Chapter 4 introduces our calibration algorithms. Implementation of our system is discussed in Chapter 5. Chapter 6 outlines the experiments performed and the results. Finally, Chapter 7 summarizes our findings and mentions future directions for our work.

## Chapter 2

### Related Work

The task of directing a laser point to a specific point in the environment is related to *visual servoing* [17]. Visual servoing describes a closed loop control method for controlling a robotic manipulator using a camera image. We are not aware of any work to control a laser directly. However, lasers have been used as a visual aide in visual servoing [19] [2].

Closely related to visual servoing is the field of active vision, which involves dynamically adjustable camera systems. The literature describes a variety of systems including networks of pan-able cameras for surveillance[9] [10] [22], and foveated sensors for multiresolution imaging and perception [11] [28] [20].

A variety of literature also exists on the use of lasers projected onto a camera image, primarily in the context of 3D modeling. Laser stripers determine structure by sweeping a laser line across a scene and observing its relative position as it moves [4][7][8][25]. Laser tracking methods project one or many laser points onto an image and track their displacement as they are moved to compute structure [21]. These methods all rely on the geometric relationship between the laser beam, its projection on the camera image, and the camera location to recover the structure of the scene. Other laser systems do not require a camera and instead rely on time of flight or phase differencing [12] [3]. These systems have been used extensively for robot navigation and mapping [16] [24].

Our work borrows significantly from camera calibration literature. Camera calibration techniques attempt to solve for intrinsic and extrinsic parameters of a camera in

a variety of ways. Most relevant is work utilizing geometric relationships to derive linear constraints and solving for the fundamental matrix [23][13][15][33][14].

As mentioned previously, our task is most related to visual servoing, but distinct in how camera observations are utilized to control the laser. In visual servoing, the error between the actual and desired position as observed in the camera image is used as a feedback mechanism to a closed control loop. Visual servoing can be applied to a more general set of control problems where the exact transformation mapping observation to action is not known or easily solveable. Our task is a more simple case that allows us to solve for this transformation exactly and bypass the visual servo feedback loop.

# Chapter 3

## Background

This section introduces notation and concepts relevant to the description of our algorithm for controlling a laser pointer using a stereo camera.

### 3.1 Laser Model

We model our laser control architecture as a black box with two input parameters  $(u, v)$  that determine the direction  $\hat{X}_L$  of the laser beam.  $(u, v)$  are denoted *laser units* ( $lu$ ) and are assumed to lie in the interval  $[-1, 1]$ , which defines their extreme positions. The translation vector defining the source of the laser beam is constant, and fixed at the origin of the laser unit coordinate system (i.e. it equals  $[0, 0, 0]$ ). From the vantage point of the laser unit, the z-axis is straight ahead, the y-axis is up, and the x-axis is to the left (Figure 3.1).

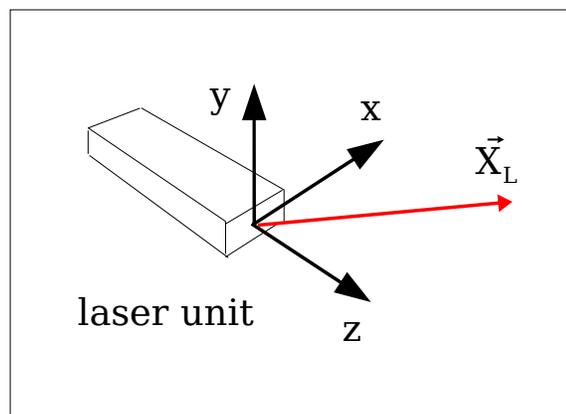


Figure 3.1: Laser coordinate system

We model laser units as having a linear relationship to the direction of the laser beam:

$$\begin{aligned}\mathbf{x}_L &= \begin{bmatrix} u & v & 1 \end{bmatrix}^T \\ w\mathbf{x}_L &= \mathbf{A}_L \vec{X}_L\end{aligned}$$

Where  $w$  is an arbitrary scale,  $\mathbf{A}_L$  is a  $3 \times 3$  matrix, and  $\vec{X}_L$  is a 3D point in the laser coordinate system. The direction of the laser beam is given by the normalized vector  $\hat{X}_L$ . Note that  $\mathbf{x}_L$  is simply the homogeneous coordinate of  $\vec{x}_L$ .

## 3.2 Camera

Each camera is assigned a coordinate system with the origin at the center of projection  $\mathbf{C}$  and oriented similar to the laser - z-axis along the viewing axis of the camera, y-axis up and x-axis left. We model our cameras as the standard pinhole perspective

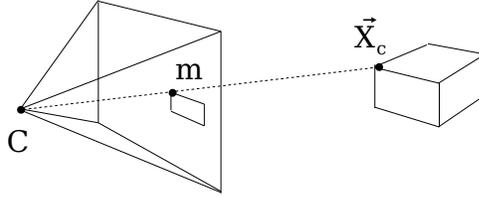


Figure 3.2: Pinhole perspective projection camera model

projection (Figure 3.2). A 3d point in the camera coordinate system is notated  $\vec{X}_C$ , and that point projected on the camera image is denoted  $\vec{m}$ , where:

$$\begin{aligned}s\mathbf{m} &= \mathbf{A}_C \vec{X}_C \\ \mathbf{m} &= \begin{bmatrix} x & y & 1 \end{bmatrix}^T\end{aligned}$$

Where  $\mathbf{A}_C$  is a  $3 \times 3$  matrix that encodes the camera intrinsic parameters (zoom, skew, focal length, image center),  $\mathbf{m}$  is the  $3 \times 1$  homogeneous vector of  $\vec{m}$ , and  $s$  is the projective depth. The laser and camera coordinate system are related by a rotation and translation, such that:

$$\mathbf{R}\vec{X}_C + \mathbf{T} = \vec{X}_L$$

Where  $\mathbf{R}$  is a  $3 \times 3$  rotation matrix and  $\mathbf{T}$  is a  $3 \times 1$  translation vector.

By substituting for  $\vec{X}_L$ , we get the following relation between  $\vec{X}_C$  and  $\mathbf{x}_L$ :

$$\mathbf{A}_L(\mathbf{R}\vec{X}_C + \mathbf{T}) = w\mathbf{x}_L$$

By converting  $\vec{X}_C$  to homogeneous coordinates:

$$\mathbf{X}_C = \begin{bmatrix} X & Y & Z & 1 \end{bmatrix}^T$$

This can be written as the matrix product:

$$\mathbf{H}\mathbf{X}_C = w\mathbf{x}_L$$

Where  $\mathbf{H}$  is the  $3 \times 4$  augmented matrix:

$$\mathbf{H} = \mathbf{A}_L[\mathbf{R}|\mathbf{T}]$$

$\mathbf{H}$  is the direct transformation between 3D camera coordinates and laser units. Section 4.1 presents an algorithm to solve for  $\mathbf{H}$ .

### 3.3 Multiple Cameras

The case of two cameras (Figure 3.3) is identical to that of a camera and laser. Each camera (centered at  $\mathbf{C}$  and  $\mathbf{C}'$ ) has its own 3D coordinate system, with points  $X_C$  in one camera's coordinate system related to points  $X'_C$  in the other by a rotation and translation:

$$\mathbf{R}\vec{X}_C + \mathbf{T} = \vec{X}'_C$$

Plugging this into the projection equation for the homogeneous image point  $\mathbf{m}'$ :

$$\mathbf{A}'(\mathbf{R}\vec{X}_C + \mathbf{T}) = s'\mathbf{m}'$$

Note that the projection equation can be rewritten as:

$$\vec{X}_C = s\mathbf{A}^{-1}\mathbf{m}$$

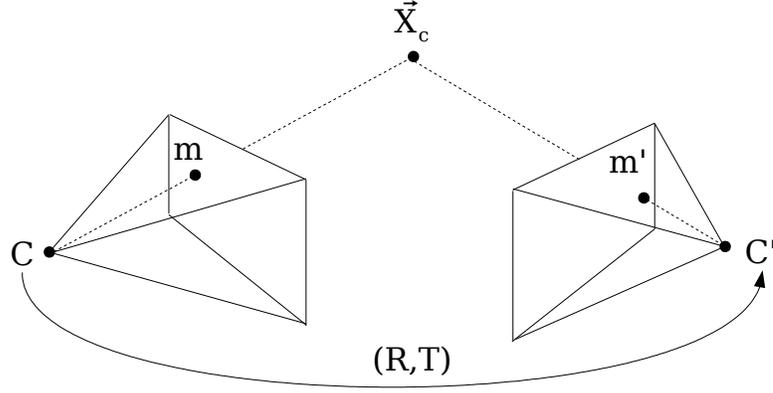


Figure 3.3: Multiple cameras

Plugging this in above:

$$\mathbf{A}'(\mathbf{R}(s\mathbf{A}^{-1}\mathbf{m}) + \mathbf{T}) = s'\mathbf{m}'$$

