REPLACEMENT OF HUMAN OPERATOR WITH VISION-GUIDED ROBOT FOR ELECTRONIC COMPONENT PICK-AND-PLACE WORK CELL

by

David George Vutetakis Jr

A thesis submitted to the faculty of
The University of North Carolina at Charlotte
in partial fulfillment of the requirements
for the degree of Master of Science in
Applied Energy and Electromechanical Systems

Charlotte

2016

Approved by:

_____
Dr. Aidan F. Browne

_____
Dr. Min C. Shin

_____
Dr. Maciej Noras

ABSTRACT

DAVID GEORGE VUTETAKIS JR. Replacement of human operator with vision-guided robot for electronic component pick-and-place work cell. (Under the direction of DR. AIDAN F. BROWNE)

Picking and placing parts is a fundamental operation involved in circuit board manufacturing and assembly plants. Pin thru-hole (PTH) components with non-standard features are incompatible with most existing high speed pick-and-place systems; this results in the use of manual operators to perform the circuit board assembly process. The inclusion of a human element in an otherwise automated manufacturing plant introduces a variety of unfavorable effects including decreased throughput rates and increased errors. In this research, I have developed an initial framework which allows for automated circuit board assembly of non-standard electrical component types. An articulated robotic manipulator with 6 degrees of freedom (DOF) was guided by precisely calibrated machine vision cameras to obtain workspace information. A model based object recognition approach was used for the identification and subsequent localization of components and their placements, which demonstrated robust and repeatable results among the different components tested. The experimental data demonstrates a decrease over the current average human throughput rate as well as reduced error frequency, with future work identified to further optimize the performance results.

DEDICATION

I would like to dedicate this thesis to my father who has inspired me from a young age. His support and guidance has been a cornerstone of my successes thus far, and I am proud to follow in his footsteps.

ACKNOWLEDGEMENTS

I would like to acknowledge my advisor, Dr. Aidan F. Browne, for his assistance throughout my research. Dr. Browne is also primarily responsible for providing the equipment, parts and materials I needed to complete my work. I would also like to acknowledge Dr. Min Shin for his support and guidance, as well as Dr. Sherlock for his valued input and suggestions. I appreciate the assistance from one of my fellow graduate students, Andrew Paustian, for his time and assistance with CAD modelling and 3D printing. Final gratitude is also given to my father for his tireless backing and financial support, as well as the graduate school of UNC Charlotte for their additional financial support. Most importantly, I would like to thank God for his endless support and guidance.

TABLE OF CONTENTS

CHAPTER 1: INTRODUCTION

1.1 Background

The circuit board manufacturing process can become quite complex due to the extremely wide variety of printed circuit board designs and the electronic components with which they are populated. There are many different categories for these parts, but their mounting on a circuit board can be classified into two general categories: surface mount technology (SMT) or pin thru-hole technology (PTH). Figure 1 displays multiple surface mount components on a circuit board, two of which are highlighted as typical SMT components.

