Kinematic Calibration of a Serial Robotic Arm Using a Linear Movement Constraint

by

Katie DiCola, **B. Eng.**

A thesis submitted to the
Faculty of Graduate Studies and Research
in partial fulfillment of the requirements for the degree of

Master of Applied Science in Mechanical Engineering

Ottawa-Carleton Institute for Mechanical and Aerospace Engineering
Department of Mechanical and Aerospace Engineering
Carleton University
Ottawa, Ontario
January, 2016

The undersigned hereby recommends to the
Faculty of Graduate Studies and Research
acceptance of the thesis

Kinematic Calibration of a Serial Robotic Arm Using a Linear Movement Constraint

submitted by **Katie DiCola**, **B. Eng.**

in partial fulfillment of the requirements for the degree of

Master of Applied Science in Mechanical Engineering

Professor M. J. D. Hayes, Thesis Supervisor

Professor Metin Yaras, Chair,
Department of Mechanical and Aerospace Engineering

Ottawa-Carleton Institute for Mechanical and Aerospace Engineering
Department of Mechanical and Aerospace Engineering
Carleton University
January, 2016

Abstract

Kinematic calibration of a robot arm is necessary for the performance of many tasks. The calibration process is frequently a tradeoff between affordability and accuracy. High accuracy calibration frequently requires expensive sensing equipment.

In this thesis, a method is developed to calibrate a robotic arm using minimal sensors. A simulation of the calibration process is developed that shows accurate determination of angular offsets in the robot model. These offsets account for the majority of error in robot movement. This method was performed on two industrial robotic arms: a Fanuc S-420iF and a Motoman MH-180. The joint angle offsets were determined in each case, though accuracy was lower in practice than in simulation.

The method was determined to be viable in order to determine joint angle offsets. With some modifications, it is also viable to determine link twist offsets.

This is the dedication...

Acknowledgments

I would like to acknowldege ....

Table of Contents

List of Tables

List of Figures

Chapter 1

Introduction

1.1 Motivation

As the prevalence of robot use increases, the necessity of calibration measures increases as well. Calibration is a method of reducing the error in a robot's movement without physically modifying the robot. Accuracy and small-scale precision are becoming more important for robotic manipulators. It would be impossible to perform at this level of precision without calibration. Many high precision tasks are now being accomplished with robotic arms [1]. Robotic manipulators can frequently be used for surgical procedures, in which minor errors can severely affect patients [2]. Precision manufacturing (such as small electronics) can also be accomplished with a robotic manipulator. In order to achieve this level of precision, manipulator calibration must be performed.

When manufacturing robots were first intruduced in factories in the 1960s, they were used for simple pick-and-place tasks, with poses that were taught manually. An operator would specify the exact motion the robot would perform, and then that motion would be performed repeatedly. This type of task is easily accomplished without calibration. As robotics became more ubiquitous in factories, more complex and exact motions were required. Manipulator calibration is essential for tasks that require a robot to move to a pose that has not been manually taught, such as any task that requires sensing a location and moving to that location. Calibration is a process by which the accuracy of the robot can be substantially increased by modifying only the software controlling the robot's movement. The goal of a calibration routine is to identify the small differences between the robot's geometry as it is recorded in the controller software and the actual physical dimensions of the robot.

There is typically a balance between expense and accuracy in calibration routines. Very high accuracy is expensive to achieve. The necessary measurement tools can never provide a perfectly accurate picture of the real robot geometry. The precision of measurements will affect the success of the calibration routine, and the accuracy of its results. A very high accuracy measurement, of any length, angle, or other geometric feature of a robot, can be very expensive. Calibration routines typically employ complex mathematics to reduce the necessary amount of measurements without substantially decreasing the accuracy of the calibration routine.

The calibration method discussed in this paper is an attempt to perform a reasonably accurate calibration with a small amount of sensors, and thus a less expensive overall procedure.

1.2 Objectives

There are three main objectives in this experiment:

- Use a simulation to verify the chosen method. This method is a modified form of an existing method: the virtual linear constraint [3] (discussed in Subsection **??**) with an additional constraint of rotation about this line (discussed in Chapter 3).
- Determine a physical setup that will allow for this constraint to be applied. In order to constrain rotational motion about this line, it must be measured. This method is described in Chapter 3.
- Use this setup to perform a calibration routine on a physical robot.

1.3 Statement of Originality

To the best of the knowledge of the author the following contributions presented in this thesis stem from original ideas and results.

1. Constraint of rotation about laser line during calibration using a cross-shape laser in place of a typical point-shape laser line.
2. Design of a two-camera assembly to read changes in robot's pose.

3. Design of an image processing algorithm allowing for precise measurement of small changes in position and orientation of a cross shape on an image.

4. Design of a movement pattern used to find robot poses that satisfy a constraint of constant orientation and linear movement.

1.4 Thesis Overview

- In Chapter 2, the theory behind robotic arm movement and kinematic calibration are discussed. Methods of calibration are discussed, from simple to complex. The methods upon which much of this thesis is based are identified and described.

- In Chapter 3, the overall experimental method is described. This includes the way in which data will be gathered from the robots, as well as the ways in which camera feedback will be analyzed.

- In Chapter 4, the simulated calibration is described fully. In the simple simulation, sample data is generated automatically to match the desired movement pattern of the robot, and the calibration is performed with this sample data. In the complex simulation, the full process of data gathering is simulated, including each robot movement, and the camera feedback.

- In Chapter 5, modifications to the experimental method given in Chapter 3 are discussed. These are modifications that are not included in the computer simulation. The physical experiment required several modifications to the overall process in order to function correctly.

- In Chapter 6, results are shown. This includes results of the simple and complex simulations, and experimental results. A discussion of these results is provided.

- In Chapter 7, the conclusions of this experiment are presented.

Chapter 2

Theory and Literature Review

This chapter provides the necessary background. A review of existing literature on calibration methods is also included. Original contributions are mentioned in context of existing literature, and literature contribution to original work is discussed.

2.1 Manipulator Kinematics

In order to calibrate a serial robot, the movement of the robot must be modeled. In this section, the basic kinematics used to represent robot movement will be discussed for the specific robot type on which the calibration is performed.

2.1.1 Types of Robotic Manipulators

There are a wide variety of robotic manipulators. While some resemble the structure of a human arm and hand, others may have substantially different structure. Parallel manipulators are supported by multiple kinematic chains [4]. A well-known example of a parallel manipulator is the Gough-Stewart Platform [5], a platform supported by several parallel controllable arms. These may offer higher accuracy [6], but have a smaller workspace and a more complex kinematic model. A serial manipulator has a more convetional form, usually with a shape resembling a human arm. These offer a larger workspace but overall lower accuracy, and lower stiffness [6]. A common type of serial manipulator has six revolute joints, though there are many configurations with different numbers and types of joints. The robots used in this thesis ([7], [8]) are both serial manipulators with six revolute joints. Some manipulators use a hybrid serial-parallel configuration [9], [10] in order to keep the advantages of a parallel

manipulator but extend the available workspace.

2.1.2 Joints

A joint is a way of constraining the relative motion between one link of a robot and another. A lower pair is a term used to describe a connection between two bodies which have surface contact relative to each other [11]. In order to accurately represent the movement of a manipulator, the structure of the manipulator needs to be fully described. The structure of a serial manipulator is a sequence of links connected by joints. There are a variety of different types of joints, with varying degrees of freedom. The two most common joint types are revolute joints and prismatic joints, each with a single degree of freedom. A revolute joint allows rotation about a single axis. A prismatic joint allows translation along a specific axis. Both of these are illustrated in Figure 2.1 (modified from [12]).

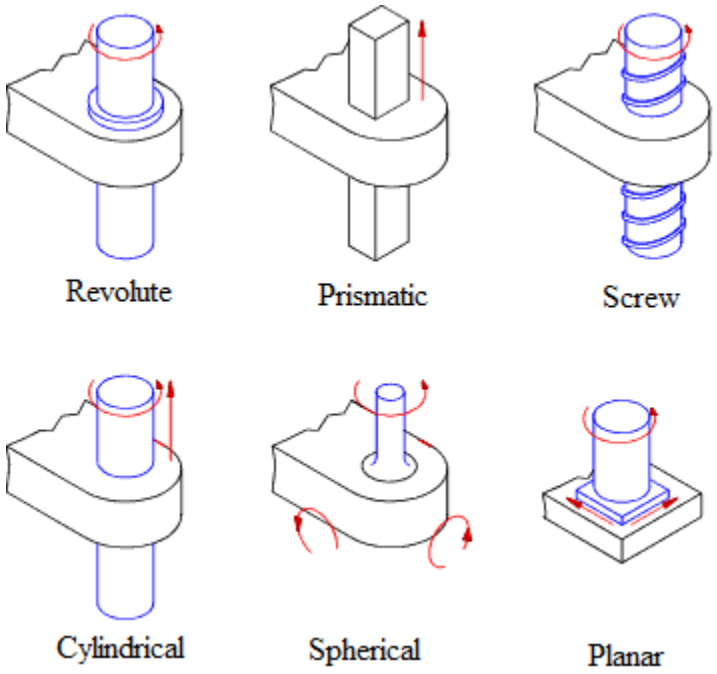

Figure 2.1: The Six Possible Lower Pairs

There are six types of lower pairs in total, with varying degrees of freedom, though the remaining four will not be discussed here. The robots used in this thesis have only revolute joints.

2.1.3 Attachment of Reference Frames to Links

The most commonly used representation of robot geometry, as created by Denavit & Hartenberg [13] involves attaching a reference frame to each link of the robot and using certain parameters to describe the relative position of these reference frames. For each link on the robot, there is a reference frame, attached in a specific position to describe the movement of the joint between the link and the surrounding links. The modified method for attaching these frames for a given link (link i) is given below. This procedure is as described by Craig [11]. The difference between the original and modified methods is described in Subsection 2.1.5

- The reference frame is attached rigidly to link i (i.e. the reference frame will move with the link)
- $\hat{Z}_i$ is placed such that it corresponds with the axis of rotation of the joint between link i and link i-1.
- The origin of the frame is placed at the intersection of $\hat{Z}_i$ and $\hat{X}_{i-1}$.
- $\hat{X}_i$ is placed along the common normal between $\hat{Z}_i$ and $\hat{Z}_{i+1}$ (or, normal to the plane created by $\hat{Z}_i$ and $\hat{Z}_{i+1}$, should they intersect)
- $\hat{Y}_i$ is placed following the right-hand rule, once $\hat{Z}_i$ and $\hat{X}_i$ have been placed.

A depiction of the frame attachment for an arbitrary link is shown in Figure 2.2.

2.1.4 Denavit-Hartenberg Parameters

Once the reference frames for each joint have been assigned, the structure and position of the robot can be described entirely through a set of parameters. These are known as Denavit-Hartenberg parameters (or simply DH parameters), of which there are 24 total for a six-axis serial arm [13]. For each link, there are four parameters. The definition of these parameters is given below:

- Link Length (a_i): The distance from $\hat{Z}_i$ to $\hat{Z}_{i+1}$, along $\hat{X}_i$.
- Link Offset (d_i): The distance from $\hat{X}_{i-1}$ to $\hat{X}_i$, along $\hat{Z}_i$.
- Link Twist (α_i): The angle from $\hat{Z}_i$ to $\hat{Z}_{i+1}$, about $\hat{X}_i$.

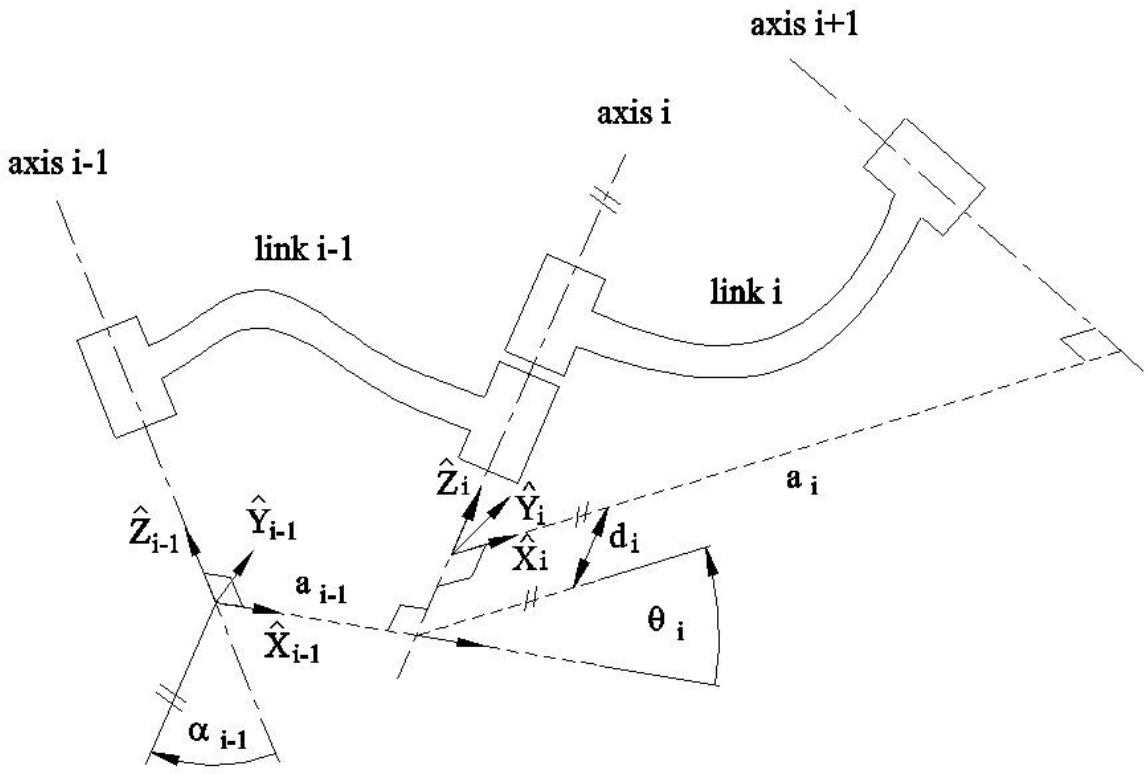

Figure 2.2: DH parameters and Reference Frame Attachment on a Link

- Joint Angle (θ_i): The angle from $\hat{X}_{i-1}$ to $\hat{X}_i$, about $\hat{Z}_i$.

The parameters are used to fully describe the position and orientation of each reference frame relative to the previous reference frame. An example of these parameters on a link is given in Figure 2.2. During the operation of the robot, movement is made by rotating the joints. This changes the joint angles (θ). The other 3 parameters (a, d, and α) are inherent in the construction of the robot, and ideally would remain constant for each link. In practice, there can be some variation in these parameters, which will be discussed further in Subsection 2.2.4.

2.1.5 Variation in DH Parametrizations

There are several variants of the DH parameter notation in use. The original notation given in [13] was developed primarily for use in single closed loop mechanisms (a closed loop mechanism is one such that the last link and the first link are connected to each other, forming a loop). Modified versions are better designed to work for an open loop mechanism such as a robot arm. These are referred to as modified DH parameters. All of the notation presented here uses modified DH parameters. A comparison of the original DH notation and two forms of modified DH parameters can be found in [14], including distal and proximal variants. The proximal variant is the form used here. "Distal" and "proximal" here describe the placement of the reference frames on links. The distal variant places frame i on the distal end of link i (where link i meets link i+1) and the proximal variant places frame i at the proximal end of link i (where link i meets link i-1). The primary advantage of the proximal variant is transparency. The parameters for link i are measured along the axes of frame i. In the original DH notation and the distal variant, some parameters are measured along axes of the previous joint. A disadvantage of the proximal variant is the use of multiple indices in calculating the position of the next reference frame. The values used in determining the transformation from frame i-1 to frame i are: θ_i,d_i,α_{i-1}, and a_{i-1}. In the original and distal variant, the required values would be θ_i,d_i,α_i, and a_i. These transformations are discussed in detail in Subsection 2.1.6.