Rewriting:

$$\mathbf{m}' = \frac{s}{s'}\mathbf{A}'\mathbf{R}\mathbf{A}^{-1}\mathbf{m} + \mathbf{A}'\mathbf{T}$$

This equation implies that  $\mathbf{m}'$  lies on the line  $l'$  going through the points  $\mathbf{e}' = \mathbf{A}'\mathbf{T}$  and  $\mathbf{A}'\mathbf{R}(\mathbf{A}^{-1}\mathbf{m})$  (see Figure 3.4)[27][14]. Three homogeneous points  $(\mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_3)$  are determined to be collinear in projective space[6] if:

$$\mathbf{x}_1^T(\mathbf{x}_2 \times \mathbf{x}_3) = 0$$

In other words,

$$\mathbf{m}'^T(\mathbf{A}'\mathbf{T} \times \mathbf{A}'\mathbf{R}(\mathbf{A}^{-1}\mathbf{m})) = 0$$

Using the property that for a non-singular square matrix  $K$ ,  $Ka \times Kb = K^*(a \times b)$ , where  $K^*$  is the cofactor matrix of  $K$  and  $K^* = \det(K)(K^T)^{-1}$ , this becomes:

$$\mathbf{m}'^T(\det(\mathbf{A}')(\mathbf{A}'^{-T})(\mathbf{T} \times \mathbf{R}(\mathbf{A}^{-1}\mathbf{m}))) = 0$$

Where  $\mathbf{A}'^{-T}$  is shorthand for  $(\mathbf{A}'^T)^{-1}$ .

Then dividing out the  $\det(\mathbf{A}')$ :

$$\mathbf{m}'^T(\mathbf{A}'^{-T}(\mathbf{T} \times \mathbf{R}(\mathbf{A}^{-1}\mathbf{m}))) = 0$$

By defining the matrix  $[\mathbf{T}]$  as follows (which is equivalent to the cross product):

$$[\mathbf{T}] = \begin{pmatrix} 0 & -t.z & t.y \\ t.z & 0 & -t.x \\ -t.y & t.x & 0 \end{pmatrix}$$

This equation becomes:

$$\mathbf{m}'^T \mathbf{A}'^{-T} [\mathbf{T}] \mathbf{R} \mathbf{A}^{-1} \mathbf{m} = 0$$

This matrix  $\mathbf{A}'^{-T} [\mathbf{T}] \mathbf{R} \mathbf{A}^{-1}$  is called the Fundamental Matrix, denoted  $\mathbf{F}$ , and the following equation is called the Epipolar constraint[13][33]:

$$\mathbf{m}'^T \mathbf{F} \mathbf{m} = 0$$

We will use the Epipolar constraints between the laser beam and each camera to solve for the transformation from image points to laser controls in Section 4.2.

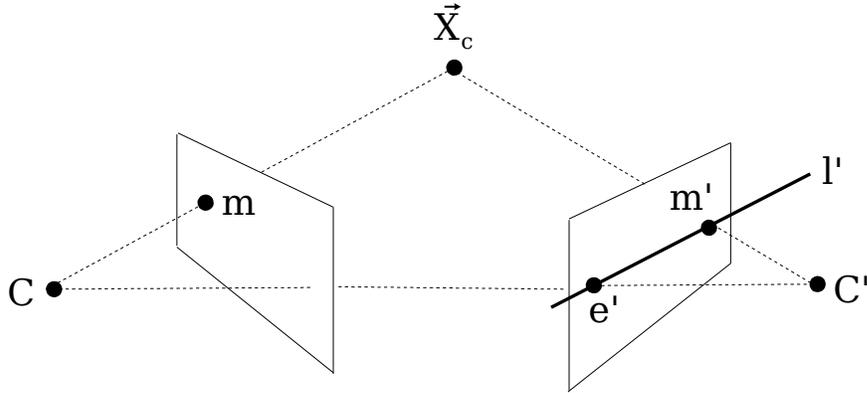


Figure 3.4: Epipolar constraint

# Chapter 4

## Laser Calibration

The goal of the system is to solve for the laser controls necessary to send the laser dot to point in space, specified in camera coordinates. We present two methods of accomplishing this. The first solves for the transformation  $\mathbf{H}$  directly, but requires, as input, many examples of the 3D camera coordinates of the laser dot on the camera image. The second is an indirect method that utilizes the epipolar constraint and requires examples of corresponding 2D image coordinate of the laser dot in 2 or more uncalibrated cameras.

### 4.1 Direct Calibration

As outlined in Section 3.2, the transformation between 3D camera coordinates and laser units is modeled as a  $3 \times 4$  homography matrix,  $\mathbf{H}$ .

$$\mathbf{H} = \begin{bmatrix} h_1 & h_2 & h_3 & h_4 \\ h_5 & h_6 & h_7 & h_8 \\ h_9 & h_{10} & h_{11} & h_{12} \end{bmatrix}$$

Where

$$\mathbf{HX}_C = w\mathbf{x}_L$$

In order to solve for  $\mathbf{H}$ , examples of corresponding  $\vec{x}_L$  and  $\vec{X}_C$  are collected. Each correspondence provides three constraints on  $\mathbf{H}$  of the form:

$$\begin{aligned} Xh_1 + Yh_2 + Zh_3 + h_4 &= wx_L \\ Xh_5 + Yh_6 + Zh_7 + h_8 &= wy_L \\ Xh_9 + Yh_{10} + Zh_{11} + h_{12} &= w \end{aligned}$$

However,  $w$  is a free variable, so this is truly only two constraints on the parameters of  $\mathbf{H}$ . By substituting the third equation for  $w$  in the first two equations,  $w$  is eliminated and the following two linear constraints remain:

$$\begin{aligned} Xh_1 + Yh_2 + Zh_3 + h_4 &= x_L(Xh_9 + Yh_{10} + Zh_{11} + h_{12}) \\ Xh_5 + Yh_6 + Zh_7 + h_8 &= y_L(Xh_9 + Yh_{10} + Zh_{11} + h_{12}) \end{aligned}$$

By reorganizing terms and stacking constraints, we get a linear system of equation in terms of  $\mathbf{H}$ .

$$\mathbf{U}\vec{h} = \mathbf{0}$$

Where

$$\mathbf{U} = \begin{bmatrix} X & Y & Z & 1 & 0 & 0 & 0 & 0 & -x_L X & -x_L Y & -x_L Z & -x_L \\ 0 & 0 & 0 & 0 & X & Y & Z & 1 & -y_L X & -y_L Y & -y_L Z & -y_L \\ \dots & & & & & & & & & & & \end{bmatrix}$$

$$\vec{h} = [h_1 \ h_2 \ h_3 \ h_4 \ h_5 \ h_6 \ h_7 \ h_8 \ h_9 \ h_{10} \ h_{11} \ h_{12}]^T$$

With 12 free variables, this system can be solved with 6 or more correspondences between laser units and 3D camera coordinates. The trivial solution,  $\vec{h} = \mathbf{0}$  is not useful, so we perform a singular value decomposition and choose  $\vec{h}$  to be the eigenvector with the smallest non-zero eigenvalue [34]. The results for this method are presented in Chapter 6.