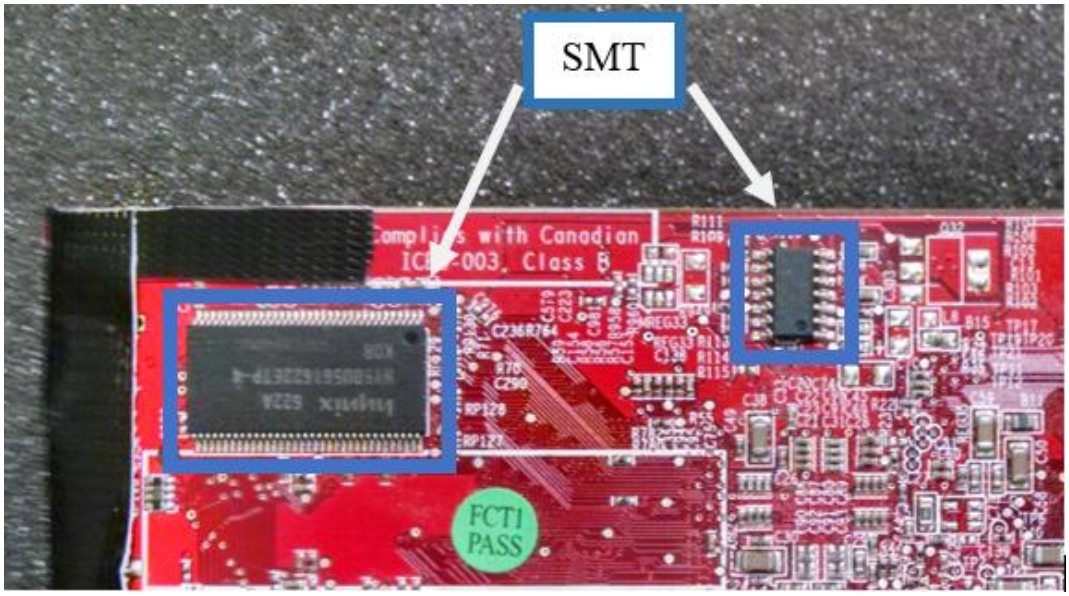

Figure 1: Example SMT components

SMT parts do not use pin leads, but instead use metallic tabs which are easily soldered to circuit board pads after placement. The tabs do not frequently require adjustment or other conditioning which improves the assembly throughput rate. SMT parts can also be manufactured smaller than the thru-hole counterparts and at reduced cost. Smaller sizes allow higher component density and increased utilization of board surface area. For these reasons, SMT is a preferred circuit board placement methodology in most industries.

The alternative to SMT is the use of pin thru-holes, which was the most common approach until the early 1990's when it was surpassed by SMT. PTH parts have protruding pin leads that are inserted in lead holes on a circuit board, which provides improved physical mounting strength. Components that may be exposed to high mechanical stress must be mounted using thru-holes for additional strength, and some components may simply be unavailable as an SMT. Axial leads protrude from opposite ends along a single axis of a component as opposed to radial leads which are projected in one direction from the same surface (Figure 2). The leads of radial components are generally arranged into rows and are referred to as either single in-line package (SIP) or dual in-line package (DIP) as shown in Figure 3, but may occasionally take other forms as well.

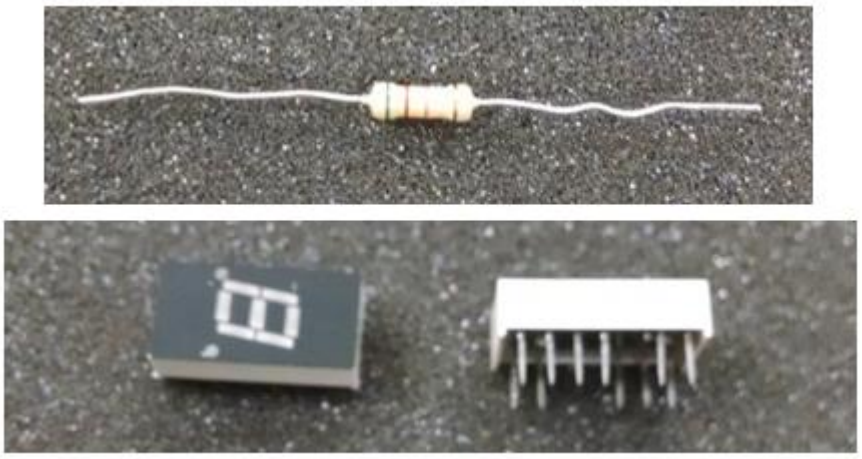

Figure 2: Radial component (top); axial component (bottom)

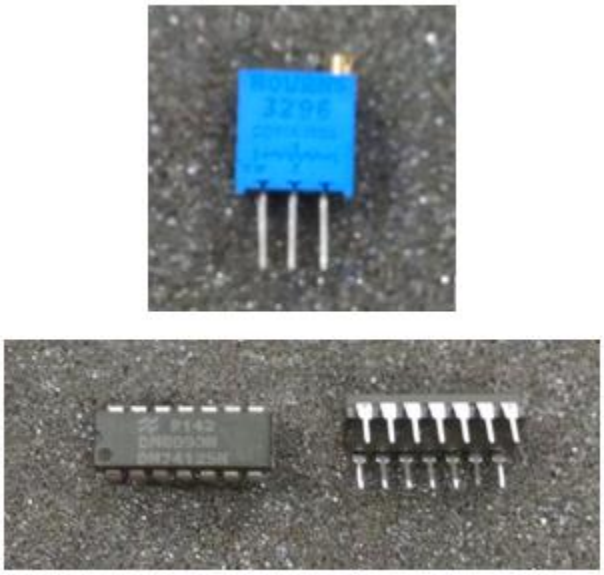

Figure 3: Single in-line package (top); dual in-line package (bottom)

Most PCB manufacturing plants are typically not involved in the layout of the board or the design of the components and must therefore have many diverse automation

systems to handle the diverse task requirements. The components needed for a particular board assembly may arrive in several packaging forms, such as reels, trays, tubes or bulk.

Reels are common for the majority of SMT parts as they can store large component volumes which reduces delays involved with initially feeding the reels into the assembly machinery. PTH components also frequently implement reel packaging, but sometimes may also use trays or tubes. This packaging method allows additional protection of components and may also be necessary if a component cannot be practically packaged using reels due to large dimensions. Occasionally components that are purchased in bulk may be packaged freely with no useful containment characteristics. These parts are typically transferred to trays or bins to be used during the assembly process, but are sometimes taped by the PCB manufacturer.