2.1.6 Forward Kinematics

The method for determining a reference frame transformation, from the base frame to the end effector reference frame, is referred to as forward kinematics. The process

of forward kinematics takes the known DH parameters of a specific robot at a specific position, and uses them to calculate the position and orientation of the robot's end effector, usually in world frame coordinates. A set of transformation matrices is calculated, each one representing the transformation from one reference frame to the next. The transformation from frame i-1 to frame i is represented by ${}^{i-1}_{i}A$. ${}^{i-1}_{i}A$ can be calculated from DH parameters using Equation 2.1. (Note that here c and s are used as abbreviations for cos and sin).

$$ {}^{i-1}_{i}A = \begin{bmatrix} c\theta_i & -s\theta_i & 0 & a_{i-1} \\ s\theta_i c\alpha_{i-1} & c\theta_i c\alpha_{i-1} & -s\alpha_{i-1} & -s\alpha_{i-1}d_i \\ s\theta_i s\alpha_{i-1} & c\theta_i s\alpha_{i-1} & c\alpha_{i-1} & c\alpha_{i-1}d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{2.1} $$

For a robot with six joints, this will yield six matrices which, when multiplied together, form a transformation matrix from the base frame to the reference frame attached to the robot's end effector, as shown in Equation 2.2. There may, in some cases, be an additional transformation matrix to account for a specific tool attached to the robot's arm (A_{tool}). A_{tool} is specific to the tool attached and remains constant throughout the process of moving the robot.

$$ \begin{bmatrix} T_{11} & T_{12} & T_{13} & T_{14} \\ T_{21} & T_{22} & T_{23} & T_{24} \\ T_{31} & T_{32} & T_{33} & T_{34} \\ 0 & 0 & 0 & 1 \end{bmatrix} = T = {}^{0}_{1}A * {}^{1}_{2}A * {}^{2}_{3}A * {}^{3}_{4}A * {}^{4}_{5}A * {}^{5}_{6}A * A_{tool} \tag{2.2} $$

The position of the end effector, in base frame coordinates, is given by Equation 2.3.

$$ \begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = \begin{bmatrix} T_{14} \\ T_{24} \\ T_{34} \end{bmatrix} \tag{2.3} $$

The orientation of the end effector is also embedded in T. The orientation is

usually expressed using 3 angles, W, P, and R, where the top left 3x3 submatrix of T is the product of 3 elemental rotation matrices, as expressed in Equation 2.4, for example.

$$\begin{bmatrix} T_{11} & T_{12} & T_{13} \\ T_{21} & T_{22} & T_{23} \\ T_{31} & T_{32} & T_{33} \end{bmatrix} = Rz(R) * Ry(P) * Rz(W) \tag{2.4}$$

Where Rx, Ry, and Rz are basic 3-dimensional rotation matrices. Note that these equations use "atan2", a function available in many programming languages, including MATLAB (which was used here). This function calculates the four-quadrant inverse tangent, which accounts for the sign of the inputs to determine the result.

To find the values of WPR from T, Equations 2.5, 2.6, and 2.7 can be used.

$$P = atan2\left(-T_{31}, \sqrt{T_{11}^2 + T_{21}^2}\right) \tag{2.5}$$

$$W = atan2\left(\frac{T_{32}}{\cos(P)}, \frac{T_{33}}{\cos(P)}\right) \tag{2.6}$$

$$R = atan2\left(\frac{T_{21}}{\cos(P)}, \frac{T_{11}}{\cos(P)}\right) \tag{2.7}$$

This gives the final result of the forward kinematics. Starting with 18 parameters describing the robot's structure (a,d, and α values), the tool transformation matrix (A_{tool}), and the six joint angles (θ), the final position and orientation of the robot is determined (X,Y,Z,W,P,R).

2.1.7 Inverse Kinematics

Inverse kinematics is the process through which we determine the set of joint angles which will place the end effector at a specific position and orientation. This is the inverse of the forward kinematics. The known values are the 18 parameters describing the robot's structure (a,d, and α values), the tool transformation matrix (A_{tool}), and the desired end effector position and orientation (X Y Z W P R). The result of this computation will be the six joint angles (θ) required to place the end effector at this position and orientation.

The inverse kinematics for serial manipulators is substantially more mathematically complex than the forward kinematics. For any given end effector position, there will not necessarily be a single correct set of joint angles. For a robot with six revolute joints, there may be up to sixteen different solutions for a given pose [11]. There may also be no solution.

The general solution to this problem for a robot with six revolute joints is quite complex. Various complex methods exist for solving the general inverse kinematics, such as [15], [16]. However, certain special cases exist that allow for simplified solutions. The robots used here have a specific structure in which the last 3 joint axes intersect in a single point, referred to as the "wrist" point. For this special case, a simplified solution was created by Pieper [17]. The equations presented here are a modified version of those presented in [11].

The T matrix for the whole system must be calculated from the position and orientation (X Y Z W P R). This can be done using Equations 2.3 and 2.4 in Subsection 2.1.6. In this case, the T matrix is modified to exclude A_{tool}, as shown in Equation 2.8:

$$T = T_{actual} * A_{tool}^{-1} \tag{2.8}$$

From this new T matrix, the end effector's XYZ position (p_{ee}) can be taken as in Equation 2.3. Following this, the wrist point (p_{wrist}) is calculated using Equation 2.9:

$$p_{wrist} = p_{ee} - d_6 * T_z \tag{2.9}$$

where d_6 is the offset of the last link, and T_z is the z column of the rotation matrix in T (the first 3 entries of the third column of T).

The first angle, θ_1, is found simply using Equation 2.10:

$$\frac{p_{wrist,X}}{p_{wrist,Y}} = \frac{\cos(\theta_1)}{\sin(\theta_1)} \tag{2.10}$$

Next, the calculation for θ_3 is performed. It is more complex and requires several equations. First, the squared magnitude of the wrist position is calculated in Equation 2.11. Following this, the possible values of θ_3 are the solutions to Equation 2.12, which depends on k values and f values given in Equations 2.13 and 2.14:

$$r = p_{wrist,X}^2 + p_{wrist,Y}^2 + p_{wrist,Z}^2 \tag{2.11}$$

$$\frac{(r - k_3)^2}{4a_1^2} + \frac{(p_{wrist,Z} - k_4)^2}{\sin^2 \alpha_1} = k_1^2 + k_2^2 \tag{2.12}$$

$$\begin{aligned} k_1 &= f_1 + a_2 \\ k_2 &= \sin(\alpha_2) f_3 - \cos(\alpha_2) f_2 \\ k_3 &= f_1^2 + f_2^2 + f_3^2 + a_1^2 + a_2^2 + 2 f_1 a_2 \\ k_4 &= f_2 \cos(\alpha_1) \sin(\alpha_2) + f_3 \cos(\alpha_1) \cos(\alpha_2) \end{aligned} \tag{2.13}$$

$$\begin{aligned} f_1 &= \sin(\theta_3) \sin(\alpha_3) d_4 + a_3 \cos(\theta_3) \\ f_2 &= a_3 \sin(\theta_3) - \cos(\theta_3) \sin(\alpha_3) d_4 \\ f_3 &= \cos(\alpha_3) d_4 + d_3 \end{aligned} \tag{2.14}$$

Combining these f and k values in Equation 2.12 produces a large equation which relies on θ_3, but not on any of the other joint angles.

At this point, the calculation for θ_2 is performed. Only Equation 2.15 is required here. This equation is solved using all possible values of θ_1 and θ_3, to get all possible

sets of the first 3 joint angles.

$$\begin{aligned}
P_{wrist,y} =&((\sin(\theta_1)\cos(\theta_2)+\cos(\theta_1)\cos(\alpha_1)\sin(\theta_2))\cos(\theta_3)+(-\sin(\theta_1)\sin(\theta_2)\cos(\alpha_2)\\
&+\cos(\theta_1)\cos(\alpha_1)\cos(\theta_2)\cos(\alpha_2)-\cos(\theta_1)\sin(\alpha_1)\sin(\alpha_2))\sin(\theta_3))a_4\\
&+((\sin(\theta_1)\cos(\theta_2)+\cos(\theta_1)\cos(\alpha_1)\sin(\theta_2))\sin(\theta_3)\sin(\alpha_3)\\
&-(-\sin(\theta_1)\sin(\theta_2)\cos(\alpha_2)+\cos(\theta_1)\cos(\alpha_1)\cos(\theta_2)\cos(\alpha_2)\\
&-\cos(\theta_1)\sin(\alpha_1)\sin(\alpha_2))\cos(\theta_3)\sin(\alpha_3)+(\sin(\theta_1)\sin(\theta_2)\sin(\alpha_2)\\
&-\cos(\theta_1)\cos(\alpha_1)\cos(\theta_2)\sin(\alpha_2)-\cos(\theta_1)\sin(\alpha_1)\cos(\alpha_2))\cos(\alpha_3))d_4\\
&+(\sin(\theta_1)\cos(\theta_2)+\cos(\theta_1)\cos(\alpha_1)\sin(\theta_2))a_3\cos(\theta_3)\\
&+(-\sin(\theta_1)\sin(\theta_2)\cos(\alpha_2)+\cos(\theta_1)\cos(\alpha_1)\cos(\theta_2)\cos(\alpha_2)\\
&-\cos(\theta_1)\sin(\alpha_1)\sin(\alpha_2))a_3\sin(\theta_3)+(\sin(\theta_1)\sin(\theta_2)\sin(\alpha_2)\\
&-\cos(\theta_1)\cos(\alpha_1)\cos(\theta_2)\sin(\alpha_2)-\cos(\theta_1)\sin(\alpha_1)\cos(\alpha_2))d_3\\
&+\sin(\theta_1)a_2\cos(\theta_2)+\cos(\theta_1)\cos(\alpha_1)a_2\sin(\theta_2)-\cos(\theta_1)\sin(\alpha_1)d_2\\
&+a_1\sin(\theta_1)
\end{aligned} \tag{2.15}$$

This should result in a set of all possible values of θ_1, θ_2, and θ_3. There may be no possible combinations, in which case the pose is not reachable by the robot. There may be a single combination. There may also be multiple combinations, in which case it will be necessary to determine which one is the most well-suited to the task.

Once all possible sets of the first 3 joint angles have been determined, then the final 3 joint angles are relatively simple to determine. This is done by splitting the transformation matrix, T, into two parts. Equation 2.16 shows two transformation matrices, T_a, which accounts for the first 3 joints, and T_b, which accounts for the last 3 joints.

$$\begin{aligned}
T_a &= A_1A_2A_3\\
T_b &= A_4A_5A_6
\end{aligned} \tag{2.16}$$

The value of T has been calculated initially (in Equation 2.8). For each set of θ_1, θ_2, and θ_3, we are able to calculate T_a, using Equation 2.1 to calculate the A matrices. This allows us to calculate T_b, as in Equation 2.17.

$$T_b = (T_a)^{-1} * T \tag{2.17}$$

Given the nature of the wrist point, each of the last 3 joints contributes in a predictable way to the orientation of the tool frame. The values of θ_4, θ_5, and θ_6 are then able to be determined using Equations 2.18, 2.19, and 2.20.

$$\theta_5 = atan2\left(\sqrt{(T_{b,31})^2 + (T_{b,32})^2}, -T_{b,33}\right) \tag{2.18}$$

$$\theta_4 = atan2\left(\frac{T_{b,32}}{sin(\theta_5)}, \frac{T_{b,31}}{sin(\theta_5)}\right) \tag{2.19}$$

$$\theta_4 = atan2\left(\frac{T_{b,23}}{sin(\theta_5)}, \frac{T_{b,13}}{sin(\theta_5)}\right) \tag{2.20}$$

The result of this calculation is all possible sets of six joint angles that will allow the robot to reach the given tool position and orientation. If multiple possibilities are found, one must be selected. In the case of small movements, a good choice may be the set of joint angles which are closest in value to the angles of the previous pose.

2.2 Calibration

Calibration is a process through which the accuracy of a robot's movement can be substantially enhanced. In this section, the use of calibration and the conventional methods for calibration of a conventional serial robot arm are described.

2.2.1 Effect of Calibration

The kinematic equations presented above in Section 2.1 are able to mathematically describe the movement of a robot arm. In practice, there are many factors which can affect the accuracy of the results. Calibration is a process that can reduce some of these factors substantially, and increase the accuracy of the robot, using changes in software rather than physical changes to the robot.

In order to control a robot arm, a mathematical model of the arm must be used. The accuracy of movements depends on how closely the mathematical model represents the actual motion of the robot arm. The model described in Section 2.1 describes a "perfect" robot: joint lengths, offsets, and joint twist all manufactured exactly to

specifications, and all joints able to be moved to exact angles, unaffected by any errors in movement. In practice, a robot arm will be affected by all of these things. The calibration process aims to determine how these errors affect robot movement and change the model to better reflect the physical structure of the robot.

In [18], three levels of calibration are identified. Level 1 focuses on joint angles, Level 2 on the entire kinematic model, and Level 3 on the kinematic model as well as other factors.

2.2.2 Joint Angle Errors and Link Errors

Most of the error in a robot's pose is usually found in the joint angle offset values. The joint angle that a robot will record in its controller software is usually inaccurate, by a predictable amount. The error between the actual joint angle and the controller-recorded joint angle will remain relatively consistent, and this value is known as the joint angle offset, $\delta\theta$. Judd and Knasinski [19] found that joint angle offsets account for 90% of the root-mean-square (RMS) error of a robot's pose. As such, determining the values $\delta\theta$ for each joint can reduce error by approximately 90%.

Link error is similar to joint error in that we are looking for offsets. These are offsets in the remaining DH parameter values: α, a, and d. These values account for error in manufacturing of the links and deformation over time. This error accounts for about 5% of the RMS error of a robot's pose [19]. The remaining 5% is due to non-geometric error, discussed in Subsection 2.2.4.

2.2.3 Repeatability and Accuracy

In order to describe how accurately a robot is able to perform tasks, two different measures are used, repeatability and accuracy. These two measures are described well in [20].

The repeatability of a robot describes how closely a robot can return to a taught pose. Repeatability is expressed in units of length. A robot is moved to a given pose, this pose is recorded in the controller, and then after some additional task, the robot is returned to this recorded pose. The pose to which the robot returns will be within a certain distance of the initial pose. For example, a robot with a repeatability of 30 μm will be able to return to within 30μm of the initial pose. Because the robot is moving to a known pose, repeatability does not depend on the offset errors discussed

in Subsection 2.2.2. Many of the tasks that a robot arm is designed to perform are simple and repetitive, for instance, picking up and object in one location and placing it another, a "pick and place" task, or moving its tool along a known path. In these cases, the exact poses needed for the task are taught in advance. Thus, the repeatability is the only value that will affect these tasks.

The accuracy of a robot describes how closely the robot can move to a new pose in its workspace which has not been taught, but rather computed by the controller. An example task would be to move a robot's end effector to a specific point or along a specific path defined in the world coordinate system, without accurately teaching these positions in advance. In order to perform this task, the exact set of joint angles required for each pose in this path must be calculated. These sets of joint angles will depend on accurately knowing the actual values of the DH parameters. As such, offset error in the DH parameters can have a substantial effect on these tasks. In this thesis, accuracy, like repeatability, is expressed in units of length. It is defined as the maximum distance between the desired end effector position and the actual end effector position. A low accuracy value is well-suited to tasks in which the robot moves more autonomously: it is able to determine where it should move to and then make that movement.

If the joint angle errors and link errors were determined perfectly, then the repeatability and accuracy of a robot arm would be equal. The only error remaining in the robot's movement would be nongeometric.

2.2.4 Nongeometric Error

As mentioned in Subsection 2.2.2, offsets in joints and link properties account for approximately 95% of the total RMS error in the system. The remaining 5% is caused by nongeometric factors. As nongeometric errors are not considered in this analysis, only a brief description of possible sources of error is given here. Nongeometric error can come from many different sources, with each source contributing only minute parts of the total error. Nongeometric errors will affect the repeatability of a robot as well as its accuracy. Modeling error that is nongeometric can be more difficult than modeling geometric errors, as it frequently depends on external factors.