## 4.2 Epipolar Calibration Method

An alternate solution is to apply the epipolar constraint between the laser and camera. The epipolar constraint is typically defined between corresponding points on two images. However, by thinking of the laser beam as an inverted beam of light, it has exactly the same behavior as a beam of light in the pinhole perspective camera model. Therefore, by thinking of the laser as an inverse camera, and laser coordinates as pixel coordinates on a virtual image, the epipolar constraint applies in the same way. As a result, there exists a  $3 \times 3$  fundamental matrix  $\mathbf{F}$  for which:

$$\mathbf{x}_L^T \mathbf{F} \mathbf{m} = 0$$

Where  $\mathbf{x}_L$  is a homogeneous laser coordinate  $([u, v, 1])$ , and  $\mathbf{m}$  is the homogeneous image coordinate  $([x, y, 1])$  where the laser dot is projected. Each camera  $i$  adds an additional constraint on  $\mathbf{x}_L$ :

$$\mathbf{x}_L^T \mathbf{F}_i \mathbf{m}_i = 0$$

Computer Vision literature has examined the epipolar constraint extensively and describes a variety of ways to solve for the fundamental matrix [33]. We utilize the normalized 8 point algorithm [23] [15], as an implementation was readily available[18]. This method requires 8 or more corresponding  $\mathbf{x}_L$  and  $\mathbf{m}$  for each camera.

Having calculated  $\mathbf{F}$ , our end goal is: given image coordinates of the point we would like to target the laser on, we would like to calculate the necessary coordinates to send the laser. Each fundamental matrix provides one constraint on  $\mathbf{x}_L$ :

$$x_L(f_1 m_x + f_2 m_y + f_3) + y_L(f_4 m_x + f_5 m_y + f_6) + f_7 m_x + f_8 m_y + f_9 = 0$$

Which can be rewritten as:

$$x_L(f_1 m_x + f_2 m_y + f_3) + y_L(f_4 m_x + f_5 m_y + f_6) = -(f_7 m_x + f_8 m_y + f_9)$$

Table 4.1: Comparison of requirements for laser calibration methods

Method	Direct	Epipolar
Number of correspondences required for calibration	6	$8 \times 2$
Requires 3D coordinates?	Yes	No
Correspondence required at run time?	No	Yes
Requires calibrated cameras?	No	Yes

However, each camera provides an additional constraint, and these can be stacked to create a linear system.

$$\begin{bmatrix} f_1 m_x + f_2 m_y + f_3 & f_4 m_x + f_5 m_y + f_6 \\ f'_1 m'_x + f'_2 m'_y + f'_3 & f'_4 m'_x + f'_5 m'_y + f'_6 \\ \dots & \dots \end{bmatrix} \begin{bmatrix} x_L \\ y_L \end{bmatrix} = \begin{bmatrix} -(f_7 m_x + f_8 m_y + f_9) \\ -(f'_7 m'_x + f'_8 m'_y + f'_9) \\ \dots \end{bmatrix}$$

With two or more cameras this system can be solved using linear least squares. The main downside to this method is that the image coordinates of the targeted point must be known in at least two cameras at every time step. In other words, in a two camera system, the stereo correspondence between the left and right images must be known for the point to be targeted by the laser.

Both the direct and epipolar calibration methods model the same geometry and should be equally expressive. The main difference between them is in their input requirements. An overview of the requirements for each method is given in Table 4.1. The primary advantage of the direct approach is that 3D sensor data can be utilized explicitly. The epipolar approach is advantageous in that a calibrated camera is not required. Additionally, the greater foundation of previous work dedicated to solving for the fundamental matrix likely makes this a reliable method. Experimental results comparing these two methods are shown in Figure 6.4 and discussed in Chapter 6.

# Chapter 5

## Implementation

This section describes details relevant to our implementation of the laser targeting system.

### 5.1 Laser Unit

The laser unit used in our system is a modified Polytec OFV 056 Scanning Vibrometer (Figure 5.1) [26]. The direction of the laser beam is determined by the angle of 2 sequential mirrors in the laser's path. Each of these mirrors is mounted on a servo motor [29] which is controlled by a PCI digital/analog conversion card (Figure 5.2).



Figure 5.1: Polytec Laser with Tyzx Stereo Camera

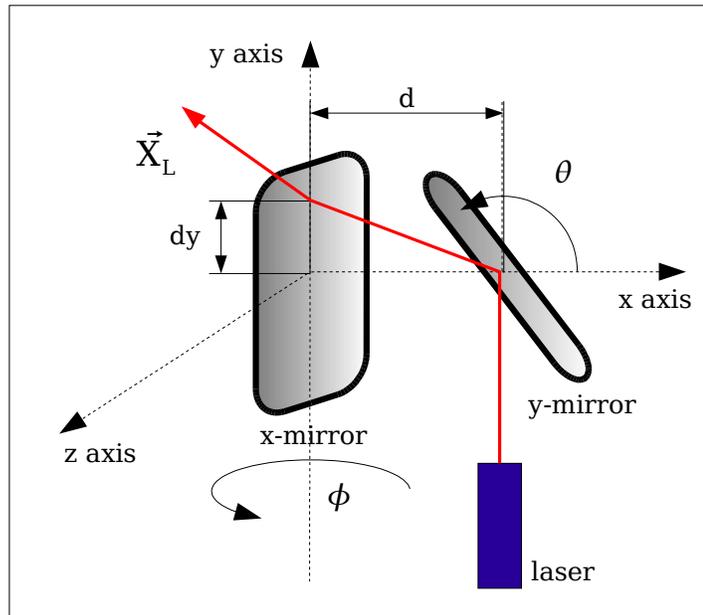


Figure 5.2: Diagram of laser mirror layout.

The exact relationship between laser units and control for this unit was unknown, so we plot a regularly spaced grid of laser coordinates on the image plane. As can be seen in Figure 5.3, there is significant distortion at the extremes of the image that indicates our unit may break some of our assumptions. Examining the actual laser unit, there are two probable causes: first, the mirrors are separated by some distance  $d$ , such that the angle  $\theta$  of the first mirror causes the laser beam to hit the second mirror at a variable position  $dy$  along its axis (Figure 5.2). This violates our fixed origin assumption. Second, it is probable that the  $(u, v)$  laser inputs map to mirror angles, and consequently laser angles, thereby violating our linearity assumption.

In order to analyze the effect of these violations, some measurements were performed. The maximum laser angles were measured as  $\pm 20^\circ$  in the x and y directions, and the distance  $d$  between mirror axes was measured as 40 mm. The distance between the mirrors causes a translation in the origin of the laser beam as the angle varies. This translation varies in the y-direction only by  $d \cdot \tan(\theta)$ . At the maximum angle of  $20^\circ$ , the error is  $40 \cdot \tan(20^\circ) = 14.5$  mm.

To determine the effect of our linear model versus an angular model, consider the laser targeting a point  $X = [x, y, z]$ . To simplify, for the linear model, we assume the

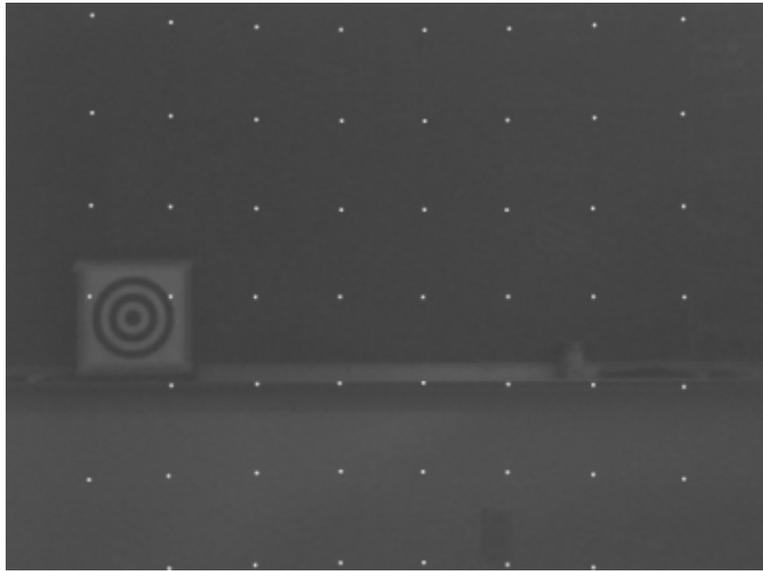


Figure 5.3: Distortion in laser control.

directional vector is  $[u, v, 1]$  and for the angular model, the laser inputs control laser angles directly ( $\theta = u, \phi = v$ ). Therefore, the targeted points are:

$$\begin{aligned} X_{linear} &= \begin{bmatrix} zu & zv & z \end{bmatrix} \\ X_{angular} &= \begin{bmatrix} z\tan(u) & z\tan(v) & z \end{bmatrix} \end{aligned}$$

The error between them is then:

$$\sqrt{(zu - z\tan(u))^2 + (zv - z\tan(v))^2}$$

The maximum error occurs when both mirrors are at their  $20^\circ$  position, and is therefore  $z\sqrt{2}\tan(20^\circ)$ . At 1 m, the error is 7 mm, but increases linearly with depth.