Thru-hole components sometimes require various preparation and conditioning depending on the type of original component packaging. Lead forming may be required to ensure all pins have the correct orientation and spacing. Lead trimming is often performed to remove excess length, which may occur before placement or after soldering. Clinching is also commonly required to ensure placed components remain secure during handling and soldering [1]. While automated pick-and-place systems currently exist which can perform the lead conditioning and assembly tasks, these systems often must be custom designed to a dedicated category of components, which is expensive and time consuming. More importantly, odd component types which have non-standard characteristics are mostly incompatible with these systems, and a dedicated machine for every potential component would be impractical. A common solution to this problem is to perform the conditioning and insertion of components manually. In order accomplish

this, the boards must be removed from the high speed feeders and placed on portable racks which are taken to manual work cells occupied by human operators. The components to be placed are often kept in bins or trays at the work cells where they are conditioned, inspected and inserted as the boards arrive. The automated manufacturing process is interrupted until the manual insertions are complete and the boards are placed back on the line. This method introduces high variations in timing which makes precise process scheduling difficult. It also results in drastically reduced throughput rates in the assembly process and significant error increases.

Some attempts have been made to develop systems to assist the manual process. Halbo et. al [1] describes a hybrid automation system which uses light projection to visually highlight which type component to pick and its corresponding placement on the board. Using this system, an experienced operator may be capable of placing 100 components per hour, taking approximately 36 seconds per component, but achieving lower error rates than without the assistance. Consultation with a local PCB manufacturer quoted their workers being capable of achieving an average placement time of as low as 5 seconds per part, but experience typical errors of 1% or higher. This is significantly below the dedicated assembly machines which can process 50,000 parts or more per hour.

Research attempts have also been made to remove the human element using vision systems such as Liebes et. al [2] [3] who developed an assembly system which uses a SCARA robotic manipulator coupled with machine vision, referred to as the flexible assembly intelligent robot (FLAIR). A major difficulty involved with odd components is the need to develop custom feeding systems which can cause manufacturing delays and

increased prices. To mitigate this, the FLAIR system developed a modular feeder using a tube system. However, it was recognized that a disadvantage of this method is that it limits the compatible part types to those readily available with tube packaging. This system showed significant potential to handle a variety of nonstandard components. Although the assembly quality results were promising with this approach, reliable performance with odd-components was still a challenge [2]. Additional research methods demonstrating superior results were absent in the literature search, a claim supported in consultation with a major circuit board manufacturer which was actively seeking research to replace their current human work cells with more flexible automation systems.

Research relating to the general pick-and-place assembly operations is an ongoing field of development. Ono et al [4] used a vision guided robot approach to identify and pick coiled springs from a pile. The results showed the capability of the system to perform the operation despite the complex features of the springs. However, the performance may not exceed the capabilities of a human operator performing the equivalent task, with a sample success rate of 77%. Other approaches involved metal components such nuts, as well as more complex sheet metal shapes [5]. The placement quality showed up to 100% successful placement, but cycle times approached up to 15 seconds at an accuracy up to ± 1mm which would not meet the requirement of a circuit board assembly. Many other researchers have implemented similar methods which involve the use of a dexterous robotic manipulator guided by a machine vision system [6] [7] [8] [9], an approach which shows great potential for further research in circuit board assembly tasks.

1.2 Research Overview

The current manual circuit board assembly process involves many unique tasks before a component is inserted into the circuit board. However, the scope of this research is limited primarily to the automation of the pick and place task using a robot manipulator. The requirements to automate the pick-and-place operation include:

- localization of the electronic components within the part tray
- localization of corresponding placement positions on the circuit board
- picking up components and evaluating part quality
- insertion of component at corresponding localized placement location

The research contained in this thesis should provide supporting evidence that a precision robotic manipulator in concert with a calibrated machine vision system can be successfully used to meet the outlined requirements to automate the pick and place process for non-standard electronic components. Specifically, the desired outcome would show that the automated process can be performed with a lower number of placement errors and faster throughput rates over the current human work cells.

The proposed methodology to accomplish these research goals involves a 6-axis robotic manipulator with pneumatic clamp tooling, as shown in Figure 4 and Figure 5. A vision system comprised of two machine vision cameras supplement the necessary information for automated operation. Within the robots area of operation, separate workspaces are established to contain the tray of components as well as the circuit board.