Thermal error can have a substantial effect on the repeatability of a robot. Changes in temperature can cause expansion and contraction in links, changing the overall structure of the robot. A robot working in a typical environment will have a

"warm up" period as the movement of the robot's motors warms up the nearby links. Leitner et al. [**?**] analyzed the repeatability of a robot and its warm up period. The warm up period was 500 minutes, and the fully warmed-up robot had a repeatability value of 0.02mm, a tenfold decrease from the repeatability value obtained when this warm-up period was neglected (0.2mm). This is in an environment of overall constant temperature. Some robots must operate in environments with significantly varying temperatures. Reducing thermal error can be done with a temperature sensor and a predictive model, as in [21].

In addition to thermal error, link shape may change due to heavy loads on the robot arm. The structure of a serial robot has several beam-like links in order. A heavy load placed at the end of the arm can cause bending in the links. This deflection will change dynamically as the robot moves. [22] Joints frequently have some backlash. Given the same target angle, a specific joint may arrive at slightly different locations depending on what direction the joint is rotating from. This is caused by backlash, inherent in the construction of the joint. Complex methods exist to model joint backlash [23]. Age may cause wear on robotic components, which can introduce additional error [24]

2.3 Methods of Calibration

There are many existing calibration methods. These methods vary in cost, accuracy, complexity, measurement methods, and processing of calibration data. Some known methods are included in this section.

These methods are generally split into two types: measuring the robot's position accurately (Subsection **??**), or constraining the robot's position in some way (Subsection **??**). These are called "open loop" and "closed loop" methods: methods in which the end effector can move freely, and methods in which the end effector is constrained, as if an additional joint were included between the end effector and the ground. Both methods require some way of accurately reading the pose of the robot. These measurements can be difficult to collect and require a trained operator and specific equipment [25].

An open loop method will require collecting a set of pose data: moving the robot to a given pose, recording the controller's joint angles, and recording either complete or partial pose measurements for each pose. A closed loop method will require moving

the robot to a set of various poses, all of which satisfy some constraint, and then recording the controller's joint angles at each pose.

2.3.1 Manual Joint Mastering with Indicators

This is the simplest method of calibration available. It is a closed loop method in which the joint angles of the robot are constrained. In this method, a certain position of the robot is known, in which some part of a link will align in some way with a part of the next link at a given joint angle. The operator of the robot will move that joint until these indicators are correctly aligned and then record that joint angle. The measured alignment can be dependent on the actual geometry of the link, or can be measured with specific indicator markings applied to the link. [18] The results of this process are very inaccurate compared with more complex calibration methods, as it relies on the operator judging the position of the robot simply by looking, and uses a single pose. The results may be further affected by additional problems, such as indicator marks being painted over, removed, or moved and incorrectly replaced. As such, this is considered a very rough calibration method for work that does not require high accuracy. This method is recommended in robot operation manuals [26].

2.3.2 Manual Joint Mastering with Precise Measurement

This is a closed loop method which involves moving the robot accurately to a single pose using a constructed jig with dial gauges. This method is also recommended in robot operation manuals [26]. In this method, a jig must be accurately manufactured and attached to the robot arm. Dial gauges on the jig are able to measure small differences in angle, allowing an operator to adjust the robot pose until it matches the desired reference pose. This process allows for greater precision than using indicator marks, as it involves precise measurement of angles rather than having the robot operator judge by sight. However, the results are only as precise as the jig itself, so this jig must be manufactured with very high precision. Only a single pose is measured in this method.

2.3.3 Full Pose Measurement

In an open loop method, the pose of the robot is measured either fully or partially. Ideally, the pose would be fully measured as this provides the maximum amount of

information with which to perform calibration. However, in practice partial measurement methods may be substantially more efficient. Most open loop methods use some form of partial measurement. The full pose of the robot is expressed in six variables: three for position and three for orientation. The method described in [27] uses a coordinate measuring machine (CMM) to measure the full pose of the robot.The procedure for measuring both the position and orientation variables involved a precision-manufactured tool attached to the robot end effector with several targets for the CMM to measure. In this case, the targets were five balls attached in a known configuration. Depending on the position of the end effector in the CMM's workspace, three to four of these balls were able to be measured. By measuring the position of all possible balls, the pose of the robot was measured. The ability to precisely measure the entire pose of the robot simplifies the calibration process. Calibration is done by comparing the actual measured pose of the robot with the pose stored in the robot controller over a number of poses. The technique described in [28] uses a CMM to position a calibration board. Calibration is performed by observing this board using a CCD camera mounted on the robot's end effector. A similar technique is used in [29], which is discussed further in Subsection 2.3.4 The technique described in [30] uses an ultrasonic range sensor. The sensor array used in this case has several sound emitters and several microphones. Distance between these is measured using the time-of-flight of an ultrasonic wave: by measuring the time difference between the emitter sending the ultrasonic wave and the microphone hearing it, distance can be measured accurately. This system must account for fluctuations in temperature and air pressure which can affect the local speed of sound. A complete sensor array is constructed using three microphones and several sound emitters. These sound emitters are placed in known locations in the base frame and also on specific links of the robot in order to measure the pose.

2.3.4 Relative Measurements

Using relative measurements is also an open loop method of calibration. Like using absolute measurements, this system may measure all of some of the robot's pose variables. However, these pose variables are not measured in the world coordinate system. Instead, each measurement is taken relative to the previous pose measurement. Relative position calibration measures the change in position between a pose

and a reference pose. The technique in [31] uses a ruler to measure position differences. Relative measurements are taken by comparing photos taken of the ruler at different poses as the robot moves. A similar method is described in [32], in which a camera mounted on the robot's end effector measures the position along a precision-manufactured ruler or straight edge. Additionally, a stereo laser system is used to measure the distance from the straightedge to the tool. In [29], both relative position and orientation are measured. This is done using a camera mounted on the robot's end effector and a reference object that can be recognized easily by analyzing the images from this camera. In this case, the object is a chess board. Another system using relative measurements is [33]. This system uses a metallic grid and a laser distance sensor. The laser distance sensor is mounted on the robot and can sense the distance towards the grid plate with high precision. Additionally, the laser allows for the position on the grid plate to be measured, though not with the same high level of accuracy as the distance.

Once the relative pose measurements have been taken, the calibration can be performed using a Jacobian matrix and singular value decomposition (SVD). This process is well described in [34].

The system is represented with Equation 2.21

$$\Delta x = J_{rel}\Delta\zeta \tag{2.21}$$

Where J_{rel} is the relative Jacobian, and Δx is the measure of relative position and orientation. For each point other than point 1,Δx is the ith relative pose (the difference between pose i and a refence pose). It is shown in Equation 2.22.

$$\Delta x_i = \begin{bmatrix} \Delta x_i \\ \Delta y_i \\ \Delta z_i \\ \Delta w_i \\ \Delta p_i \\ \Delta r_i \end{bmatrix} \tag{2.22}$$

Results of this equation are stacked to produce a single vector of length $6(n-1)$

(where n is the total number of points).

The Jacobian is calculated for all points. It is composed of partial derivatives measuring the effect of changing any one parameter on the overall equation. There are 24 total partial derivatives for each point (six each for the joint angles, joint twists, link lengths, and link offsets). For any given parameters (for example, the joint angles) the Jacobian is calculated using Equation

$$J_\theta = \begin{bmatrix} \frac{\delta x}{\delta\theta_1} & \frac{\delta x}{\delta\theta_2} & \frac{\delta x}{\delta\theta_3} & \frac{\delta x}{\delta\theta_4} & \frac{\delta x}{\delta\theta_5} & \frac{\delta x}{\delta\theta_6} \\ \frac{\delta y}{\delta\theta_1} & \frac{\delta y}{\delta\theta_2} & \frac{\delta y}{\delta\theta_3} & \frac{\delta y}{\delta\theta_4} & \frac{\delta y}{\delta\theta_5} & \frac{\delta y}{\delta\theta_6} \\ \frac{\delta z}{\delta\theta_1} & \frac{\delta z}{\delta\theta_2} & \frac{\delta z}{\delta\theta_3} & \frac{\delta z}{\delta\theta_4} & \frac{\delta z}{\delta\theta_5} & \frac{\delta z}{\delta\theta_6} \\ \frac{\delta w}{\delta\theta_1} & \frac{\delta w}{\delta\theta_2} & \frac{\delta w}{\delta\theta_3} & \frac{\delta w}{\delta\theta_4} & \frac{\delta w}{\delta\theta_5} & \frac{\delta w}{\delta\theta_6} \\ \frac{\delta p}{\delta\theta_1} & \frac{\delta p}{\delta\theta_2} & \frac{\delta p}{\delta\theta_3} & \frac{\delta p}{\delta\theta_4} & \frac{\delta p}{\delta\theta_5} & \frac{\delta p}{\delta\theta_6} \\ \frac{\delta r}{\delta\theta_1} & \frac{\delta r}{\delta\theta_2} & \frac{\delta r}{\delta\theta_3} & \frac{\delta r}{\delta\theta_4} & \frac{\delta r}{\delta\theta_5} & \frac{\delta r}{\delta\theta_6} \end{bmatrix} \tag{2.23}$$

Each of these partial derivatives must be solved using any symbolic math software. This allows us to calculate the numerical value of the Jacobian directly using the parameter values at any point, without performing a derivation each time.

The full Jacobian matrix of any point is given in Equation 2.24.

$$J_i = \begin{bmatrix} J_\theta & J_\alpha & J_a & J_d \end{bmatrix} \tag{2.24}$$

The relative Jacobian of a point other than point 1 is given in Equation 2.25

$$J_{rel,i} = J_i - J_1 \tag{2.25}$$

And the full Jacobian matrix is constructed by stacking the relative Jacobian of every point other than point 1. The final size of this matrix is $6(n-1) \times 24$.

In this case, we have more equations than unknowns, and a least squares approximation must be performed using singular value decomposition. The results of this are estimates of all 24 error offsets.

SVD is a technique used to solve a linear least-squares problem [35]. In this case, by taking many pose measurements, we have an overdetermined system, and thus the solution will be a linear least-squares approximation. Functions to perform SVD are frequently included in mathematical software, (such as in Matlab's optimization

toolbox). This will take J_{rel} and decompose into 3 matrices: U, S, and V. In this way, the pseudo-inverse of J_{rel} which gives a least-square approximation of this overdetermined system is given by:

$$J^{+} = (V)(S^{-1})(U^{T}) \tag{2.26}$$

Given that S is a diagonal matrix, the inverse of S can be calculated easily. Each diagonal element in S^{-1} is simply the inverse of that element in S. However, values of S that are near zero must be eliminated as they contribute to substantial error [31].

The result of this is an approximation of the inverse of the Jacobian matrix given in Equation 2.21, which allows for a solution to that equation, to find the error values for each parameter.

2.3.5 Measurement using Theodolites

One measuring technique frequently used to determine end-effector position is the use of theodolites. These were originally used as surveying tools as early as the 1500s, and modern versions allow for high precision measurement [36]. A basic theodolite consists of a telescope mounted in such a way that it can be adjusted in a horizontal or vertical direction, and the angle of the telescope can be very accurately measured. When the telescope is pointed at a distant object, these angles allow us to accurately measure the exact direction to that object. While typical theodolites used for surveying are designed to measure distant objects, the concept can also be used to measure closer-range objects such as the poses of a robot in its workspace. These are automatic theodolites, which do not require manual adjustment. A method is described in [37] in which a combination theodolite and vision system is used to determine the line of sight to an target object on the robot's end effector to measure the pose. The rotation of the theodolite is low resolution, and so must be combined with the attached vision system to provide precise angle measurements. The target object is illuminated in order to be seen easily. In this way, a partial pose measurement is taken. This theodolite and vision system must itself be calibrated in order for precise measurements to be taken [38]. A different method using theodolites is described in [39]. This measurement uses triangulation combined with an interferometer to determine full position measurements. Two theodolites are used, and therefore the intersection of the two lines of sight must be the object. The accuracy of triangulation

depends on precision knowledge of the position of each theodolite, thus calibration must be performed on the measurement system as well. The method described in [40] is similar to this, although it uses two automatic laser tracking interferometers.

2.3.6 Fixed Pose Closed Loop Methods

While measuring the position of the end effector in space is very useful for calibration, it is not always possible or efficient to have a full measurement system. Closed loop systems rely on constraining the motion of the robot in some predictable way in space. With this constraint applied, the robot, which on its own is an open kinematic chain, can be treated as a closed kinematic chain. In many cases, motion can be very precisely constrained with mininal sensing equipment when compared with the open loop methods discussed above. Calibration can be performed using many different types of constraints. The simplest constraint is a completely fixed end point [41]. While this completely restricts the motion of a typical 6 degree-of-freedom robot, it is useful in the calibration of "redundant" robots - those with additional links. This can also be used to model two manipulators attached together at the end as a single closed kinematic chain. For a 6 degree-of-freedom robot a less restrictive constraint is required so that many different poses can be selected. One type of constraint is a peg attached to the end effector must be inserted into various holes in known locations. The system in [42] uses a precision manufactured metal dime with several holes in it. The location of these holes relative to each other is well known. This gives several possible positions for the robot, and additionally constains the orientation, although not completely, as the robot is still able to rotate about the axis of the peg. A similar system is given in [41], in which the robot's motion is described as "opening a door", free to rotate in only one axis. These systems constrain 5 degrees of freedom, leaving the robot free to rotate in one axis only. The system in [43] uses a similar constructed object, a "calibration cube", with many possible holes. However, this system uses a tool with two pegs which must be simultaneously inserted into adjacent holes. This prevents rotation about the peg axis, resulting in a system of many possible poses, each fully constrained (fixed in position and orientation). The method described in [44] uses a much more complex end point constraint. This method uses a constructed hexapod device with six telescoping ballbars, which attaches to the robot's tool with magnetic contacts. This device can be taken apart and rearranged into several configurations, resulting in a total of 72

possible fixed poses of the robot.

2.3.7 Closed Loop Methods with Fewer Constraints

Many closed loop methods have far less restrictive constraints on the robot's pose. These methods may constrain robot motion in as little as one degree of freedom. These constraints may still be used to perform calibration. In [45], a system is used in which the only constraint is point contact with a plane in a known location. This constrains only one degree of freedom. A probe is mounted on the robot's end effector, and the robot is constrained to move such that this probe is touching a precision-manufactured plane. Another possible method uses multiple planes, as a single plane may not always be enough for a full calibration [46]. A point contact constraint restricts the motion of the robot such that some part of the robot's tool must remain at a given point. However, the orientation can vary. This can be expressed as a ball joint, which allows full rotation but no translation. As such, 3 degrees of freedom are constrained. In [47] and [48], a point contact constraint is used to calibrate a robot.

2.3.8 Closed Loop Methods with Laser Line

A common method for constraining the end effector motion is by attaching a laser pointer to the robot, which must always be pointing at one given location. The calibration method used in this thesis features the use of a laser line. The laser dot must be either measured by some device or judged visually by the operator [49]. While judging by eye produces less precise measurements, pointing the laser at a distant object magnifies the effect of orientation errors, so this is more precise that visually judging the robot's joint angles as in Subsection 2.3.1. Using a camera or other sensor provides much more precise measurements. In [50], a system is proposed that uses a position sensitive detector (PSD) to detect the laser dot. The advantages of a PSD over other camera types are speed and accuracy compared to cameras of similar cost. However, PSDs can only sense a single dot of light. With the use of a simple laser pointer, use of a PSD is advantageous, however in a situation where a more complex shape must be observed, such as in this thesis, PSDs cannot be used. This is discussed in Chapter 3. In [3], an assembly of 2 PSDs is used to further constrain the motion of the robot. By reflecting the laser line from one PSD to the other and measuring both, a system can be set up in which the laser line

orientation must remain constant, and the robot must move only along the laser line. The calibration method performed in this thesis is a similar laser-line based method with an assembly of two cameras and a beam splitter to read the laser line, rather than two PSDs. This assembly is able to provide the same constraint on the robot's pose as above, and additionally constrain rotation about the laser line axis.