These are damaging discoveries, but to take them into account adds non-linearity into our system, and therefore prevents us from attaining a solution directly. These defects are only evident at extreme angles, so it is assumed that the object targeted by the laser will be roughly centered in the image, thereby minimizing the effect of these assumptions.

## 5.2 Stereo Camera

We utilize a dedicated hardware stereo solution from Tyzx, Inc. as our sensor platform [31],[32]. It provides a rigid housing for the pre-calibrated stereo cameras (Figure 5.4), which feed directly into a custom PCI card dedicated to computing stereo correspondence. This PCI card is capable of computing up to 2.6 billion pixel-disparity evaluations per second, and provides 512x480 images at 30 fps [31]. Taking advantage of this solution allows us to bypass the hurdles inherent to stereo vision, and get 3D information directly from the hardware driver API. One disadvantage to this solution is that the depth data is sparse: depth data can only be calculated in areas of the image with sufficient texture. We addressed this problem by ignoring calibration points without depth.



Figure 5.4: Tyzx stereo camera housing.

## 5.3 Calibration

Our calibration method requires corresponding points in image and laser coordinates. We utilized two approaches for determining these correspondences: a forward method, where a laser coordinate is given, and the red dot is found on the image, and an inverse method, where an image coordinate is given, and the red dot is moved there through iterative optimization routine.

### 5.3.1 Forward Method

The forward method moves the laser in a grid and for each laser position ( $\vec{x}_L$ ), it records the image coordinate ( $\vec{x}$ ) where the laser dot is found. The position of the laser dot is specified manually by the operator through a mouse click, or automatically by a red dot detection algorithm described below.

### 5.3.2 Inverse Method

The inverse calibration method specifies an image coordinate, and attempts to move the laser dot there using an iterative minimization algorithm. The minimization is essentially a feedback loop between the laser and the red dot detection in the camera image. It is given in Algorithm 1. As an implementation detail, let it be noted that when the laser dot was on or near a chessboard corner, it would cause the corner detection algorithm to mis-detect the corner. As a result, the iterative convergence algorithm would never detect the red laser dot and the corner in the same position and an infinite loop would result. To avoid this problem, the laser was moved off the camera image when the corner detection image was captured.

---

#### Algorithm 1 Iterative Laser Targeting Algorithm

---

- 1: the target point is at image coordinate  $(a, b)$
- 2: **repeat**
- 3:   the laser position is  $(u, v)$
- 4:   the laser dot is detected in the image at  $(x, y)$ .
- 5:   the laser is moved by the amount:
- 6:    $(u, v) = [u + \alpha(a - x), v + \beta(b - y)]$
- 7: **until**  $a == x$  **AND**  $b == y$

Where  $\alpha, \beta$  are scale factors that are determined empirically to minimize  $(a - x, b - y)$ .

---

### 5.3.3 Red Dot Detection

The red dot detection algorithm attempts to automate the finding of the laser dot projected on the camera image. A number of issues had to be considered to make this algorithm robust. The first is that the center of the laser tended to saturate all the camera color channels. Thus, it was not actually red, but white. Secondly, the projection of the laser in the camera image caused some undesirable artifacts (shown in Figure 5.5) that may confuse a dot detection algorithm. With these considerations

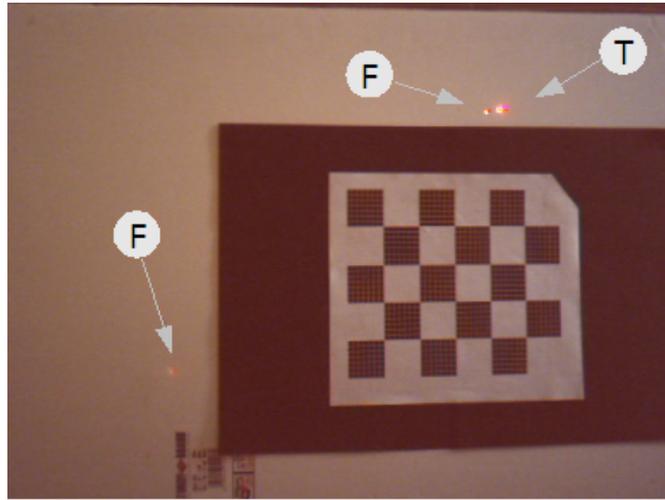


Figure 5.5: Laser artifacts (F) and true laser position (T) in camera image.

in mind we outline the algorithm in Algorithm 2. The basic idea of the algorithm is to (1) determine which pixels in the image are attributable to the laser beam using background subtraction (2) zero out pixels that are not (3) compute the weighted center of mass of those pixels based on the brightness of the red color channel (4) refine this position estimate over a smaller region.

In practice, the algorithm finds the red dot consistently in an office environment with a moderate level of ambient lighting. If the overhead fluorescent lights are turned on, the laser dot becomes virtually invisible in the camera image.

### 5.3.4 Point Selection

One major aspect of calibration quality is the set of calibration points. The Forward Method allows control of which laser coordinates we wish to calibrate to, and in section 6.2 we evaluate various ranges and densities to determine the optimal set. The Inverse Method allows us to specify the image coordinate we wish to move the laser to. This is much more advantageous because the calibration can be an interactive procedure. The operator can tune the calibration to be accurate in a particular place in the environment by adding calibration points at that point in the image.

Our strategy for point selection was to automate as much as possible. We used a chessboard calibration pattern, which allows us to utilize a corner detection algorithm

**Algorithm 2** Detect Red Laser Dot in Image

---

```

1: Move laser out of image
2: Capture a background image of the scene, BG
3: BG = RedChannel(BG)
4: Move laser to desired location
5: Capture image I
6: I = RedChannel(I)
7: I' = I - BG
8: I' = dilate(I')
9: for all  $(x, y) \in \mathbf{I} \mid \mathbf{I}(x, y) < \textit{threshold}$  do
10:   I( $x, y$ ) = 0
11: end for
12:  $(u, v)$  = CenterOfMass(I)
13: for all  $(x, y) \in I \mid (x, y) \notin (u \pm \textit{width}, v \pm \textit{width})$  do
14:   I( $x, y$ ) = 0
15: end for
16:  $(u, v)$  = CenterOfMass(I)
17: return  $(u, v)$ 

```

---

to automatically specify the set of target points. We utilize Vezhnevets' excellent corner detection enhancement[30] to Bouguet's Matlab Camera Calibration Toolbox [5]. The result of corner detection is shown in figure 5.6. Let us emphasize that our calibration method does not require a calibration pattern, it is simply a convenient way of automatically, accurately, and consistently specifying a set of image points in multiple camera images. Experimental results examining the effect of the point set are given in Figure 6.3 and discussed in Section 6.3.3.

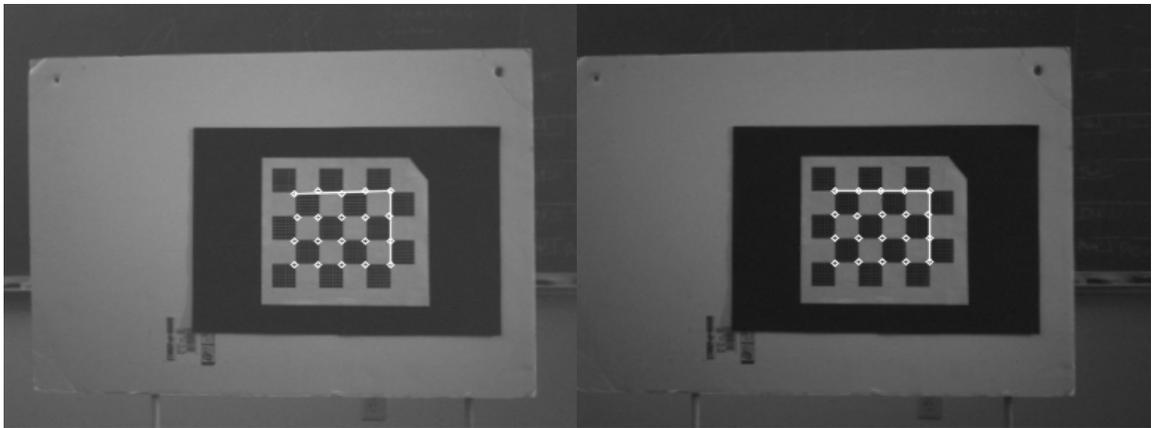


Figure 5.6: Corner detection result for stereo image pair.