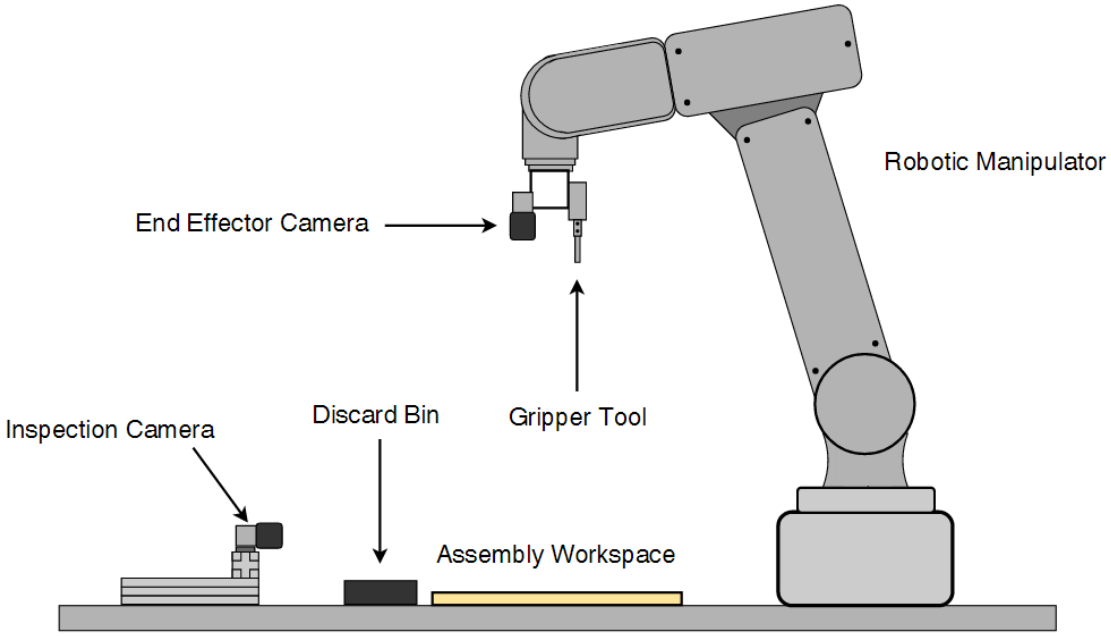

Figure 4: Experimental workspace elements (side view)