Chapter 3

Experimental Method

This chapter describes the basic experimental method used for robot calibration in this thesis. A laser is used to enforce a linear constraint, similar to methods discussed in Subsection **??**. The laser is attached to the robot's end effector, which is then instructed to point the laser into a box containing an assembly of two cameras, which can read the position and direction of the laser line. These methods are modified in order to fully constrain the end effector orientation. To constrain rotation about the laser axis, an optical device is applied over the laser attachment such that it projects a cross shape, rather than simply a dot. By keeping the orientation of this cross constant, rotation will be full constrained. However, the use of PSDs (Position Sensitive Devices) is no longer possible when sensing a cross shape rather than a dot, as these devices can only sense a single light spot. Hence, an assembly of two cameras is used.

3.1 Robots

This experiment was performed with 3 different robots: two of the same model, FANUC S-420iF [7], shown in Figure 3.1, and one Yaskawa Motoman MH180 [8], shown in Figure 3.2. The properties of these robots are described in Table 3.1.

3.2 Camera Assembly

The ultimate goals of the camera assembly are:

Robot	FANUC S-420iF	Motoman MH180
Maximum Reach (horizontal)	2.852 m	2.702 m
Maximum Reach (vertical, above base)	3.034 m	3.061 m
Maximum Payload	120 kg	180 kg
Repeatability	±0.4 mm	±0.2 mm

Table 3.1: Table of Robot Properties

Figure 3.1: Fanuc S-420iF Robot Arm

Figure 3.2: Motoman MH-180 Robot Arm

- to measure the location of a point on the laser line;
- to determine the angle of the laser line in space;
- to measure the rotation of the robot's end effector about the axis of the laser line;

The goal of the calibration routine will be to adjust robot pose to keep these values as consistent as possible. This will simulate an end-effector constraint in which the end effector orientation must remain completely consistent and the movement of the robot must be along one specific axis: that of the laser line.

3.2.1 Geometry of Camera Assembly

Two cameras must be set up in such a way that both can sense the laser line. A beam splitter is used for this purpose. The beam splitter will divide the laser line into two: one beam is transmitted directly in its original direction and the other is reflected. By putting a beam splitter at a 45 degree angle, the original laser cross will be projected in two directions, at 90 degrees from each other.

By placing the two cameras at different distances from the beam splitter, this system will effectively measure two different points on the laser line, which allows us to determine the angle of the laser line in space. The function of this setup is shown in Figure 3.3. The nominal (or desired) laser line is represented as a thicker line. This is a representation of a laser line perfectly orthogonal to the first camera plane. Upon reaching the beam splitter, it splits into two lines, each of which intersects a camera plane directly in the centre. One other line is provided to demonstrate how this camera setup reads different laser lines. With a single camera setup, only a single point on the laser line would be seen. It would be impossible to distinguish two lines at different orientations if they intersected the camera plane at the same point. In this way, the two lines shown in Figure 3.3 would be indistinguishable with only one camera, as they both intersect Camera Plane 1 at the same point. With the addition of a second camera, these lines can be differentiated, as they intersect Camera Plane 2 at different locations.

As the image reaching the second camera is a direct reflection, this image will be reversed in processing. This reversed image will serve the same function as if an image was taken at the location of the imaginary camera plane. This reflected setup

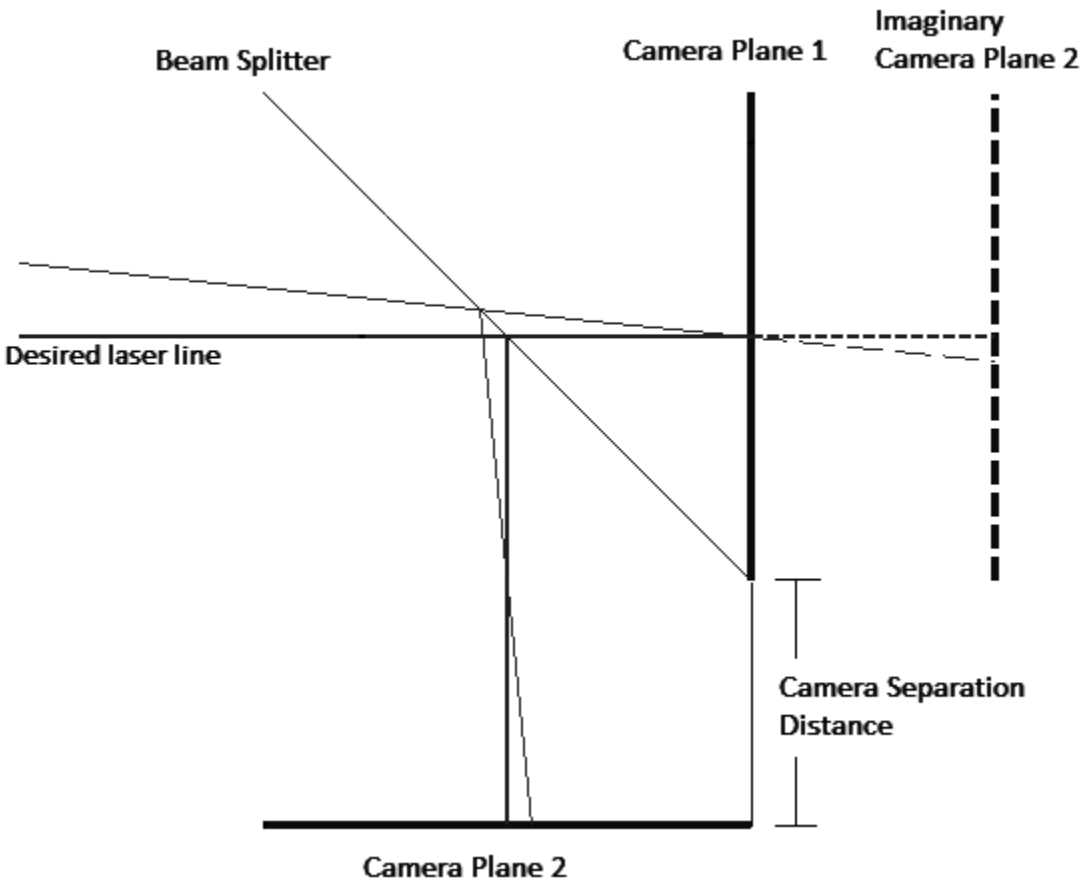

Figure 3.3: Use of Beam Splitter and Two Cameras

allows us to effectively measure two points along the line. The camera separation distance must be non-zero in this case, so that the two cameras are not measuring the same image.

The rotation about the laser axis can be determined simply using the orientation of the cross on the image taken from either camera.

3.2.2 Physical Setup of Camera Assembly

The final assembly was constructed to fit inside a sealed metal box, to reduce dust on the sensitive optical components and to reduce ambient light in the images. Figure 3.4 gives a top view of the assembled box. Figure 3.5 shows a photo of the completed

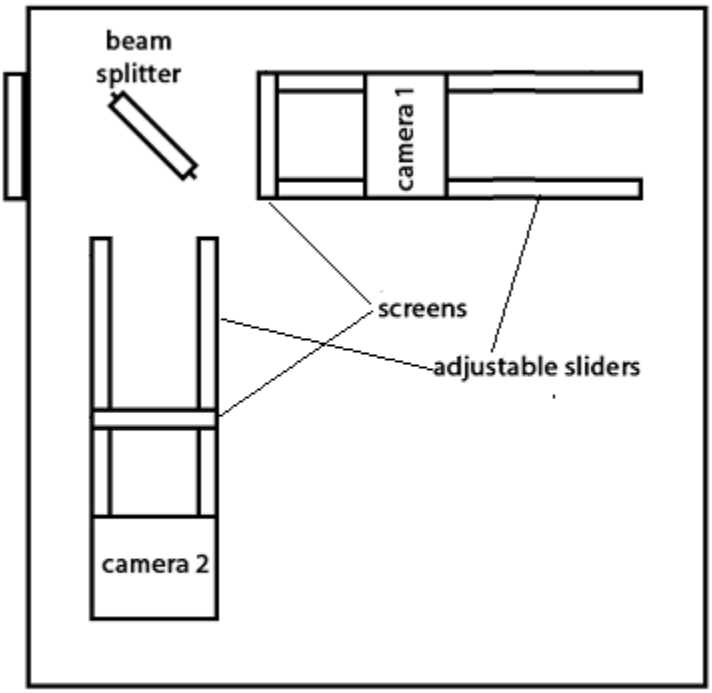

Figure 3.4: Constructed Camera Assembly

Figure 3.5: Photo of Camera Assembly

box.

On the left side of the box is a window through which the beam enters. The beam splitter is placed directly inside this window, at a 45 degree angle. The two cameras are placed in the direction of the laser line. Each camera is mounted such that its distance from the beam splitter can be adjusted through the use of sliders. This allows us to change the camera separation distance (as described previously in Subsection 3.2.1). Allowing the laser to enter the cameras directly does not provide an acceptable image, as too much light enters the camera. To account for this, a screen is mounted to the camera assembly. It is attached such that the distance between the camera and the screen remains constant regardless of the position of the camera. These mylar screens prevent the laser light from entering the camera directly. Instead, the cross is projected on the screen and the camera is taking an image of this projected cross. Mylar is used instead of plain paper to avoid some of the texture of the paper showing up in the images.

The full manufacturer specifications for the cameras are available in [51]. CMOS cameras were chosen over CCD cameras as they provide greater speed for a simple image such as the cross-shape image [52]. No colour is required in the image, so a monochrome camera was selected. A red filter is attached to the camera for ease

of sensing the red-coloured laser line. The cameras provide a high resolution image, with a 5 megapixel resolution (1920 pixels vertically and 2560 pixels horizontally). The cameras communicate with the computer over USB.

The focus and aperture settings of the cameras were manually adjusted. The aperture was set such that the cross appears large in the image, but does not bleed outwards into the black space in the rest of the image. The focus settings were set such that the cross appeared somewhat blurred. This was done because a perfectly focussed camera was able to see the texture of the mylar screen, while the slightly out-of-focus image eliminated this effect. The cross filter on the laser created an effect which made the cross look as if it was made up of many small dots of light, rather than two lines. The slightly out-of-focus cameras eliminated appearance of this effect in the image. The cameras were connected with the computer over USB.

3.3 Image Analysis

This section describes the process of obtaining an image from the cameras and extracting the necessary information from that image. An example of an input image used in this process can be seen in Figure 3.6

The accuracy of the calibration depends heavily on the accuracy of the image analysis. Because of this, the image analysis is a multi-stage process meant to increase the overall accuracy. The image is given some initial processing, followed by an estimation of the cross centre and angle using Hough transform. Next, this approximation is used to generate regions of interest (ROI), in which a moment algorithm is performed to find the cross centre location and angle.

3.3.1 Initial Processing

There are some preparatory steps that must be taken before analyzing the cross positions. First, the image is converted to black and white. The result of this is seen in Figure 3.7 In Figure 3.6, two shapes can be seen on either side that are detached from the main cross shape. These appear only on the images taken with camera 1, and occur due to the mounting of the beam splitter. The beam splitter is small enough that the edges of the cross can be seen around it on camera 1. These

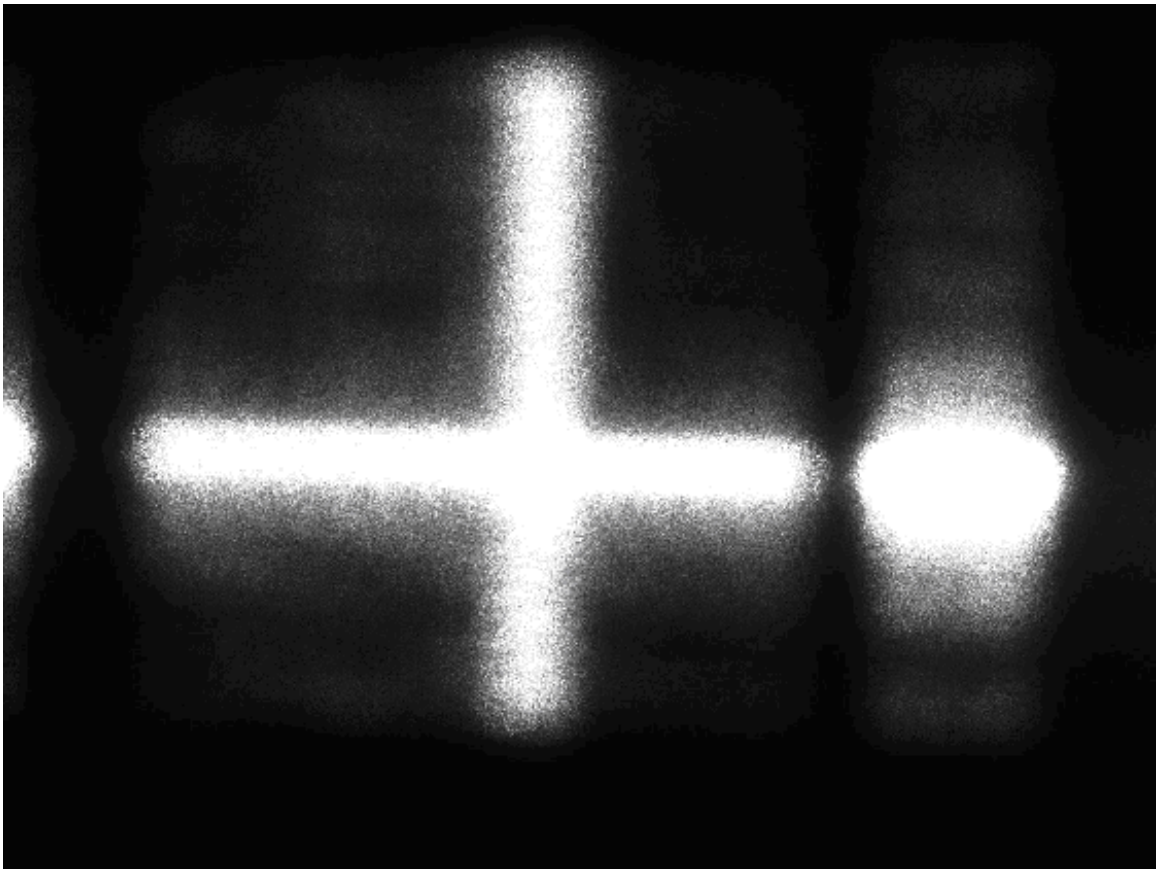

Figure 3.6: Input Image

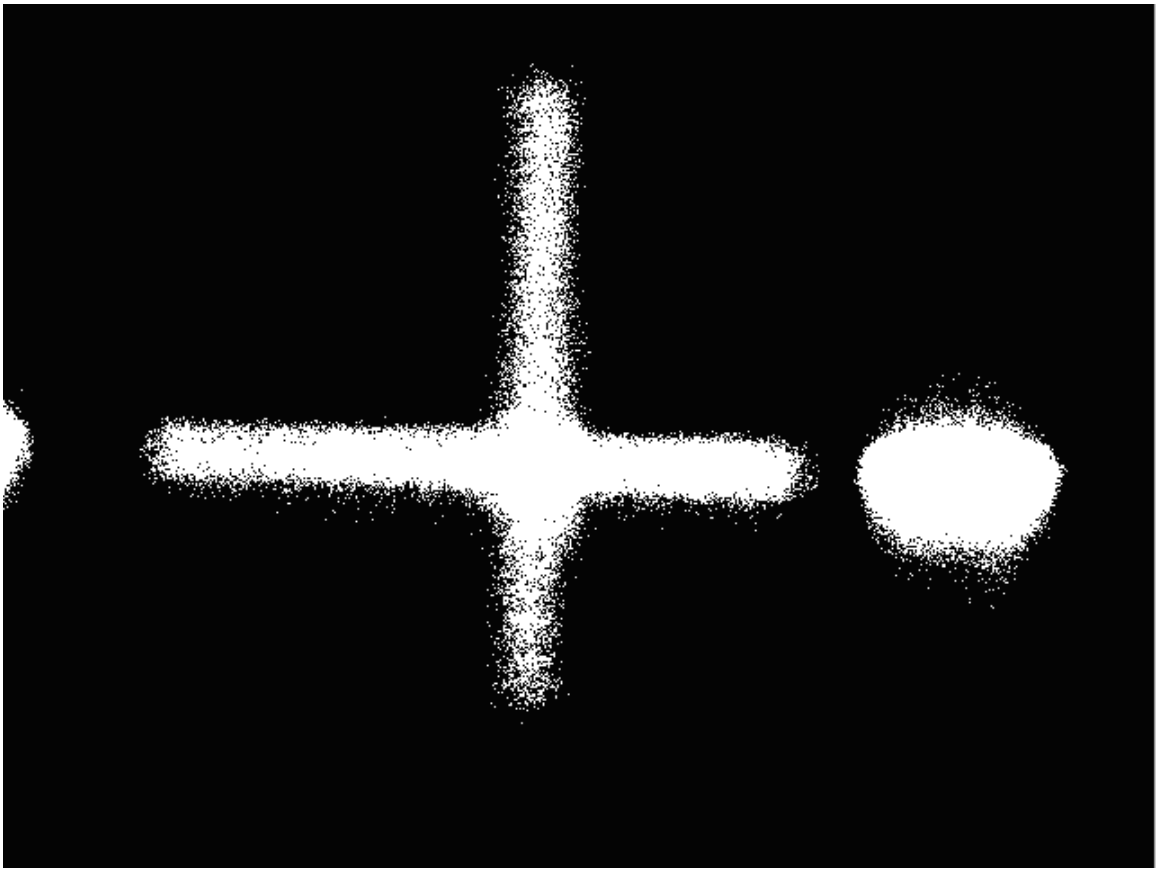

Figure 3.7: Input Image Converted to Black and White

edges must be removed for analysis. These are simply removed by determining the horizontal location of the outer edges of the central cross. While analyzing the image, everything outside of this horizontal area is ignored (treated as a simple black pixel).