# Chapter 6

## Experimental Results

We perform a series of experiments to compare the two calibration methods and determine how the the various parameters affect the quality of the calibration.

### 6.1 Calibration Procedure

Calibration was performed in an office environment using a textured planar surface that could be easily positioned at different depths. The calibration procedure moves the laser in a square grid of regular spacing, and for the manual calibration methods, the user is asked to click the point on the camera image where the laser dot is projected. This procedure is repeated with the textured plane at various distances in front of the camera.

### 6.2 Measurement Procedure

For each calibration configuration, accuracy was measured by sampling 4 points in 3 positions: *close* (80cm), *medium* (120cm), and *far* (180cm). In addition, the *far* position was offset to the left side of the camera. The target points were automatically detected using a corner detection algorithm and a chessboard calibration pattern. In each position, the four points were the extreme corners of the chessboard as detected by the corner detection algorithm. The chessboard pattern was overlaid with a finer grid of  $5 \times 5$  mm squares for hands-free measurement. We measured accuracy as the signed  $(x, y)$  distance in mm on the chessboard plane from the target to the actual position of the laser dot. We report the average euclidean distance for each point set

individually, then the overall mean. The calibration algorithm is a minimization over pixel differences, so points further from the camera will naturally have a higher error. We report error as a function of depth to illustrate this relationship in Figure 6.6.

## 6.3 Results

This section describes the experiments performed and their results.

### 6.3.1 Error w.r.t the number of calibration planes.

In this experiment, we performed the calibration with 2, 3 and 4 depth planes and report the accuracy of each. The depth planes were roughly fronto-parallel to the camera and spaced uniformly between 700mm and 1800mm away. The size of the calibration grid was  $.6 lu$ . The number of points per plane was varied to maintain the same total number of points used in the calibration. Two planes were tested with 32 points total, three with 48, and four with 36. The average error for each number of planes is reported in Figure 6.1. The secondary plots show the error broken down by the depth of the points. The results show little effect as the number of planes

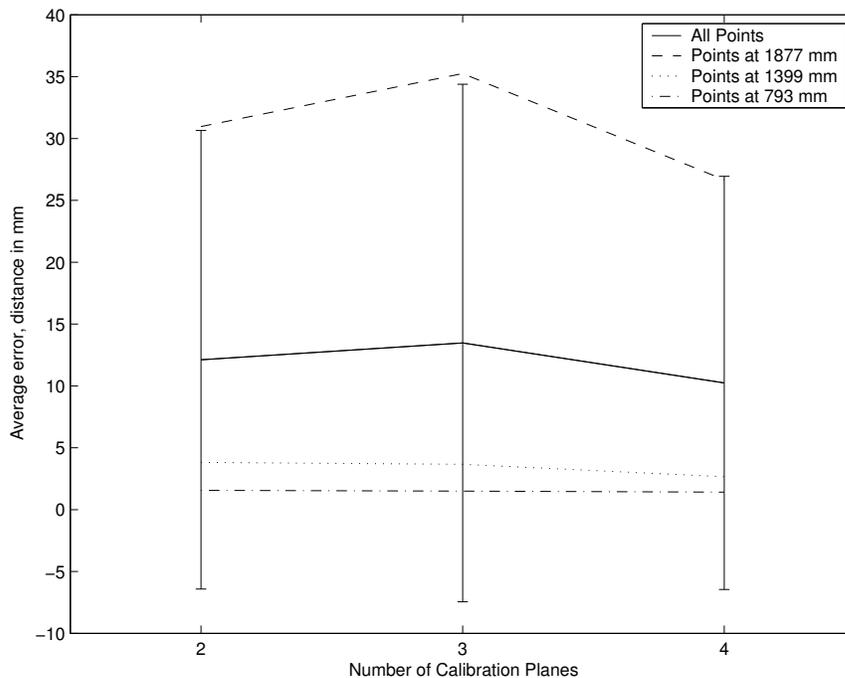


Figure 6.1: Average error vs. the number of calibration planes.

increases beyond two. The slight differences in the graph are within the margin of error of the data. Calibration with only one depth plane is a degenerate configuration [1], and leads to a rank 2  $\mathbf{H}$  matrix in the absence of noise. When used, it gives errors in excess of a meter at depths other than the calibration depth.

From these results, we conclude that the extra labor for more depth planes is not worthwhile, and perform all subsequent calibrations with 2 depth planes.

### 6.3.2 Error w.r.t. calibration grid size.

In this experiment, we varied the size of the calibration grid. The calibration was performed with 2 depth planes and a  $5 \times 5$  grid of laser points ranging from  $(-x_L, -y_L)$  to  $(x_L, y_L)$ , where  $(x_L, y_L)$  was varied from  $.2 lu$  to  $1.0 lu$  by  $.2 lu$  increments. The average error for each size is reported in Figure 6.2. Results over all tested points show a slight downward trend as the size of the grid increases. This trend is well within the error of the data. However, analyzing the error for the *far* point set, there is a distinct improvement as the grid size increases. Also notable is that the error for the *close* points is significantly lower for the smaller grid. One possible explanation is

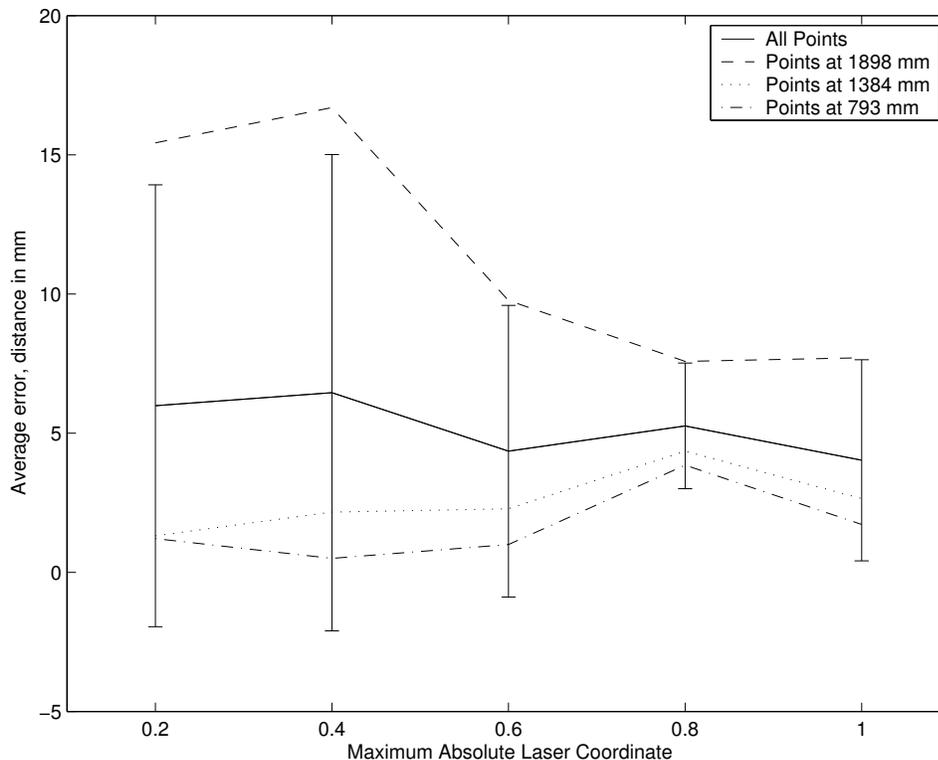


Figure 6.2: Average error vs. size of the calibration grid.

that as the grid size increased, sometimes the laser dot did not show up in the camera image, therefore leaving the larger sizes with fewer calibration points. Indeed, the total number of calibration points for the data points are (50, 50, 50, 39, 24). Error for *close* points is roughly constant up until  $.8 lu$ , when we see the total number of data points drop to 39. Peeking ahead to Figure which plots error as a function of point density, we are disappointed to see that overall error stay fairly constant beyond the  $5 \times 5$  grid. We choose to minimize the error of the *far* point set at the expense of close points, and use the  $1.0 lu$  grid in subsequent calibrations.