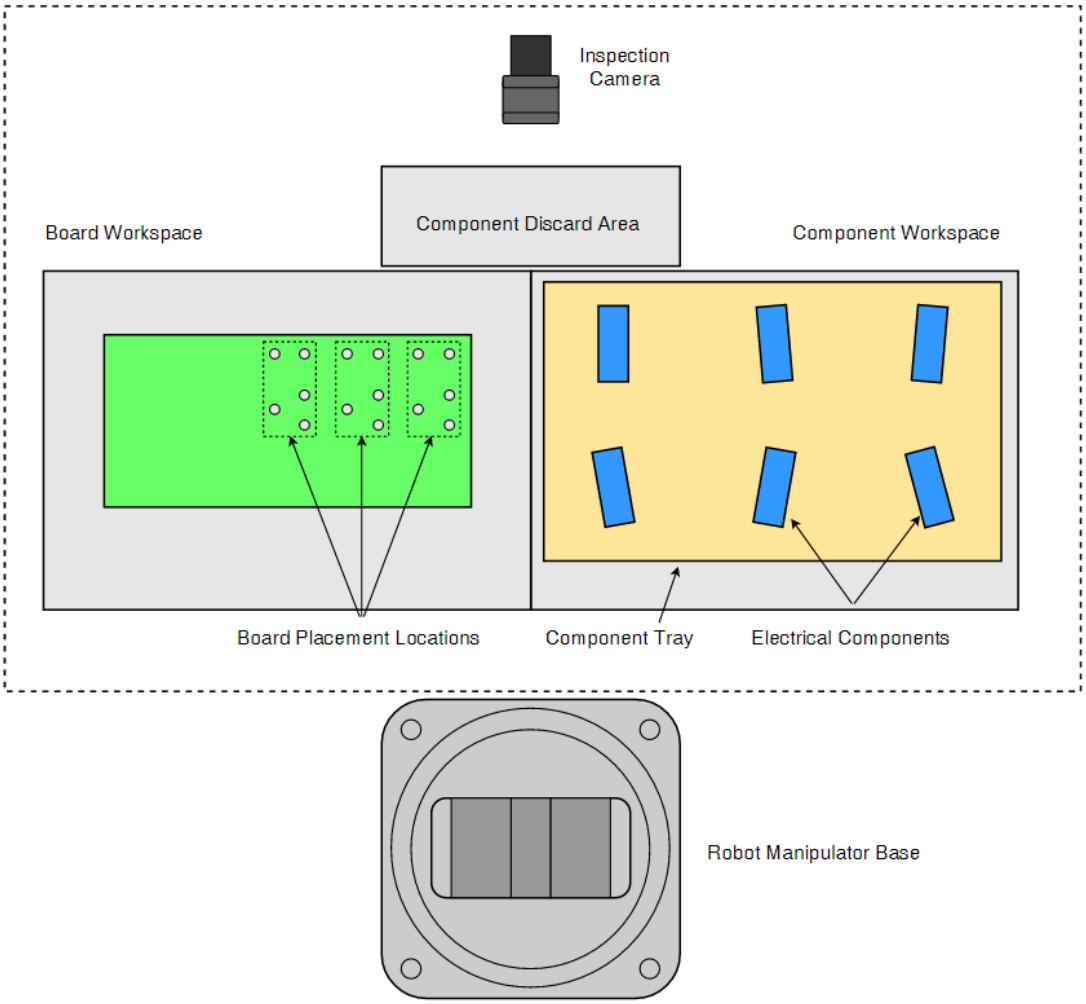

Figure 5: Experimental workspace elements (top view, rotated 90°)

An experimental procedure was developed to analyze the circuit board assembly performance of the system using. This procedure begins by first preparing a component tray and placing it in its workspace. Each tray contains only one type of components, but the quantity and pose may vary for different experiments. The board is similarly placed in the workspace using fixed mounting structures. Descriptive parameters regarding the component, circuit board, and task operations are input to the experimental program

followed by the command to begin the procedure. Once the process begins, the robot positions the camera at different viewing locations to acquire images of the tray and board which are used to identify and localize the components and board placements. Once this information is calculated, a single component is picked and positioned in front of the inspection camera at multiple viewing angles to determine precision pin offsets. Applying these offsets to the reference placement coordinates, the gripper inserts the component into the circuit board and releases it. This process can then be repeated as dictated by the particular testing goals.

CHAPTER 2: SYSTEM HARDWARE

2.1 Robotic Manipulator

A Denso VP-6242 was used to perform the mechanical manipulation tasks. The VP-6242 is a robotic arm with 6 degrees of freedom (DOF) capable of moving an end effector to any point in 3 dimensional Cartesian coordinates within its workspace. The arm has an overall length of 420 mm with an effective workspace radius of 432 mm as depicted in Figure 6 and Figure 7 [10]. This area was sufficient due to the small nature of typical electrical components and circuit boards used in this research.

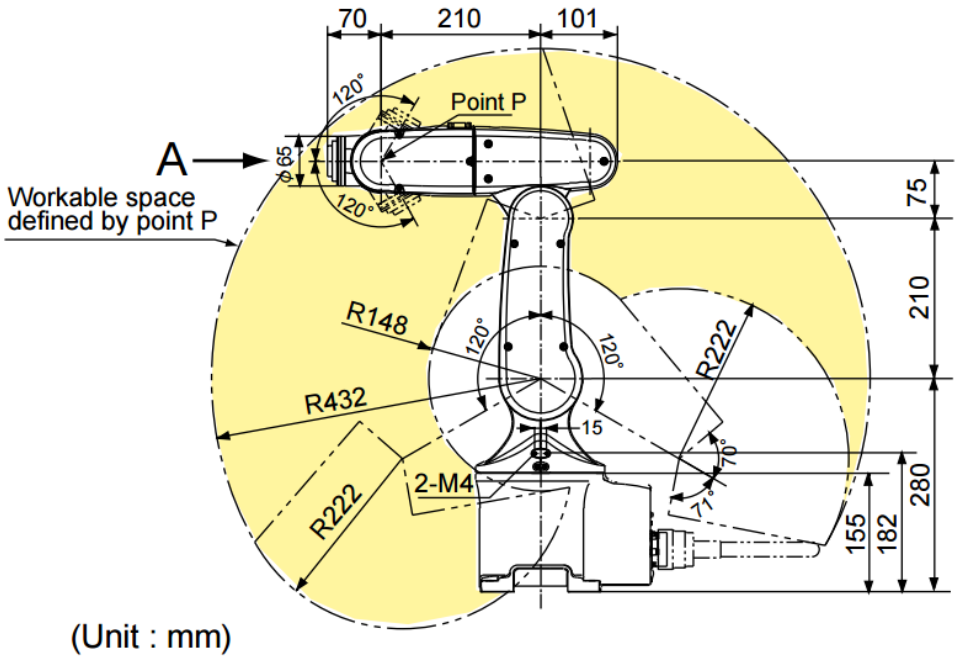

Figure 6: Robotic manipulator maximum work space- side view [10]