3.3.2 Initial Estimate Using Hough Matrix

An estimate of the cross centre and angles is done using the Hough matrix. This technique was originally created by in the 1960s Paul Hough [53] as a method of identifying straight lines in images. The computational power and memory required to calculate the Hough matrix were restrictive at the time [54], but with today's computers this is trivial. The Hough transform allows us to determine the likeliness of a line in an image at any given location and angle. Later, a different parametrization was created [55], which uses r and θ rather than x and y, which allows for the detection of vertical lines. The value of θ represents the angle of the line, counterclockwise from the vertical axis. The value of r represents the perpendicular distance between the line and the origin. A more general version of this technique [56] can also be used to detect other shapes. The Hough transform allows to generate a matrix and determine which values of r and θ are likeliest to contain a line. It results in a two dimensional matrix of values (the Hough matrix), where the highest values represent the most likely places for a line to be, in the image. Figure 3.8 shows the Hough matrix of a black and white cross image similar to Figure 3.7. The white areas of Figure 3.8 represent high values.

Two peaks are located on this graph at approximately 90 degrees apart. These represent an approximation of the two lines making up the cross. They are represented in Figure 3.8 as small circles. By taking these line locations and finding their intersection, an approximation of the cross centre is made. The angle of the near-vertical line is used as an approximation of the cross angle. Because of the thickness of the lines of the cross on the image, further analysis is used to refine the cross centre location detection. The centre and angle values determined using the Hough transform allow us to locate regions of interest for the moment algorithm.

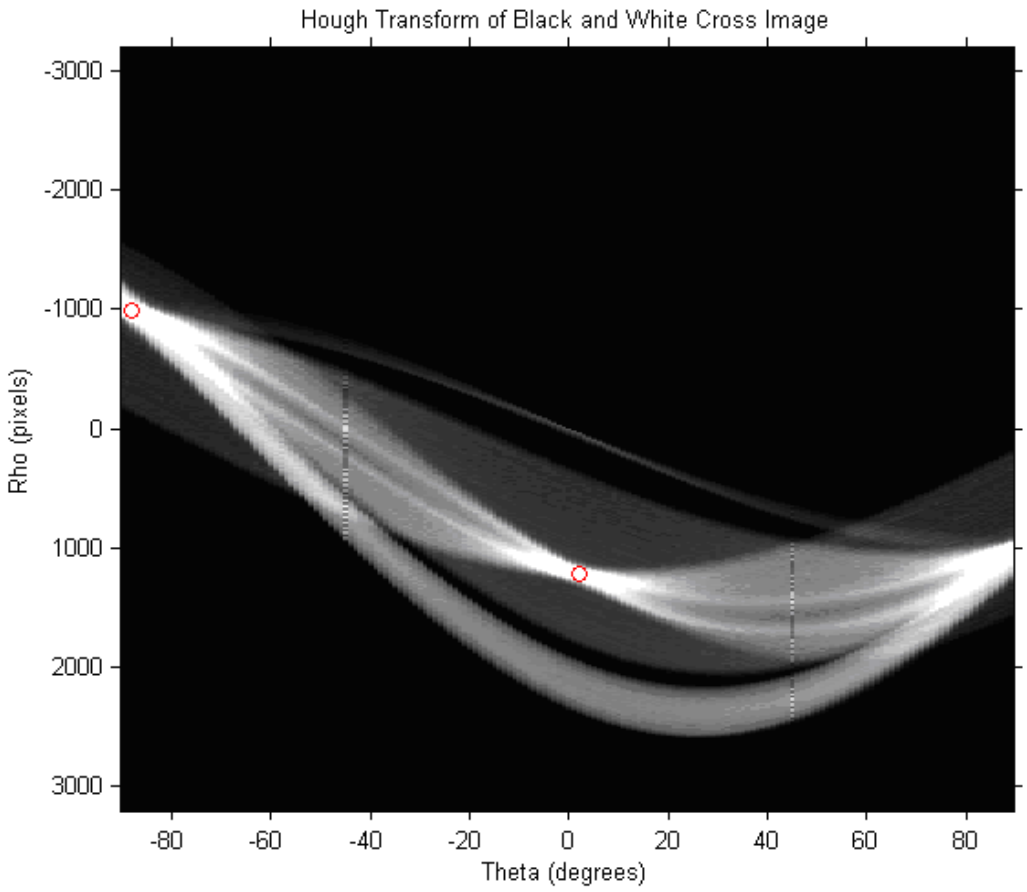

Figure 3.8: Result of Hough Transform on Cross Image

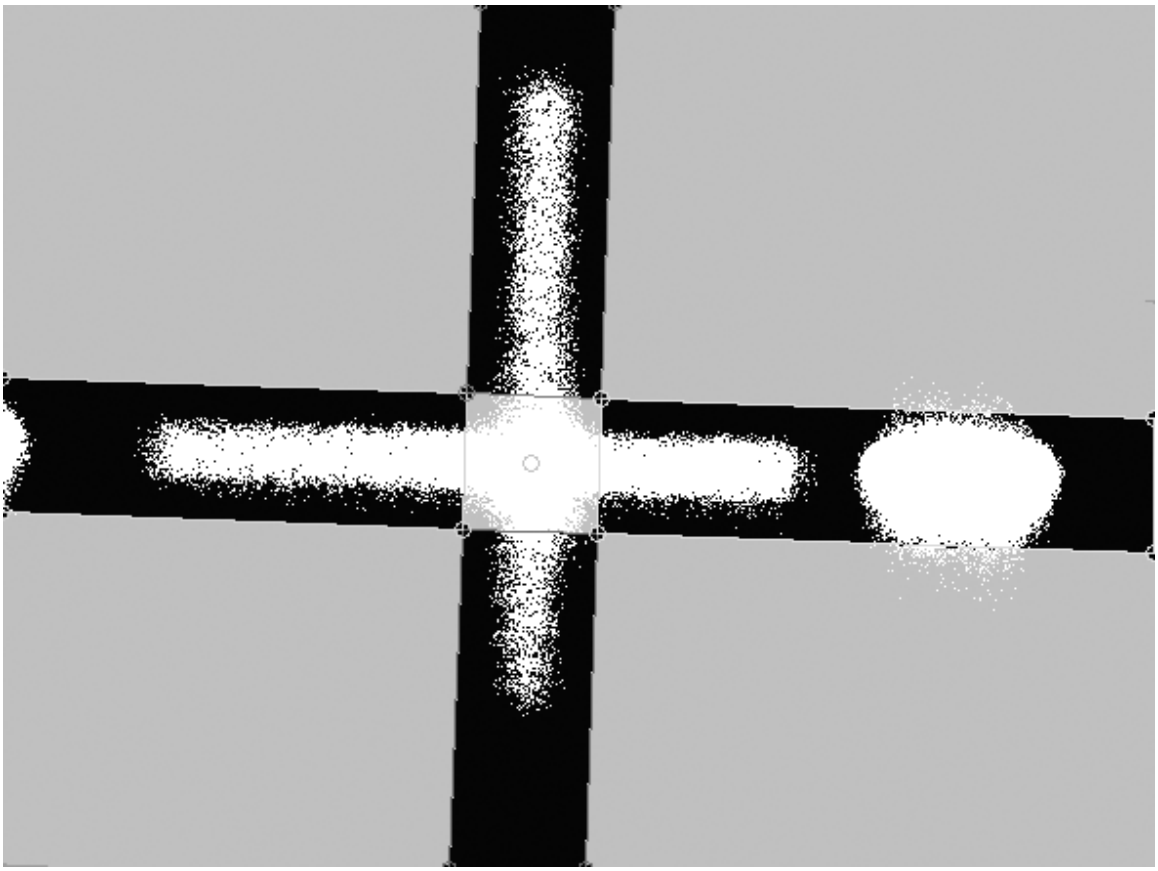

Figure 3.9: Regions of Interest on Cross Image

3.3.3 Finding Regions of Interest (ROI)

The selection of the regions of interest is shown in Figure 3.9. It shows four ROI, each one enclosing one "arm" of the cross. The centre area is not part of any ROI.

The shape of these ROI is specifically chosen to fulfill the following criteria:

- None of the centre area of the cross will be included.
- Each ROI will fully enclose one "arm" of the cross, so that all white pixels of that arm will be included in the analysis.
- The width of the ROI will not be excessively large, as this increases the time required to perform the moment algorithm.
- Each ROI is designated as horizontal or vertical, depending on the orientation of the line. In all image analysis cases, the cross orientation only changes minimally, thus during analysis there will always be two approximately horizontal ROI and two approximately vertical ROI.

Note that, as mentioned in Subsection 3.3.1, the white edges detached from the main cross shape will not be included in the analysis. Though they fall within the ROI, they are treated as black pixels.

3.3.4 Using the Moment Algorithm to Find the Exact Centre and Angle

The moment algorithm found in [57] is used here. This algorithm works by calculating a y value for each column of pixels at a specific x value. This y value is found using Equation 3.1

$$\frac{\sum_{y=1}^{y_{max}} y \cdot I^p(x,y)}{\sum_{y=1}^{y_{max}} I^p(x,y)} \tag{3.1}$$

Where I(x,y) represents the intensity at the location (x,y). The variable p is used as a power factor which allows for some control over the function depending on the use. Here, the value of p remains at 1.

This is performed over the two horizontal ROI. The vertical ROI are evaluated with the same equation, with the x and y switched. This equation was found to be slightly inaccurate at the far edges of the ROI, so a total of 50 points are removed

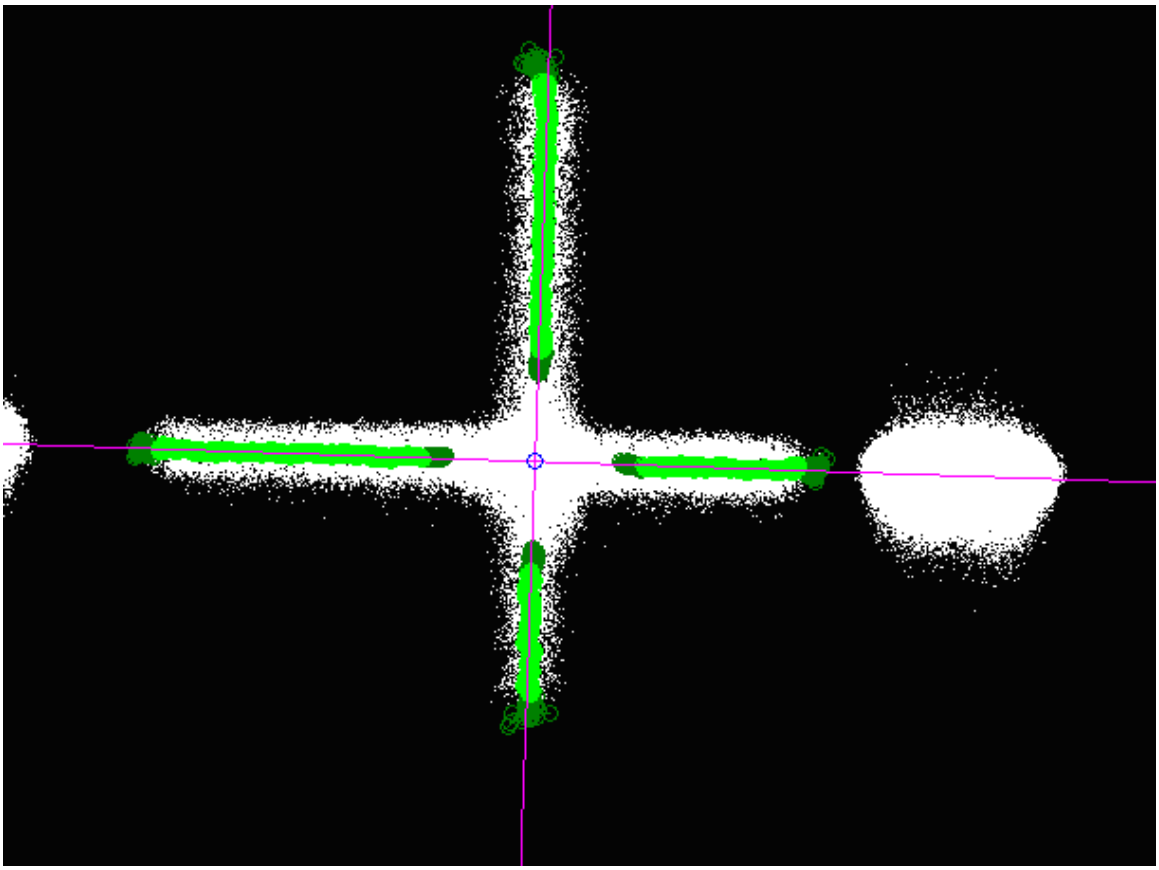

Figure 3.10: Final Cross Centre Determination

from each end. The remainder of the two horizontal ROI are merged into one. The vertical ROI are similarly merged. This results in two data sets: one set of points representing the horizontal portion of the cross, and one representing the vertical. This can be seen in Figure 3.10. In this, the thicker lines represent the points used to calculate the best fit line. The darker portions of these data sets are the 50 discarded points.

The final line locations are determined using a simple best-fit calculation on each. (The vertical line is rotated 90 degrees before the best fit calculation is performed as best fit does not work well with near vertical lines). This gives two straight lines on the image. These are also shown over the data sets in Figure 3.10. The final cross centre location is given by the intersection of these lines.

Chapter 4

Simulation of Calibration Procedure

4.1 Basic Simulation

In this simulation, measurement error and noise are not considered. The sample data set which is generated and analyzed is an ideal example. The only error in this model is the offsets in the 24 DH parameters.

4.1.1 Creating a Simulated Data Set

The system we are simulating here is able to measure and constrain the motion of the robot such that the following conditions are met:

- The orientation of the end effector remains constant.
- The movement of the robot remains along a single line (in this case the Tool Y axis).