### 6.3.3 Error w.r.t. calibration point density.

In this experiment, we varied the number of calibration points per plane. The calibration was performed with 2 depth planes, and a grid size of  $(1, 1) lu$  at point densities of 16, 25, 36, 64, 100 and 144 points per plane. The average error for each density is reported in Figure 6.3. The results show a decrease in error up to 25 points, then a leveling off. Examining the different point sets, we see a continued decrease in error

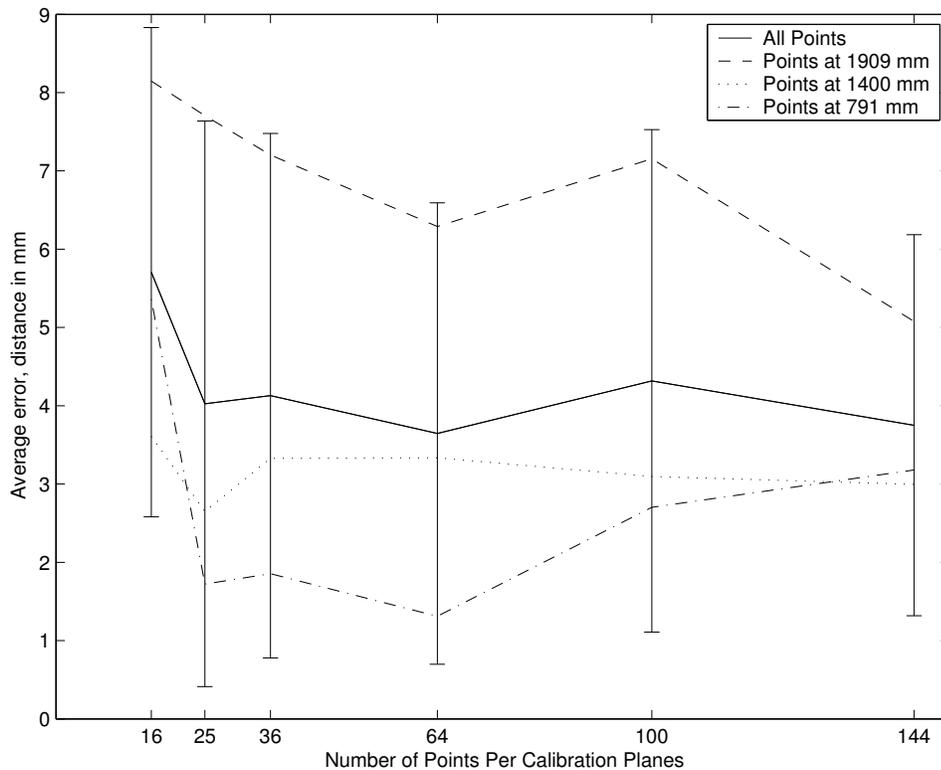


Figure 6.3: Average error vs. the number of points per calibration plane

for *far* points, but an increase for *close* points. One possible explanation for this behavior is that with a greater number of points comes greater overall distortion. Since the worst distortion occurs at the *far* plane, this may cause the least squares algorithm to compensate for these errors at the cost of the closer points. Also of note is that the results for 49 points per plane were omitted because their error was disproportionately high. Repeat tests confirmed this puzzling result. We can only conclude that this was a strange degenerate configuration for our particular configuration.

### 6.3.4 Direct vs. Epipolar method.

In this experiment, we compare the direct and epipolar methods. For each method, we perform the calibration with 2 depth planes, a grid size of  $(1, 1)$  *lu* with 100 points per plane. We use the same set of clicked points as the data set for each calibration method. The average error for each method is reported in Figure 6.4. The results show that the epipolar method with around 1 mm less error than the direct method overall, with the improvements evident in each point set. As mentioned previously,

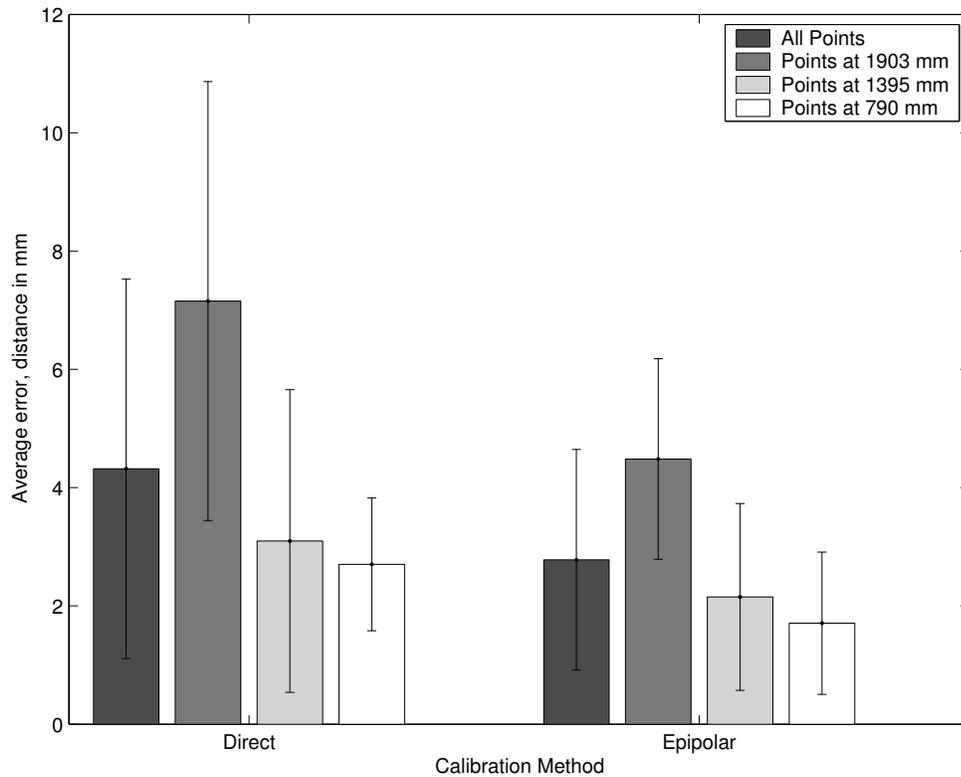


Figure 6.4: Average error for Direct vs. Epipolar calibration methods

each method models the same underlying geometry, so we can only attribute this improvement to the maturity of the fundamental matrix solution algorithm used. Another possible explanation is that the epipolar method optimizes over the image data directly, whereas there is an additional transformation to 3D coordinates using the direct method, which may introduce additional error.

### 6.3.5 Manual vs. Automatic correspondence

In this experiment, we compare manual calibration with automatic calibration for both forward and iterative inverse point selections. The forward and iterative methods rely on very different point sets, so we perform each using automatic point detection and manual point clicking. The forward calibration was performed with 2 depth planes, a grid size of (1, 1) lu and 100 points per plane. The iterative method was tested at 4 positions with 8 points at each position. The average error for each method is reported in Figure 6.5. The results show that the manual iterative calibration gives the best results seen thus far. However, it is also notable that for forward