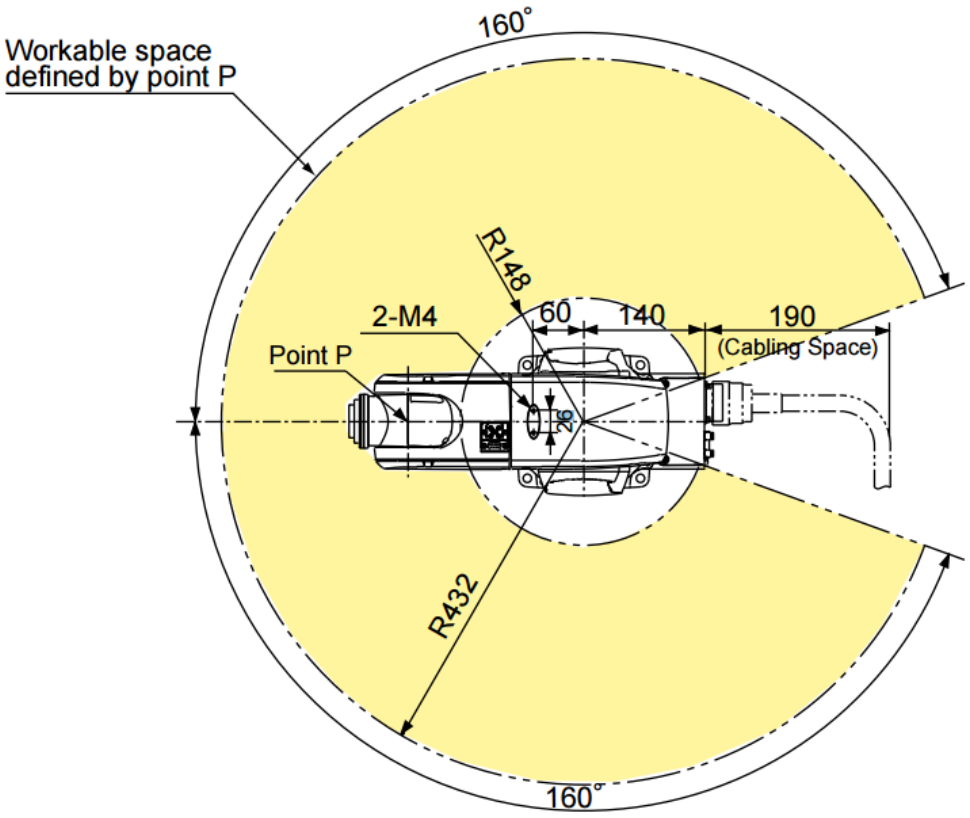

Figure 7: Robotic manipulator maximum work space- top view [10]