In this simulation, the process of moving the robot arm to positions that meet these conditions is omitted. Since the simulation relies on synthetic parameter errors, a set of robot joint angle data that is known to be correct is generated first.

The process for determining this data set starts by choosing a single line in space which will be the nominal position of the laser line. First, a set of points along this line in space is determined. Then, using the robot's inverse kinematics, each point can then be represented by a set of joint angles. This line is generated using the "real" values of the robot's DH parameters.

The selected line for this simulation spans a large amount of the workspace. The properties of this line are given in Table 4.1.

Table 4.1: Properties of Simulation Line

Length:	4030.4 mm
Start Point:	[2500, -510,-330]
End Point:	[-62.9, 2288.9, 1027.3]
Orientation:	[0.3665, -0.3491, 0.6109] (W,P,R in rad)
Total Points:	70

The final simulated error data is generated by adding synthetic error values to all of the robot's actual DH parameters. This includes each set of joint angles for each point, as well as the nominal a, d, and α values. These synthetic error values were selected roughly using Subsection 2.2.2, in which it is mentioned that joint angle offset errors account for approximately 90% of total error and link errors for 5%. These values are given in Table 4.2

Table 4.2: Synthetic Error added to DH Parameters

Link	1	2	3	4	5	6
$\delta\theta$ (degrees)	0.2	0.35	0.8	-0.7	0.2	0.1
$\delta\alpha$ (degrees)	0.01	0.02	-0.01	0.01	-0.02	0.03
δa (mm)	0.02	0.06	-0.04	0.02	0.01	-0.01
δd (mm)	0.01	0.01	0.02	-0.04	0.01	-0.01

4.1.2 Determining Joint Angle offsets

The initial correction of joint angle offsets is made using the technique described in [3].

At each point other than the first point, the transformation matrix which represents this position can be written as in Equation 4.1

$$T = \begin{bmatrix} T_{11} & T_{12} & T_{13} & T_{14} \\ T_{21} & T_{22} & T_{23} & T_{24} \\ T_{31} & T_{32} & T_{33} & T_{34} \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{4.1}$$

At each point j other than the first point, psi values given in Equations 4.2 to 4.5. These equations are modified from those in [3] in order to measure a laser line in the Tool Y axis.

$$\psi_{j1} = (T_{12})_j - (T_{12})_1 \tag{4.2}$$

$$\psi_{j2} = (T_{22})_j - (T_{22})_1 \tag{4.3}$$

$$\psi_{j3} = \frac{(T_{12})_1}{(T_{22})_1} - \frac{(T_{14})_j - (T_{14})_1}{(T_{24})_j - (T_{24})_1} \tag{4.4}$$

$$\psi_{j4} = \frac{(T_{22})_1}{(T_{32})_1} - \frac{(T_{24})_j - (T_{24})_1}{(T_{34})_j - (T_{34})_1} \tag{4.5}$$

A function is created that solves for all of these values given joint angle offset errors. These four ψ values are essentially a measure of how much the offset "guesses" which are given to the function differ from the actual offsets of the system (in this case, these are the synthetic offsets that were specified earlier). Hence, the offset values we are looking for are the values that minimize the output of this function.

This five-parameter optimization is performed using the function "lsqnonlin.m", a part of MATLAB's optimization toolbox. The result of this is the five offset angles for joints two through six. Once the initial estimates of joint angle offset are determined, the nominal angles are modified to include these offset errors.

4.1.3 Determining Link Twist offsets

The link twist offsets can be determined using a modified form of the technique used to determine joint offsets. The modifications to the technique are as follows:

- The rotation about the laser line axis is assumed to be zero. The position data is generated in this simulation such that this is true, but the technique from [3] uses data in which this rotation is not controlled.
- Direction of movement along the line is not considered.

Table 4.3: Joint angle offset determination

	Simulated Error (deg)	Identified Error (deg)	Error(%)
$\delta\theta_2$	0.35	0.3343	-4.497
$\delta\theta_3$	0.8	0.8433	5.418
$\delta\theta_4$	-0.7	-0.7718	10.2547
$\delta\theta_5$	0.2	0.1565	-21.7709
$\delta\theta_6$	0.1	0.0894	-10.6177

By considering only orientation in the determination of these factors, dependence on errors in link length and link offset is eliminated. The offset errors in angular DH parameters can now be determined with greater accuracy. Equations 4.2 to 4.5 are replaced with Equations 4.6 to 4.8 to meet these conditions:

$$\psi_{j1} = (T_{11})_j - (T_{11})_1 \tag{4.6}$$

$$\psi_{j2} = (T_{22})_j - (T_{22})_1 \tag{4.7}$$

$$\psi_{j3} = (T_{33})_j - (T_{33})_1 \tag{4.8}$$

Optimization is performed as above using these modified equations. The result of this optimization will be an offset estimate of ten parameters: The joint twist values for joints one through five and further refinement of the joint angle offsets for joints two through six. The offset errors determined here are added to the nominal DH parameters to determine the estimate of the actual DH parameters.

4.1.4 Results of the Simulated Calibration Procedure

The result of the initial determination of joint angle offsets (Subsection 4.1.2) is listed in Table 4.3.The result of the second calibration procedure (Subsection 4.1.3) for the Link Twist offsets is listed in Table 4.4. The second calibration procedure also refines the estimates of Joint Angle offset. These refined results are listed in Table 4.5. The identified error is listed to only four decimal places, so the identified error in Table 4.4 appears identical to the simulated error due to the fact that the differences are

Table 4.4: Joint twist offset determination

	Simulated Error (deg)	Identified Error (deg)	Error(%)
$\delta\alpha_1$	0.01	0.0100	1.114×10^{-7}
$\delta\alpha_2$	0.02	0.0200	-9.364×10^{-9}
$\delta\alpha_3$	-0.01	-0.0100	-2.821×10^{-8}
$\delta\alpha_4$	0.01	0.0100	2.917×10^{-7}
$\delta\alpha_5$	-0.02	-0.0200	-1.030×10^{-7}

Table 4.5: Refined joint angle offset determination

	Simulated Error (deg)	Identified Error (deg)	Error(%)
$\delta\theta_2$	0.35	0.3343	-4.497
$\delta\theta_3$	0.8	0.8231	1.968
$\delta\theta_4$	-0.7	-0.7000	4.431×10^{-9}
$\delta\theta_5$	0.2	0.2000	-2.950×10^{-8}
$\delta\theta_6$	0.1	0.0755	-28.5481

insignificantly small.

4.2 Simulation of Data Acquisition Process

The purpose of this simulation is to mimic the process of actually obtaining data from a real robot. This includes the movement of the robot, the placement of the CMOS cameras, and the feedback they would give from the laser line. This simulation method provides several advantages: it allows for a complete simulation of the required robot movement functions (to facilitate the use of this process in a real-world experiment) and it allows us to analyze the effect of errors in movement and image acquisition on the final process. During this process, an initial "nominal" position of the robot is selected. From this position, the position of the camera planes in space are designated. Following this, each additional point in the simulated data set is calculated by moving to an estimated position (determined using the robot controller) and then performing the adjustment steps described in Subsections 4.2.2 and 4.2.3.

4.2.1 Position of Camera Plane and Obtaining Camera Feedback

In the physical environment in which the experiment is conducted, an assembly with two cameras is placed near the robot. In this simulation, the virtual location and orientation of the camera assembly is recorded in world frame coordinates. The feedback of the cameras is then determined computationally. The camera assembly is represented as two parallel camera planes, with a designated separation distance. This is representative of the actual camera plane orientation with the second camera plane mirrored, as seen in Figure 3.3. The position and orientation of the first camera plane gives a new coordinate reference frame (referred to as the "camera frame" (x,y,z). The origin of the camera frame is where the centre of the camera's sensor would be. The position of the camera plane can be arbitrary in space (although it must be positioned such that the robot's tool could point the laser cross into it). In this simulation, the camera plane is selected based on the desired robot trajectory. A nominal position of the robot is used to designate the camera plane. The nominal position is the start of the calibration process. At this nominal position, the laser line intersects with the origin of both camera planes and is normal to them. The camera frame orientation is the same as the robot's nominal orientation. The camera frame origin is also along the tool Y-axis in the nominal position. Thus, the transformation matrix from the tool frame in the nominal pose to the camera frame is a simple Y-axis translation, as given in Equation 4.9, where T is the transformation from world frame to tool frame in the nominal position and d is an arbitrary distance along the Y-axis, representing the distance from the tool frame origin to the camera plane.

$$T_{camera} = T \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & d \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{4.9}$$

Once this frame is defined, the camera feedback for any given robot position can be obtained by finding the intersection of the tool Y axis at that position with the camera frame x-z plane, and expressing this point in camera frame coordinates. This will give the distance of the laser dot from the centre of the camera plane in camera

x and z coordinates.

4.2.2 Adjusting Pose Using Translation in the Tool Frame

The result of this process will effectively "move" the simulated robot along a given Tool axis to get the camera's feedback to be as close to the nominal camera feedback as possible. Nominal camera feedback is defined as the camera reading at an initial, nominal pose. The Tool X-Z plane is approximately parallel to the camera plane. Only feedback from the first camera is used here. The current simulated robot position is a "guess" position, calculated by determining a point that would be along the Tool Y-axis according to the nominal DH parameters. The algorithm steps are:

- the laser position on Camera 1 is determined;
- a second pose is found, representing a tool movement of a certain small distance along whichever axis is currently being adjusted;
- the laser position on Camera 1 is determined for this pose;
- using linear interpolation, the point between these two poses at which the laser spot on the camera is closest to its nominal position is determined;
- this pose is recorded as the final adjusted pose.

For all further adjustment steps, this adjusted pose becomes the new "guess" pose.

4.2.3 Adjusting Pose Using Rotation in the Tool Frame

While the positional changes are measured using the feedback of one camera, the rotational differences must be measured using both cameras. In order to find a point that is closest to the nominal orientation, differences in feedback from the two cameras are used. Similar to the translation adjustment, the current "guess" pose will be modified in order to reduce the error in orientation. The following steps are used:

- the current laser position is found for both cameras;
- the difference between these two positions is calculated;
- a second pose is found, representing a small tool axis rotation about whichever axis is being adjusted;

- the laser positions are found for both cameras at this position;
- the desired point is the point at which the difference between the laser positions on both cameras is the smallest. Using the differences determined at each adjustment pose, and linear interpolation, the pose that is closest in orientation to the nominal pose is found;
- this pose is recorded as the final adjusted pose.

For all further adjustment steps, this pose becomes the new "guess" pose.

4.2.4 Full Adjustment at any Pose

The full adjustment at any pose is done here using the processes described in Subsections 4.2.2 and 4.2.3 repeatedly, in both the Tool X- and Z-axes. First, the orientation of the tool is adjusted, by performing a Tool X rotation adjustment, then Tool Z rotation adjustment, and repeatedly performing these two until the difference between camera feedback between each camera is within a specified threshold of the nominal difference.

Following this, the pose of the tool is adjusted, by performing a Tool X translation adjustment, followed by a Tool Z translation adjustment. This is repeated until the feedback of the first camera is within a certain threshold from the nominal feedback of the first camera.

If, after a certain number of attempts, no acceptable pose was reached, the pose is not recorded. However, the simulation never encountered a pose for which this was necessary. Once an acceptable set of poses is determined using this method, the calibration is performed with this data set in the same way as in Subsection 4.1.2.

4.2.5 Results of Calibration Using Data Generated with Simulation

The purpose of this simulation is to model the actual movement of the robot in space in order to prove that the adjustment pattern calibrates the robot. Only the joint angle offset errors are calculated here. The results of the joint angle offset error calculation are listed in Table 4.6.

Table 4.6: Joint angle offset determination for adjustment simulation

	Input (deg)	Calculated (deg)	Error(%)
$\delta\theta_2$	0.35	0.3486	-0.4094
$\delta\theta_3$	0.8	0.7988	-0.1548
$\delta\theta_4$	-0.7	-0.7001	0.0212
$\delta\theta_5$	0.2	0.2011	0.5265
$\delta\theta_6$	0.1	0.1012	1.2178

Chapter 5

Experiment Method Refinement

5.1 Physical Setup and Experiment

While the method described in Section 4.2 is functional in the simulated environment, some modifications were made to refine it to improve the precision for an actual physical experiment. These mainly include modifications to make image analysis more precise with the movement of the robot.

5.1.1 Adjustment Process

The experimental adjustment process is similar to the method described in Subsection 4.2, which was applied to the simulation. The difference is the order in which each adjustment is performed.

The order of adjustment in the experiment is:

- rotation adjustment about X axis (Subsection 4.2.3);
- rotation adjustment about Y axis;
- translation adjustment along Z axis (Subsection 4.2.2);
- rotation adjustment about Z axis (Subsection 4.2.3);
- translation adjustment along X axis (Subsection 4.2.2);
- if measurements from cameras are within a specified threshold of 3 or 5 pixels (Subsection 5.1.2), accept the pose as correct. Otherwise, repeat this process.

This change in order was made to make sure the laser was still reaching the cameras in approximately the centre of the camera sensor, as the image analysis is not capable of interpreting images in which the cross is near the edge of the image. The order of adjustments in the simulation (fully adjusting orientation before attempting any adjustment of position) was likely to move the cross to an un-readable edge of the image. In practice, by putting the Z-axis translation adjustment immediately after the X-axis rotation adjustment, these would approximately counteract each other and return the cross to the centre of the image. The same is true of Z-axis rotation with X-axis translation.

5.1.2 Choosing Adjustment Threshold by Analyzing Repeatability

The requirements for choosing the threshold at which to consider a pose acceptable were chosen to be as small as possible while remaining feasible in the context of the experiment. In order to find the smallest feasible range of acceptable poses, an experiment was performed that analyses the overall repeatability of the robot, camera, and image processing. This experiment was performed simply by taking an image at a certain robot pose, moving the robot away from this pose and then back, and then taking another image. This was performed 10 times to generate an estimated point spread.

The results of this analysis are shown for Camera 1 in Figure 5.1 and for Camera 2 in Figure 5.2. In these figures, each circle represents a determined cross centre location. The average of all images is represented with an cross symbol. A circle is used to represent a distance of one half pixel from the average location.