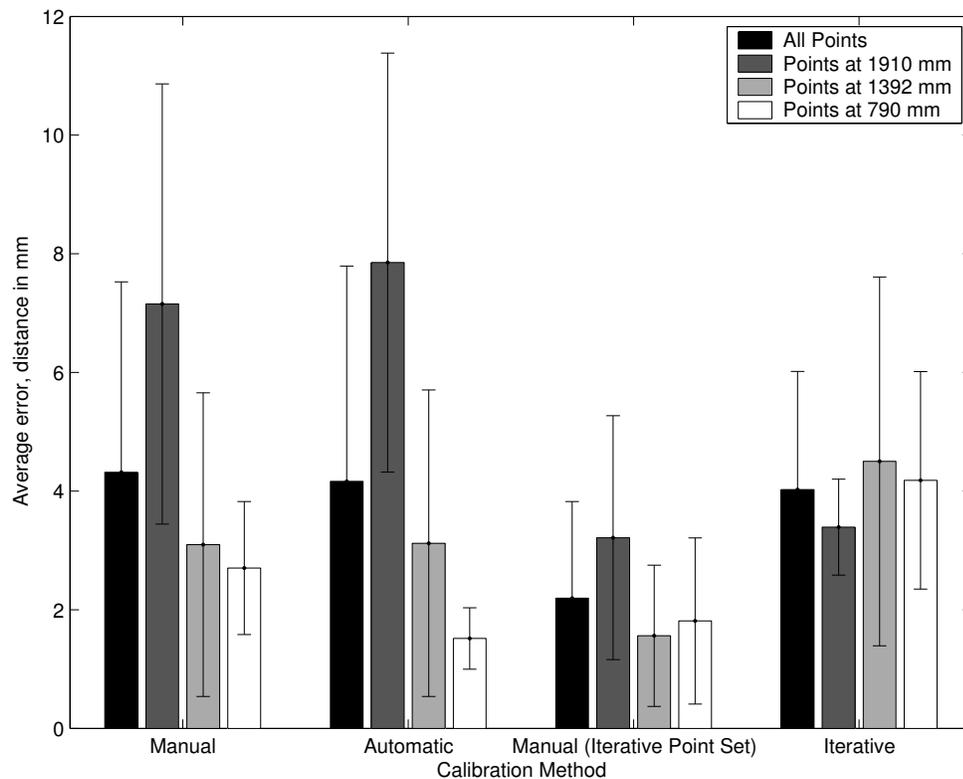


Figure 6.5: Average error for Manual vs. Automatic point correspondence.

point selection, manual and automatic methods are roughly equivalent. Automatic iterative is about the same overall as the two forward methods, although the error profile is quit flat: the distance of the points seems to have little effect on error. The main point to consider with these results is the quality of the selection of calibration points. The forward point set has 200 total points compared to 32 for the inverse set. The advantage here may have been the fine specification of target image coordinates thanks to the sub-pixel accuracy of the corner detector.

### 6.3.6 Error wrt. to depth of targeted point

In this experiment, we examine the relationship of scene to targeting error. We use the best known calibration from our previous tests, the *Manual Iterative* calibration from Figure 6.5. We test 5 sets of 4 points, each at a different depth and horizontal offset, then sort the points by z component and plot the error. The error relative to depth is given in Figure 6.6. Results are very noisy, but as expected, do show a general trend toward a higher error with greater depth.

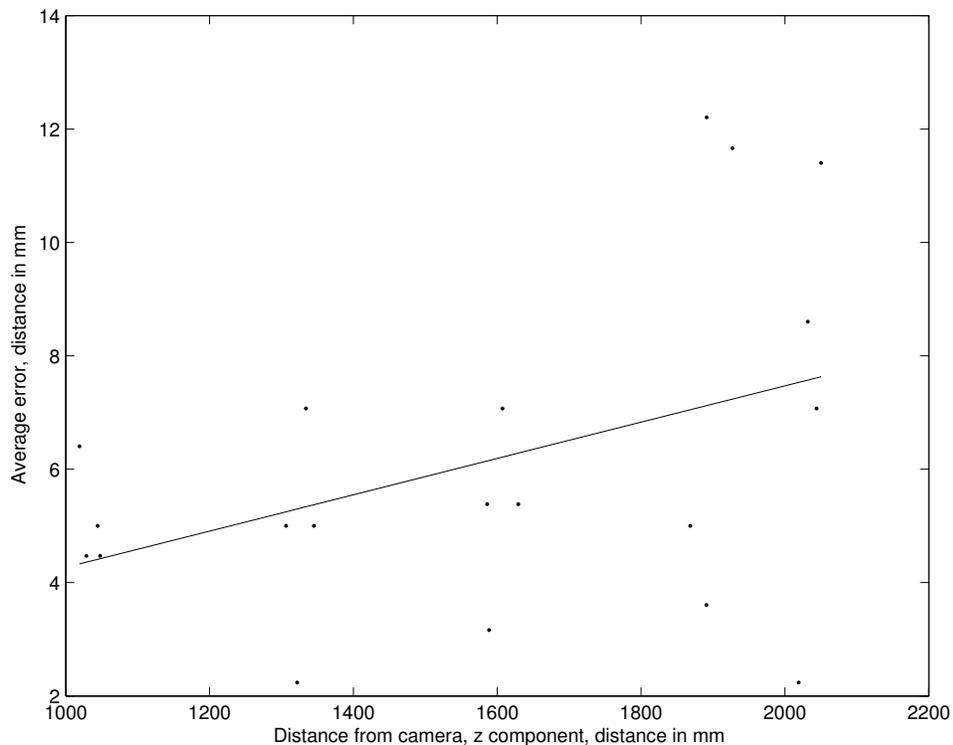


Figure 6.6: Average error vs. depth of targeted point

### 6.3.7 Error wrt. to horizontal offset of targeted point

This experiment examines the relationship between the horizontal offset of the target point (in 3D camera coordinates) to the targeting error. We use the same data set from Section 6.3.6, sort the data by the absolute value of the x component and plot the error. The error relative to the horizontal offset is given in Figure 6.7. Results are again very noisy, but as expected, there is a general trend that the greater the x-component, the greater the error.

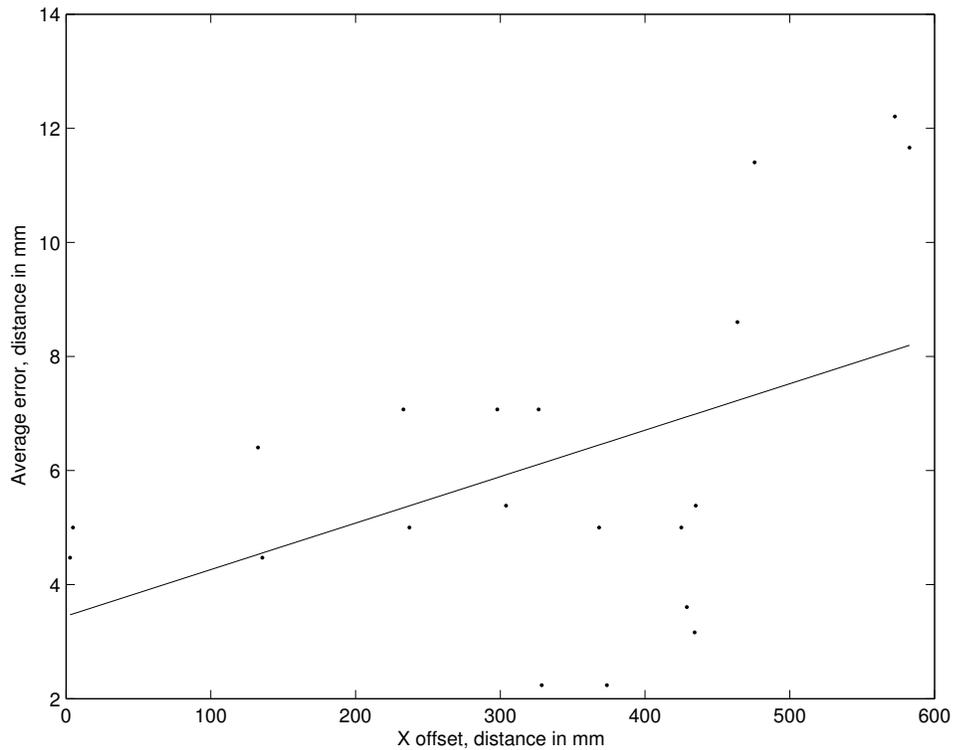


Figure 6.7: Average error vs. horizontal offset of targeted point

# Chapter 7

## Conclusion

We presented two calibration methods to aim a laser pointer in the environment. The direct method uses linear constraints to solve for the transformation between 3D camera coordinates and laser units. It is suitable for use with integrated depth sensors that provide 3d information out of the box. The epipolar method uses the epipolar constraint to solve for the fundamental matrices that describe the relationship between image coordinates and laser voltages. The epipolar method is best suited to an uncalibrated stereo camera setup because only corresponding image points are required. Experimental results verified the accuracy of both methods and determined them equally accurate.

In future work, there are number of directions we would like to pursue. First, to increase the accuracy of the system, we would like to better model our specific laser unit. In order to do this, two nonlinearities will be introduced to the transformation. Laser units dictate the angle of the mirrors, which affect the offset by the sine of the angle. Furthermore, spacing between the mirrors will shift the y component by an additional multiple of the sine of the angle. We believe that explicitly modelling these factors will alleviate the distortion experienced at large angles.