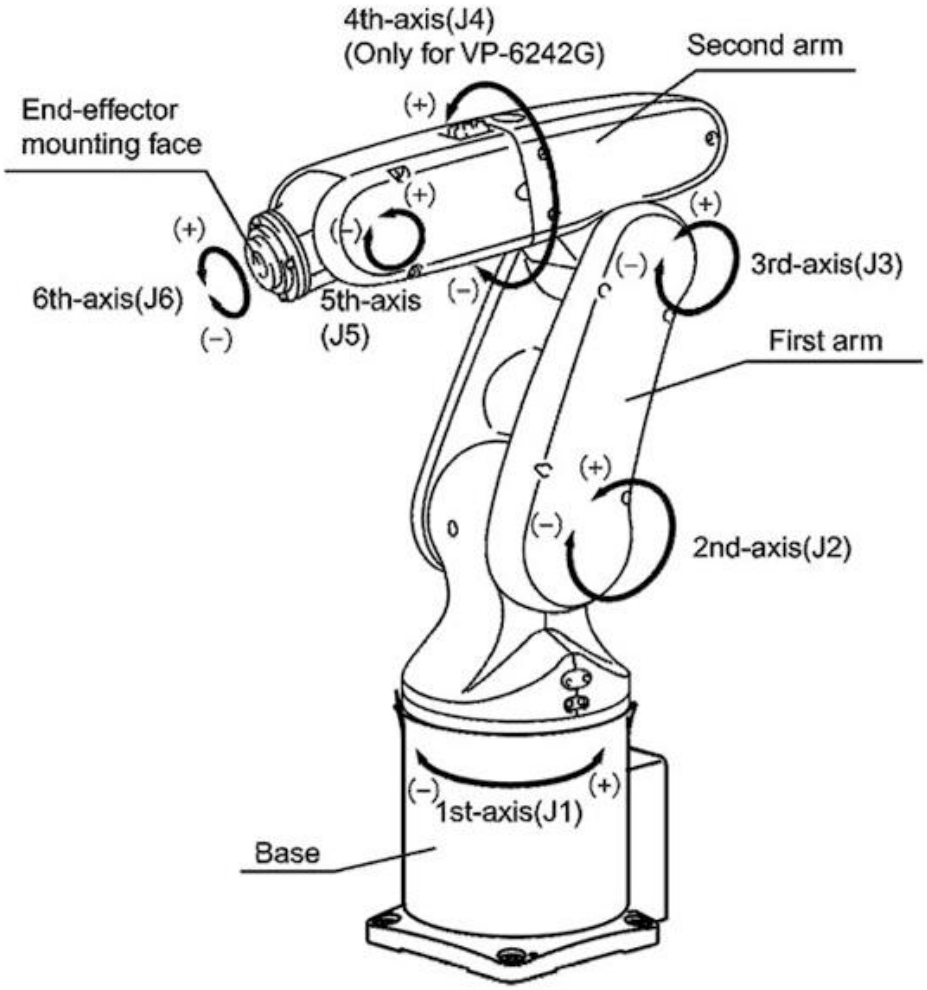

Figure 8: Robot axis labels [10]

The manipulator uses AC servomotors for each joint with absolute encoder feedback, which is critical to prevent drift error accumulation and to maintain positioning resolution over time. The robotic arm was mounted by the base on the surface of a workbench measuring 3x6 feet (0.914x1.83 m) as shown in Figure 9. In order to ensure an even flat work surface and to minimize vibrations caused by arm movements, the robot had been previously mounted to a structurally rigid steel workbench.

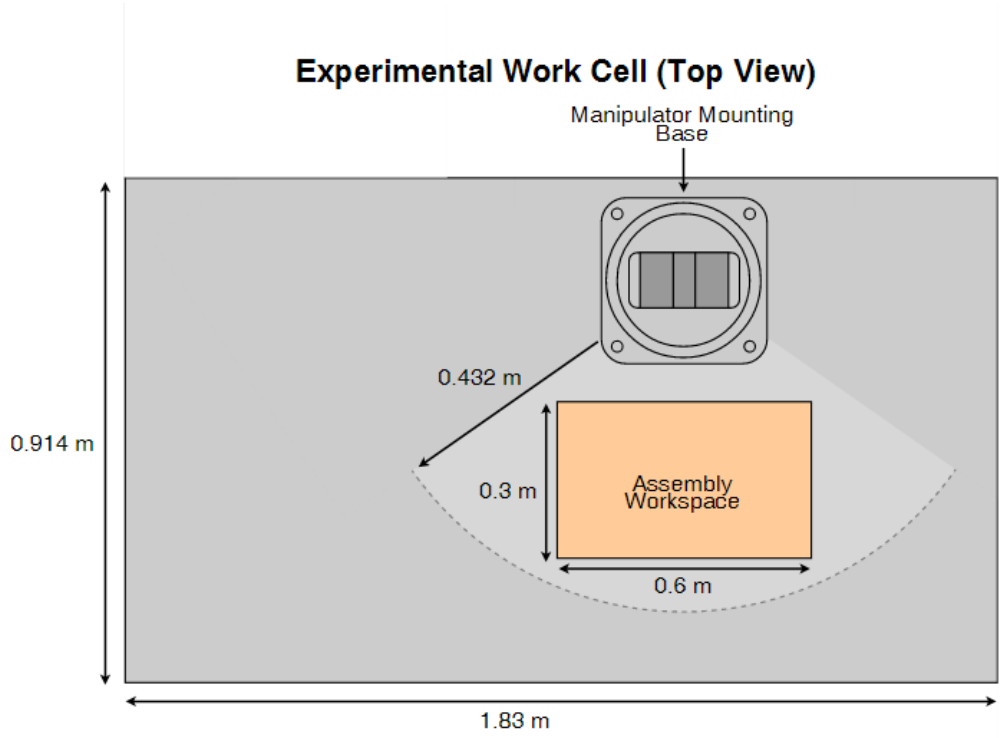

Figure 9: Robot work space dimensions

2.2 Robot Controller

The kinematic movements are controlled by an RC8 robot controller, also manufactured by Denso, which was designed to drive the VP-6242. The controller directly connects to the arm base to provide power, motor control and encoder feedback. The controller additionally provides 48 points of user input/output (I/O) signals, which includes signals for manual/automatic modes and emergency stop switches. When an emergency stop signal is sent, it directly deactivates the motor power relay to prevent any further movement. One emergency stop was located on a portable monitoring and control interface referred to as the teach pendant, and two fixed emergency stops were mounted near the work area, one on each end of the workbench. The I/O signal wires were routed

into a small breakout box, where the manual/automatic switches were also mounted to for easy access.

The arm contains internally routed pneumatic tubing, which was utilized by the end effector sub-system. A pneumatic air compressor was used to provide the pressurized air, which was routed through a solenoid valve to control its flow. This solenoid was activated by a relay controlled by an I/O signal from the RC8 controller.