It can be seen in Figures 5.1 and Figure 5.2 that Camera 1 (directly in line with the laser) has a substantially greater repeatability than Camera 2 (viewing the reflected light from the beam splitter). From this, the most restrictive threshold would seem to be 1 to 2 pixels when remaining at the same location. However, some inherent additional imaging inaccuracies do occur when moving back and forth along the line. Two different threshold levels were selected based on this: a high precision method using a 3 pixel threshold on each camera, and an average method using a 5 pixel threshold on each camera. In order for the high precision method to be successful,

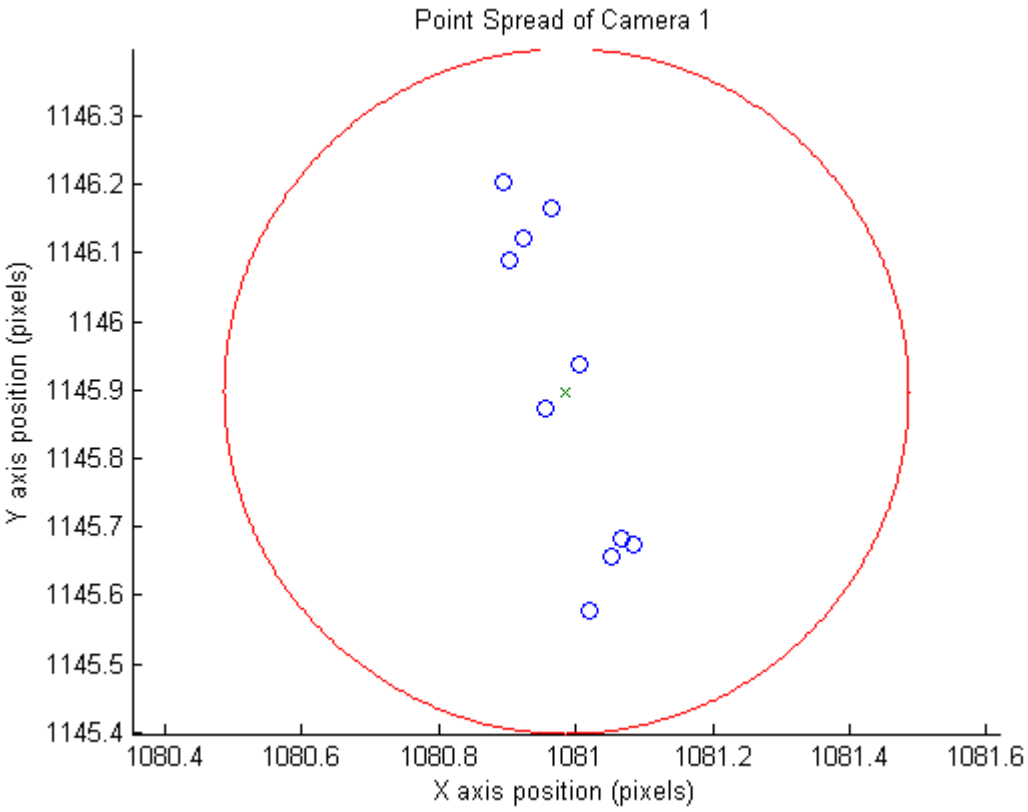

Figure 5.1: Point spread on Camera 1 of repeatability test

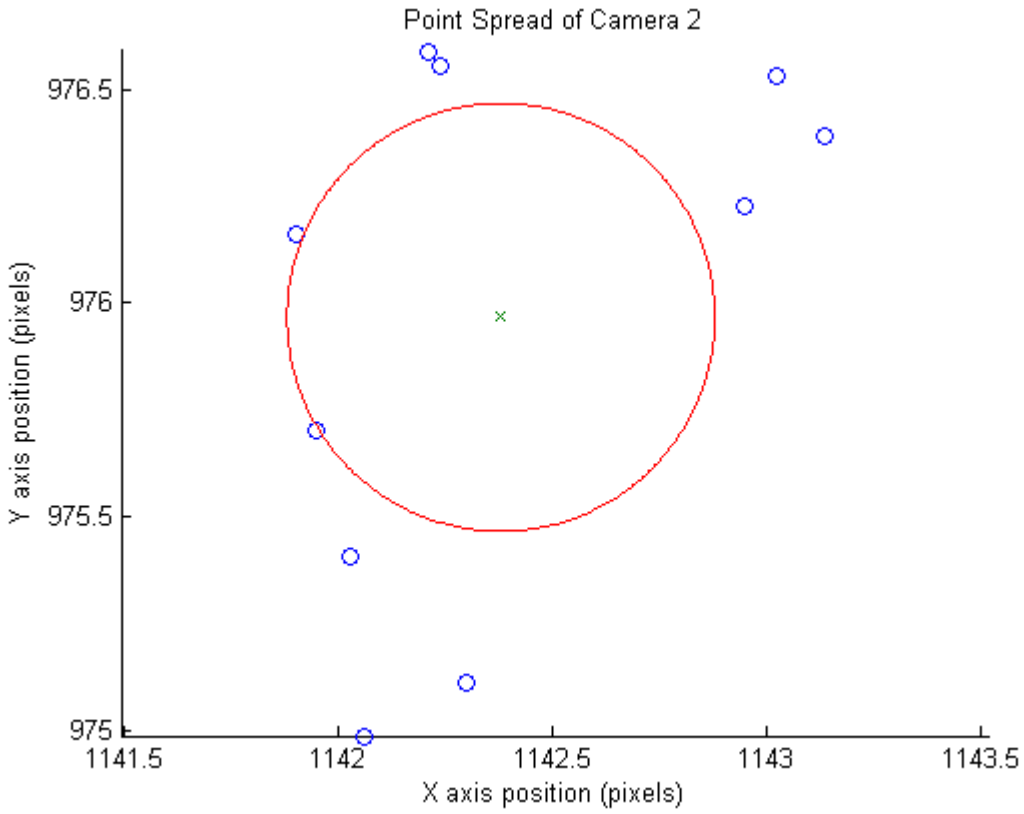

Figure 5.2: Point spread on Camera 2 of repeatability test

the overall time taken to perform a calibration increases (see Subsection 5.1.4), and an environment free from large magnitude vibrations is required.

5.1.3 Backlash Adjustment

Backlash, mentioned in Subsection 2.2.4, is difficult to model. In this case, no actual modeling is done to predict the effect of backlash on the robot pose. Instead, the movement pattern of the robot is modified such that every pose is approached from the same rotational direction in each joint. The necessity of this modification was determined with a brief analysis, which showed that the direction of approach had a noticeable effect on the cross location in images, far more than could be explained by camera and image analysis error. When approaching a certain pose from a different direction, the cross centre location was up to 30 pixels away from the initial location. This is six times the threshold used to determine even a rough equivalence.

Each movement instruction sent to the robot was replaced with a three-step process. Starting from the desired pose in world frame coordinates, the following was performed.

1. A position was identified that was equivalent to a movement of a small amount (one quarter of a degree) in each joint, from the target pose. This is the "back" position from which the target pose will be approached.

2. The robot is instructed to move to this back position.

3. The robot is instructed to move to the target pose.

Using this method, each joint will always move in the same sense when approaching target poses.

5.1.4 Calibration Time

When all steps are considered, the calibration process becomes quite lengthy. Many factors contribute to this. The backlash adjustment in Subsection 5.1.3 modifies each movement to be two full movements. To enhance precision, the images were only taken a full second after the robot finished a motion. Because of the nature of the adjustments, for each individual adjustment, several movements and images were required, and several adjustments were required for each data point. Depending

on the number of adjustments required before the imaging threshold was reached, a single data point could sometimes take up to 20 minutes.

In order to improve the calibration time, some modifications were made. The image threshold was set to be larger, as mentioned in Subsection 5.1.2. The total number of data points per set was between 10 and 15, unlike in the simulations in which 30 or more data points could be used. These resulted in a smaller data set, and lower precision in the pose measurement.

5.1.5 Laser Offset Parameters

The calibration routine relies on the laser line being oriented perfectly in the tool Y-axis. In practice, it is impossible to place the laser exactly along this line. In order to compensate for this, four additional unknown parameters are introduced that represent the differences in position and orientation between the actual laser axis and the tool Y-axis. The five-parameter optimization shown in Subsection 4.1.2 a nine-parameter optimization. The values determined for laser offset are shown in Subsections 6.3.1 and 6.3.2.

Chapter 6

Results

6.1 Simulation Results

This section gives the results of a full simulated calibration to find the joint angle and link twist offsets. This is a two-stage process. The results of the initial determination of joint angle offsets are given in Table 6.1. The results of the modified determination of joint angle offsets are given in Table 6.2. The results of the determination of the link twist offsets are given in Table 6.3.

Table 6.1: Result of the initial determination of joint angle offsets

Joint	Identified angle (degrees)	Simulated angle (degrees)	Percent error
2	0.33	0.35	-4.50
3	0.84	0.80	5.42
4	-0.77	-0.70	10.25
5	0.16	0.20	-21.77
6	0.09	0.10	-10.62

6.2 Discussion of Simulation Results

The simulation process correctly determined joint angle offsets for joints 4 and 5 with a high degree of precision. For these two joints, the determined values were

Table 6.2: Result of the modified determination of joint angle offsets

Joint	Identified angle (degrees)	Simulated angle (degrees)	Percent error
2	0.33	0.35	-4.50
3	0.82	0.80	1.97
4	-0.70	-0.70	4.43×10^{-9}
5	0.20	0.20	-2.95×10^{-8}
6	0.07	0.10	-28.54

Table 6.3: Result of the determination of link twist offsets

Joint	Identified angle (degrees)	Simulated angle (degrees)	Percent error
2	0.01	0.01	1.14×10^{-7}
3	0.02	0.02	-9.36×10^{-9}
4	-0.01	-0.01	-2.82×10^{-8}
5	0.01	0.01	2.92×10^{-7}
6	-0.02	-0.02	-1.03×10^{-7}

within a very small fraction of a degree (10^{-11}). The remaining joint angle offsets were determined with reasonable precision. The least accurate was joint 6, with 25% error, representing a difference of 0.03 degrees. The values of link twist offsets were determined with far greater precision in the simulation. This is likely due to the substantially lower values of these offsets. While simulated joint angle offsets ranged from 0.1 degrees to 0.8 degrees, the link twist offsets were all 0.02 degrees or smaller.

6.3 Experimental Results

This section gives the results of each experiment using both robots. All processed data sets are included, though not all data sets are considered in the analysis. The nature of the experiment is such that the calibration may occasionally fail completely. Vibration disturbance during image acquisition at any point in the calibration process may cause small errors, but if this vibration disturbance occurs during the acquisition of the nominal "target" image, the entire data set will be affected. These data sets are considered outliers and are excluded from the analysis. For each robot, outlying

data sets will be identified.

6.3.1 Experiment Using FANUC S-420iF Robot Arm

A total of five complete data sets were collected with this robot. The results of these data sets are given in Tables 6.4 and 6.5.

Table 6.4: Joint angle offset result of calibration process using FANUC s-420iF robot

	Joint angle offset (degrees)				
Set	Joint 2	Joint 3	Joint 4	Joint 5	Joint 6
1	0.13	0.09	0.20	-0.01	45.41
2	0.07	0.08	0.07	0.03	79.14
3	6.69	9.04	23.48	10.76	158.13
4	0.25	0.11	-0.03	0.16	101.97
5	0.18	0.12	0.18	-0.01	45.41

Table 6.5: Laser offset result of calibration process using FANUC s-420iF robot

Laser offset (degrees and mm)			
X angle	Y angle	X offset	Y offset
0.01	44.87	0.00	0.00
2.70	78.32	1.24	-0.38
31.02	96.73	0.94	-0.09
168.96	101.13	-1.67	-1.88
0.01	44.87	0.00	0.00

Of the identified joint angles, joints 2 and 3 were identified with a lower variation in results. These two joints angles were analyzed to determine which data sets are excluded as outliers. Of these five sets, Data Set 3 was identified as an outlier. This is because the values identified for Joints 2 and 3 were over 3 standard deviations from the mean value given in Table **??**. Excluding this outlier, the mean value of all data sets is presented in Table 6.6. Additionally, the standard deviation is

presented to demonstrate how consistent the measurements are. While joints 2 and 3 are identified with consisten results, in joints 4,5 and 6 the standard deviation is larger than the identified joint offset. The values obtained for these joint angles are much less reliably identified. The standard deviations for the laser offset parameters are also high, although this is to be expected, as the tool holding the laser was frequently adjusted between gathering each data set.

Table 6.6: Average of Results of Calibration of FANUC s-420iF Robot

		Average	Standard deviation
Joint Angle Offset (degrees)	Joint 2	0.16	0.07
	Joint 3	0.10	0.02
	Joint 4	0.10	0.11
	Joint 5	0.04	0.08
	Joint 6	67.98	27.68
Laser Offset	X angle	42.92	84.04
(degrees and mm)	Y angle	67.30	27.52
	X offset	-0.11	1.19
	Y offset	-0.56	0.89

6.3.2 Experiment Using Motoman MH-180 Robot

A total of eleven complete data sets were collected with this robot. The results of these data sets are given in Tables 6.7 and 6.8

Of these eleven sets, Data Sets 2, 5, 10, and 11 were identified as outliers. In this case, the values of joints 2 and 4 were considered to make this judgement. Excluding these outliers, the mean value of all data sets is presented in Table 6.9. Additionally, the standard deviation is presented to demonstrate how consistent the measurements are.

Table 6.7: Joint angle result of Calibration Process Using Motoman MH-180 Robot

	Joint angle offset (degrees)				
Set	Joint 2	Joint 3	Joint 4	Joint 5	Joint 6
1	-0.03	-0.31	0.15	-0.39	63.93
2	-3.87	-1.10	-0.55	-0.14	88.69
3	-0.02	-0.34	0.17	-0.42	-32.55
4	-0.02	-0.29	0.15	-0.37	52.64
5	-2.56	-3.26	-5.25	-1.86	-11.26
6	-0.05	-0.33	0.18	-0.45	-47.22
7	-0.02	-0.29	0.15	-0.37	52.64
8	-0.03	-0.32	0.18	-0.41	64.01
9	0.08	1.28	0.18	-1.25	-90.36
10	-2.28	0.48	0.50	-0.80	62.92
11	-4.81	12.95	5.22	11.85	268.95

Table 6.8: Laser offset result of Calibration Process Using Motoman MH-180 Robot

Laser offset (degrees and mm)			
X angle	Y angle	X offset	Y offset
-0.56	64.15	-0.01	-0.21
3.06	88.64	-0.08	-0.26
-0.20	-32.28	-0.04	-0.19
-0.39	52.84	-0.01	-0.02
1.30	-10.90	0.00	0.00
-0.20	-46.84	-0.72	-0.72
-0.39	52.84	-0.01	-0.02
-0.61	64.25	0.00	-0.01
0.01	-90.16	0.00	0.00
0.29	62.51	-0.13	0.21
0.07	4.89	0.16	-0.24

Table 6.9: Average of Results of Calibration of Motoman MH-180 Robot

		Average	Standard Deviation
Joint Angle Offset (degrees)	Joint 2	-0.01	0.04
	Joint 3	-0.09	0.60
	Joint 4	0.16	0.02
	Joint 5	-0.52	0.32
	Joint 6	9.01	64.05
Laser Offset	X angle	-0.34	0.22
(degrees and mm)	Y angle	9.26	64.02
	X offset	-0.11	0.27
	Y offset	-0.17	0.26

6.3.3 Discussion of Experimental Results

The precision of the calibration routine is determined here using consistency, because the actual joint offsets are not known. Multiple calibration routines are performed and the results are compared with each other. The standard deviation gives us a measure of the consistency of results.

The goal of the calibration procedure is to determine the value of several calibration parameters which remain constant throughout the process. Ideally, each calibration procedure would return the exact same result. However, in practice many different factors can affect the calibration procedure. The standard deviation is included with results to measure the degree to which the results remain consistent. A more consistent result will have a lower standard deviation. A consistent result indicates that the calibration procedure is able to successfully identify the parameter in question.

Some data sets obtained experimentally were deemed to be outliers. Certain factors can affect the precision of the data gathering method to the point that the analysis no longer produces reasonable results. These data sets are easily identifiable in the results because any small problem in data gathering tends to produce large errors in the analysis. Outliers were identified by examining the joint angles which had the most consistent results. The data gathered using the FANUC s-420iF gave reasonable results for some joint offsets, but not all. Consistent results were produced

for joint 4. Joints 2, 3, and 5 have reasonable results but the consistency is not high. The results for joint 6 were not at all consistent or reasonable, but this may be due to the laser offset process in calibration, discussed further in Subsection 6.3.4. The results of the laser offset process were not consistent at all. However, this is to be expected because of changes in the laser's placement between gathering each data set. The data gathered using the Motoman MH-180 robot shows a similar lack of consistency in joint 6 offsets, likely for the reason discussed in Subsection 6.3.4. The results were slightly less consistent than those from the FANUC s-420iF robot, but were larger overall. A larger offset is more difficult to correctly identify. The comparison may not be as apt here as a substantially larger number of data sets were obtained using the Motoman MH-180.

6.3.4 Relationship Between Joint 6 Angle Offset and Laser Y Angle Offset

The joint 6 angle offset results were not consistent at all when compared with each other. However, the exact same inconsistency was seen in the results for the laser's Y angle offset. The difference between these two remained very consistent. An analysis of the difference between the Laser Y offset and Joint 6 is given for each robot, in Tables 6.10 and Table 6.11. The relatively high consistency of this difference implies that these two values are related. In fact, a each of these movements (Laser Y angle rotation and Joint 6 rotation) would move the laser in the same direction. Being able to accurately place the laser would aid in determining the actual Joint 6 offset.