Finally, we present this approach as an alternative to the visual servoing approach. In future work we wish to apply this approach to our system and compare it with our methodologies.

## References

- [1] G. Adiv. Inherent ambiguities in recovering 3D motion and structure from a noisy flow field. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, pages 70–77, 1985.
- [2] Nicolas Andreff, Bernard Espiau, and Radu Horaud. Visual servoing from lines. *International Journal of Robotics Research*, 21(8):679–700, August 2002. Also available at <http://www.inrialpes.fr/movi/publi/Publications/2002/AEH02>.
- [3] F. Blais, J.A. Beraldin, and S.F. El-Hakim. Range error analysis of an integrated time-of-flight, triangulation, and photogrammetry 3d laser scanning system. In *SPIE Proceedings*, volume 4035, Orlando, FL, 2000. Also available at <http://iit-iti.nrc-cnrc.gc.ca/iit-publications-iti/docs/NRC-43649.pdf>.
- [4] R.C. Bolles, J.H. Kremers, and R.A. Cain. A simple sensor to gather three dimensional data. Technical report, Stanford University, 1981.
- [5] J-Y. Bouguet. Camera calibration for matlab. Technical report, Intel Corporation, 2001. Available at [http://www.vision.caltech.edu/bouguetj/calib\\_doc/](http://www.vision.caltech.edu/bouguetj/calib_doc/).
- [6] Herbert Busemann and Paul Joseph Kelly. *Projective Geometry and Projective Metrics*. Academic Press, New York, 1953.
- [7] C.H. Chen and A.C. Kak. Modeling and calibration of a structure light scanner for 3D robot vision. In *Proceedings of IEEE Conference on Robotics and Automation*, pages 807–815, 1987.
- [8] M. Chiarella and K. A. Pietrzak. An accurate calibration technique for 3-d laser stripe sensors. *Optics, Illumination, and Image sensing for Machine Vision IV*, 1194:176–185, 1989.

- [9] R. T. Collins and Y. Tsin. Calibration of an outdoor active camera system. In *In Proceedings of the 1999 Conference on Computer Vision and Pattern Recognition*, pages 528–534. IEEE Computer Society, 1999. Also available at <http://citeseer.csail.mit.edu/collins99calibration.html>.
- [10] J. Davis and X. Chen. Calibrating pan-tilt cameras in wide-area surveillance networks. In *Proc. of IEEE International Conference on Computer Vision*, 2003. Also available at <http://citeseer.csail.mit.edu/davis03calibrating.html>.
- [11] Fadi Dornaika and James H. Elder. Image registration for foveated omnidirectional sensing. In *ECCV '02: Proceedings of the 7th European Conference on Computer Vision-Part IV*, pages 606–620, London, UK, 2002. Springer-Verlag.
- [12] Gregory Dudek and Michael Jenkin. *Computational Principles of Mobile Robotics*, pages 67–68. Cambridge University Press, 2000.
- [13] Olivier D. Faugeras, Quang-Tuan Luong, and Stephen J. Maybank. Camera self-calibration: Theory and experiments. In *European Conference on Computer Vision*, pages 321–334, 1992. Also available at <http://citeseer.ist.psu.edu/faugeras92camera.html>.
- [14] A. Fusiello. Uncalibrated euclidean reconstruction: A review. In *Proceedings of Image and Vision Computing*, volume 18, pages 553–563, May 2000. Also available at <http://profs.sci.univr.it/~fusiello/papers/imavis1707.pdf>.
- [15] R. I. Hartley. In defence of the 8-point algorithm. In *ICCV '95: Proceedings of the Fifth International Conference on Computer Vision*, page 1064, Washington, DC, USA, 1995. IEEE Computer Society.
- [16] Martial Hebert and Takeo Kanade. Outdoor scene analysis using range data. In *Proc. 1986 IEEE International Conference on Robotics and Automation*, volume 3, pages 1426–1432, April 1986.
- [17] S. A. Hutchinson, G. D. Hager, and P. I. Corke. A tutorial on visual servo control. *IEEE Trans. Robotics and Automation*, 12(5):651–670, october 1996. Also available at <http://citeseer.ist.psu.edu/hutchinson96tutorial.html>.
- [18] Peter Kovesi. Matlab functions for computer vision and image analysis. Available at <http://www.csse.uwa.edu.au/~pk/Research/MatlabFns/index.html>.

- [19] A. Krupa, J. Gangloff, M. De Mathelin, G. Doignon Morel, L. Soler, J. Leroy, and J. Marescaux. Autonomous retrieval and positioning of surgical instruments in robotized laparoscopic surgery using visual servoing and laser pointers. In *In Proceedings of IEEE International Conference on Robotics and Automation*, volume 4, pages 3769–3774, 2002.
- [20] Y. Kuniyoshi, N. Kita, K. Sugimoto, S. Nakamura, and T. Suehiro. A foveated wide angle lens for active vision. In *Proceedings of IEEE International Conference on Robotics and Automation*, pages 2982–2985, Nagoya, Japan, 1995. Also available at <http://citeseer.csail.mit.edu/kuniyoshi95foveated.html>.
- [21] F. Lerasle and P. Danes. Projected light beams tracking for efficient 3d reconstruction. In *Proceedings of International Conference on Image Processing*, volume 3, pages 951–954, Thessaloniki, Greece, October 2001.
- [22] Ser-Nam Lim, Ahmed Elgammal, and Larry S. Davis. Image-based pan-tilt camera control in a multi-camera surveillance environment. In *Proceedings*, volume 1, pages 645–8. 2003 International Conference on Multimedia and Expo, july 2003.
- [23] H.C. Longuet-Higgins. A computer algorithm for reconstructing a scene from two projections. *Nature*, 293:133 – 135, September 1981.
- [24] F. Lu and E. Milios. Robot pose estimation in unknown environments by matching 2d range scans. In *CVPR94*, pages 935–938, 1994. Also available at <http://citeseer.csail.mit.edu/lu94robot.html>.
- [25] Alan M. McIvor. Calibration of a laser stripe profiler. In *Proceedings of Second International Conference on 3-D Digital Imaging and Modeling*, page 92, Los Alamitos, CA, USA, 1999. IEEE Computing Society.
- [26] USA Polytec. Ofv 056 scanning vibrometer. Available at <http://www.polytec.com/usa/>.
- [27] Hartley R.I. Estimation of relative camera positions for uncalibrated cameras. In *Applications of Invariance in Computer Vision*, pages 237–256, 1993.
- [28] Brian Scassellati. A binocular, foveated active vision system. Also available at <http://citeseer.csail.mit.edu/17100.html>.

- [29] Nutfield Technology. Open frame head 10mm. Available at <http://www.nutfieldtech.com/frameheads/ofh710.asp>.
- [30] Vladimir Vezhnevets. Enhanced matlab camera calibration toolbox, 2004. Available at <http://graphics.cs.msu.su/en/research/calibration/matlab.html>.
- [31] J. Woodfill, G. Gordon, and R. Buck. Tyzx DeepSea high speed stereo vision system. In *In Proceedings of the IEEE Computer Society Workshop on Real Time 3-D Sensors and Their Use*, Washington, D.C., June 2004. Conference on Computer Vision and Pattern Recognition.
- [32] John Woodfill and Brian Von Herzen. Real-time stereo vision on the PARTS reconfigurable computer. In Kenneth L. Pocek and Jeffrey Arnold, editors, *IEEE Symposium on FPGAs for Custom Computing Machines*, pages 201–210, Los Alamitos, CA, 1997. IEEE Computer Society Press. Also available at <http://citeseer.ist.psu.edu/woodfill97realtime.html>.
- [33] Zhengyou Zhang. Determining the epipolar geometry and its uncertainty: a review. *International Journal of Computer Vision* 27 (2), 1996. Also available at <http://research.microsoft.com/~zhang/Papers/IJCV-Review.pdf>.
- [34] Zhengyou Zhang. Parameter estimation techniques: A tutorial. *Image and Vision Computing Journal*, 15(1):59–76, 1997.

# Vita

Roman T. Stanchak

**Date of Birth** February 21, 1979

**Place of Birth** Randolph, Vermont

**Degrees** B.S. Computer Science, May 2001  
M.S. Computer Science, August 2005

August 2005