Table 6.10: Difference between Joint 6 Angle Offset and Laser Y Angle Offset For FANUC s-420iF

Data Set	Difference
1	0.54
2	0.82
3 (outlier)	61.40
4	0.84
5	0.54
Mean	0.69
Standard Deviation	0.17

Table 6.11: Difference between Joint 6 Angle Offset and Laser Y Angle Offset for Motoman MH-180

Data Set	Difference
1	-0.22
2 (outlier)	0.05
3	-0.27
4	-0.20
5 (outlier)	-0.36
6	-0.37
7	-0.20
8	-0.23
9	-0.21
10 (outlier)	0.41
11 (outlier)	264.06
Mean	-0.24
Standard Deviation	0.06

Chapter 7

Concluding Remarks

7.1 Summary and Conclusions

Chapter 1 provided an introduction to the use of robots in and the necessity of calibration. The motivation for this thesis was discussed: a novel method for calibration using simple, low-cost sensors. The main objectives of this thesis were listed and described. A statement of originality was included to identify the main original work described in this thesis.

Chapter 2 described the theoretical background needed for the calibration routine deveopment. The mathematical representation of general robot motion was discussed, along with DH parameter conventions used in this thesis. A representative collection of calibration methods were summarized and discussed, including the methods from which the work in this thesis drew inspiration.

Chapter 3 described the experimental method. The physical experimental tools were discussed, including the cameras, robots, and the laser. The camera assembly was described, along with the mathematical interpretation of the camera images. The method of image analysis was fully disclosed.

Chapter 4 detailed the methods used for the simulation. The mathematical method used to perform the actual calibration once data is obtained was outlined. Two simulations were reported: a simple simulation was used to establish feasability wherein data sets were fully generated without considering robot motion, and a more realistic simulation in which the process of robot motion and image results were all considered.

Chapter 5 described modifications to Chapter 4's method that were applied to the experimental data collection. These included the sequence of movements used

to collect a data point, an analysis of the repeatability of the imaging capture and analysis, modifications to avoid joint backlash, and a discussion of methods used to reduce the total time required for the calibration.

Chapter 6 provided the results of each calibration process. The simulation was able to identify 5 joint angle offsets and 5 link twist offsets. The experimental results identified 2 joint angle offsets with reasonable confidence and 2 joint angle offsets with lower confidence.

In this thesis, a method for calibration of six axis serial wrist partitioned robots was discussed, simulated, and implemented experimentally. Calibration was performed by constraining the end-effector motion to a single line in space and retaining constant end-effector orientation. This allows the calibration routine to be performed with simple and low cost sensing equipment. The linear constraint was enforced using a laser diode mounted to the robot end-effector, with an optical attachment producing a cross shape. This created a "virtual closed kinematic chain," with the laser line considered an additional link to ground, constraining the robot motion.

The simulated calibration was able to identify 10 of the 12 angular DH parameters (joing angle offsets and link twist errors). The experimental calibration routine was able to identify 5 of the 6 joint angle offsets, though only 2 were identified with a measure of statistical confidence. The reasonable precision of the simulation implies that this method is viable. The experimental precision was significantly less, and as such the link twist offsets were not able to be identified. This is because the identification of the link twist errors depends on the precision of the identified joint angle offsets. As joint angle offsets are much larger than link twist errors, the influence of the joint angle offsets that were not precisely identified will affect the link twist error identification and any results will simply be a reflection of this remaining joint angle error.

7.2 Recommendations

The experimental procedure should be redesigned to improve precision and confidence level, which might then lead to experimental identification of the link twist offsets. In order to increase the experimental precision and confidence level:

1. The experiment should be performed in an environment free from machine vibration. This experiment was performed on a factory floor during operating

hours. While every effort was made to time the experiment to avoid disturbing vibrations, this was not fully possible. In a calm environment the imaging would be substantially more accurate. The major source of vibration during this experiment was a sheet metal punch press frequently running, only a few metres from the experiment. Additionally, a press brake and a welding robot were both operating very close by.

2. Environmental factors affecting the optical equipment, such as dust and ambient light, could be reduced. The experiment operated in an incredibly dusty environment. The cameras and beam splitter were contained in a metal enclosure to keep out some dust and ambient light, but the window leading into this enclosure had to be frequently cleaned and a layer of dust can have substantial negative effects on camera sensing.

3. The process could be modified to run autonomously, and therefore run overnight without oversight. This would allow a very long calibration procedure to run.

4. A different imaging method or sensor might provide for increased accuracy. In this experiment, cameras were chosen in order to view a laser cross rather than a laser point. However, these cameras provide decreased accuracy when compared with the PSDs used in other experiments reported in the literature.

Additionally, a method used to estimate errors in the remaining DH paramenter offsets: link length (a) and link offset (d) was attempted but was not included in this thesis, as simulated results were not obtained successfully. This method is described in Appendix **??**. If the precision of the measurement acquisition and analysis were substantially increased, this method to obtain the remaining DH parameter offsets might be viable.

List of References

[1] J. Ruurda, T. J. van Vroonhoven, and I. Broeders, "Robot-assisted surgical systems: a new era in laparoscopic surgery.," *Annals of the Royal College of Surgeons of England*, vol. 84, no. 4, p. 223, 2002.

[2] Y. S. Kwoh, J. Hou, E. Jonckheere, S. Hayati, *et al.*, "A robot with improved absolute positioning accuracy for ct guided stereotactic brain surgery," *Biomedical Engineering, IEEE Transactions on*, vol. 35, no. 2, pp. 153–160, 1988.

[3] B. Du, N. Xi, and E. Nieves, "Industrial robot calibration using a virtual linear constraint," *International Journal on Smart Sensing and Intelligent Systems*, vol. 5, no. 4, pp. 987–1000, 2012.

[4] J. P. Merlet, *Parallel Robots*. Dordrecht ; Boston, MA : Kluwer Academic Publishers, 2006.

[5] D. Stewart, "A platform with six degrees of freedom," in *Proc. Inst. Mech. Eng. London*, 1965.

[6] S. Briot and I. Bonev, "Are parallel robots more accurate than serial robots?," *CSME Transactions*, vol. 31, no. 4, pp. 445–456, 2007.

[7] F. Robotics, "S-420i," 1998.

[8] Y. M. Robotics, "Mh180 mh225," 2014.

[9] T. K. Tanev, "Kinematics of a hybrid (parallel–serial) robot manipulator," *Mechanism and Machine Theory*, vol. 35, no. 9, pp. 1183–1196, 2000.

[10] L. Romdhane, "Design and analysis of a hybrid serial-parallel manipulator," *Mechanism and Machine Theory*, vol. 34, no. 7, pp. 1037–1055, 1999.

[11] J. J. Craig, *Introduction to Robotics*. Pearson ; Upper Saddle River, NJ, 2005.

[12] R. Beardmore, "Simple linkages." http://www.roymech.co.uk/Useful Tables/Mechanics/Linkages.html, 2011. [Online; accessed 20-November-2015].

[13] J. Denavit and R. S. Hartenberg, "A kinematic notation for lower-pair mechanisms based on matrices," *Trans. ASME E, Journal of Applied Mechanics*, vol. 22, pp. 215–221, June 1955.

[14] H. Lipkin, "A note on denavit-hartenberg notation in robotics," in *Proceedings of IDETC/CIE*, 2005.

[15] D. Kohli and M. Osvatic, "Inverse kinematics of general 6r and 5r, p serial manipulators," *Journal of Mechanical Design*, vol. 115, no. 4, pp. 922–931, 1993.

[16] D. Manocha and J. F. Canny, "Efficient inverse kinematics for general 6r manipulators," *Robotics and Automation, IEEE Transactions on*, vol. 10, no. 5, pp. 648–657, 1994.

[17] D. L. Pieper, "The kinematics of manipulators under computer control," tech. rep., DTIC Document, 1968.

[18] Z. Roth, B. Mooring, and B. Ravani, "An overview of robot calibration," *IEEE Journal on Robotics and Automation*, vol. 5, no. 3, pp. 377–385, 1987.

[19] R. P. Judd and A. B. Knasinski, "A technique to calibrate industrial robots with experimental verification," *Robotics and Automation, IEEE Transactions on*, vol. 6, no. 1, pp. 20–30, 1990.

[20] P. Shiakolas, K. Conrad, and T. Yih, "On the accuracy, repeatability, and degree of influence of kinematics parameters for industrial robots," *International journal of modelling and simulation*, vol. 22, no. 4, pp. 245–254, 2002.

[21] C. Gong, J. Yuan, and J. Ni, "Nongeometric error identification and compensation for robotic system by inverse calibration," *International Journal of Machine Tools and Manufacture*, vol. 40, no. 14, pp. 2119–2137, 2000.

[22] A. Zalucky and D. Hardt, "Active control of robot structure deflections," *Journal of dynamic systems, measurement, and control*, vol. 106, no. 1, pp. 63–69, 1984.

[23] M. Ruderman, F. Hoffmann, and T. Bertram, "Modeling and identification of elastic robot joints with hysteresis and backlash," *Industrial Electronics, IEEE Transactions on*, vol. 56, no. 10, pp. 3840–3847, 2009.

[24] S. Takata, A. Yamada, T. Kohda, and H. Asama, "Life cycle simulation applied to a robot manipulator-an example of aging simulation of manufacturing facilities," *CIRP Annals-Manufacturing Technology*, vol. 47, no. 1, pp. 397–400, 1998.

[25] A. Elatta, L. P. Gen, F. L. Zhi, Y. Daoyuan, and L. Fei, "An overview of robot calibration," *Information Technology Journal*, vol. 3, no. 1, pp. 74–78, 2004.

[26] FANUC, "FANUC robot ARC mate 100i model B and M-6i model B maintenance manual,"

[27] M. R. Driels, L. W. Swayze, and L. S. Potter, "Full-pose calibration of a robot manipulator using a coordinate-measuring machine," *The International Journal of Advanced Manufacturing Technology*, vol. 8, no. 1, pp. 34–41, 1993.

[28] H. Zhuang, K. Wang, and Z. S. Roth, "Simultaneous calibration of a robot and a hand-mounted camera," *Robotics and Automation, IEEE Transactions on*, vol. 11, no. 5, pp. 649–660, 1995.

[29] C.-C. D. Lu and M. J. D. Hayes, "Kinematic calibration of 6r serial manipulators using relative measurements," in *CCToMM Mechanisms, Machines, and Mechatronics (M3) Symposium*, 2013.

[30] H. W. Stone and A. C. Sanderson, "A prototype arm signature identification system," in *Robotics and Automation. Proceedings. 1987 IEEE International Conference on*, vol. 4, pp. 175–182, IEEE, 1987.

[31] N. Simpson and M. Hayes, "Simulation of a kinematic calibration procedure that employs the relative measurement concept," in *Proc. CSME Forum*, 2004.

[32] K. English, M. J. D. Hayes, M. Leitner, and C. Sallinger, "Kinematic calibration of six-axis robots," in *Proceedings of the CSME Forum, Kingston (Canada)*, 2002.

[33] I.-C. Ha, "Kinematic parameter calibration method for industrial robot manipulator using the relative position," *Journal of mechanical science and technology*, vol. 22, no. 6, pp. 1084–1090, 2008.

[34] M. Hayes and P. OLEARY, "Kinematic calibration procedure for serial robots with six revolute axes," 2001.

[35] W. H. Press, S. A. Teukolsky, W. T. Vetterling, and B. P. Flannery, *Numerical recipes in C*, vol. 2. Citeseer, 1996.

[36] F. Deumlich, *Surveying Instruments*. W. de Gruyter, 1982.

[37] M. R. Driels and U. S. Pathre, "Robot calibration using an automatic theodolite," *The International Journal of Advanced Manufacturing Technology*, vol. 9, no. 2, pp. 114–125, 1994.

[38] M. R. Driels and U. S. Pathre, "Vision-based automatic theodolite for robot calibration," *Robotics and Automation, IEEE Transactions on*, vol. 7, no. 3, pp. 351–360, 1991.

[39] J. F. Jarvis, "Microsurveying: towards robot accuracy," in *Robotics and Automation. Proceedings. 1987 IEEE International Conference on*, vol. 4, pp. 1660–1665, IEEE, 1987.

[40] K. Lau, R. Hocken, and L. Haynes, "Robot performance measurements using automatic laser tracking techniques," *Robotics and computer-integrated manufacturing*, vol. 2, no. 3, pp. 227–236, 1985.

[41] D. J. Bennett and J. M. Hollerbach, "Autonomous calibration of single-loop closed kinematic chains formed by manipulators with passive endpoint constraints," *Robotics and Automation, IEEE Transactions on*, vol. 7, no. 5, pp. 597–606, 1991.

[42] A. Omodei, G. Legnani, and R. Adamini, "Calibration of a measuring robot: Experimental results on a 5 dof structure," *Journal of Robotic Systems*, vol. 18, no. 5, pp. 237–250, 2001.

[43] L. P. Foulloy and R. B. Kelley, "Improving the precision of a robot," in *Robotics and Automation. Proceedings. 1984 IEEE International Conference on*, vol. 1, pp. 62–67, IEEE, 1984.

[44] A. Nubiola and I. A. Bonev, "Absolute robot calibration with a single telescoping ballbar," *Precision Engineering*, vol. 38, no. 3, pp. 472–480, 2014.

[45] M. Ikits and J. M. Hollerbach, "Kinematic calibration using a plane constraint," in *Robotics and Automation, 1997. Proceedings., 1997 IEEE International Conference on*, vol. 4, pp. 3191–3196, IEEE, 1997.

[46] H. Zhuang, S. H. Motaghedi, and Z. S. Roth, "Robot calibration with planar constraints," in *Robotics and Automation, 1999. Proceedings. 1999 IEEE International Conference on*, vol. 1, pp. 805–810, IEEE, 1999.

[47] M. A. Meggiolaro, G. Scriffignano, and S. Dubowsky, "Manipulator calibration using a single endpoint contact constraint," in *Proceedings of ASME Design Engineering Technical Conference, Baltimore, USA*, 2000.

[48] V. Hayward, "Calibration of a parallel robot using multiple kinematic closed loops,"

[49] C. S. Gatla, R. Lumia, J. Wood, and G. Starr, "An automated method to calibrate industrial robots using a virtual closed kinematic chain," *Robotics, IEEE Transactions on*, vol. 23, no. 6, pp. 1105–1116, 2007.

[50] C. Wang, W. Chen, and M. Tomizuka, "Robot end-effector sensing with position sensitive detector and inertial sensors," in *Robotics and Automation (ICRA), 2012 IEEE International Conference on*, pp. 5252–5257, IEEE, 2012.

[51] Edmund Optics, *EO-5012 Monochrome USB 3.0 Camera*, 2014.

[52] D. Litwiller, "Ccd vs. cmos," *Photonics Spectra*, vol. 35, no. 1, pp. 154–158, 2001.

[53] H. P. VC, "Method and means for recognizing complex patterns," Dec. 18 1962. US Patent 3,069,654.

[54] J. Illingworth and J. Kittler, "A survey of the hough transform," *Computer vision, graphics, and image processing*, vol. 44, no. 1, pp. 87–116, 1988.

[55] R. O. Duda and P. E. Hart, "Use of the hough transformation to detect lines and curves in pictures," *Communications of the ACM*, vol. 15, no. 1, pp. 11–15, 1972.

[56] D. H. Ballard, "Generalizing the hough transform to detect arbitrary shapes," *Pattern recognition*, vol. 13, no. 2, pp. 111–122, 1981.

[57] R. Ofner, P. OLeary, and M. Leitner, "A collection of algorithms for the determination of construction points in the measurement of 3d geometries via light-sectioning," in *2nd Workshop on European Scientific and Industrial Collaboration Promoting: Advanced Technologies in Manufacturing*, pp. 505–512, 1999.