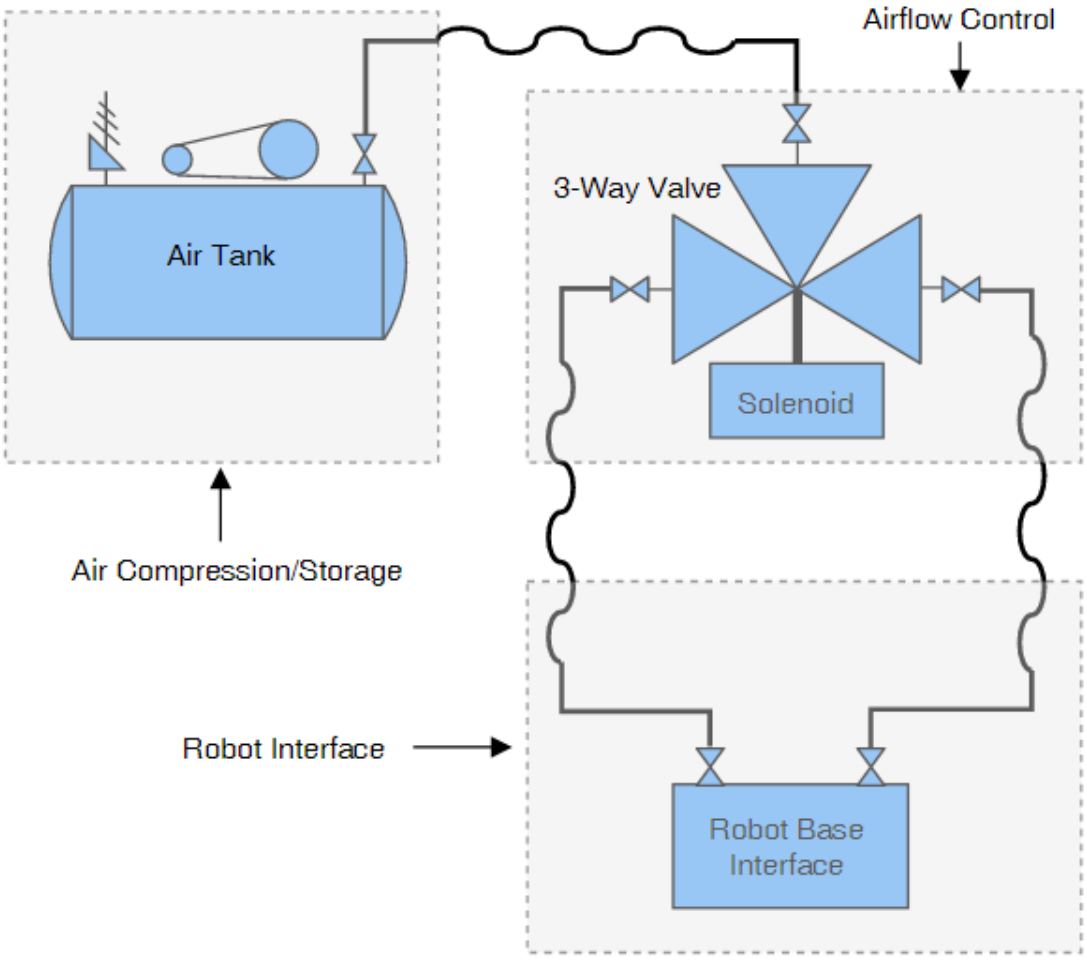

Figure 10: Pneumatic system overview

2.2.1 Controller Interfacing

The controller uses a programmable automation controller script (PacScript) language to execute program routines and movement commands, which is derived from Microsoft Visual Basic. This controller was interfaced through three methods, the initial being a teach pendant. The pendant is a tethered handheld device that provides basic interaction with the arm, such as applying power to the motors, moving the joints, and monitoring position. The pendant was primarily used in the initial setup and familiarization, but did not have the capabilities to perform complex automation tasks. The primary interface utilized was National Instruments (NI) LabVIEW System Design Software. The base development module of LabVIEW provides powerful tools for control system automation and data analysis with the capacity to utilize a variety of third party libraries. One such library is maintained by DigiMetrix in collaboration with Denso and NI that allows the direct use of LabVIEW to interface to the RC8 controller. This library converts high level function into recognizable PacScript format, which eliminated the need to directly use the PacScript language. Some of the main high level commands utilized are as follows:

- Move by coordinate name: move tool from current position to a desired position
- Approach: perform movement to a desired position offset by a variable distance along the z-axis
- Depart: perform a movement a variable distance along the z-axis
- Move Arch: perform a movement to a desired position according to variable arch parameters

To communicate with the RC8 controller, the “auto mode” signal lines must both be simultaneously pulled to 24V which allows commands to be received and executed through the Ethernet port on the RC8. This was beneficial by allowing the robot to be controlled through any network capable machine, including a host computer and the Real Time controller used by the vision system.

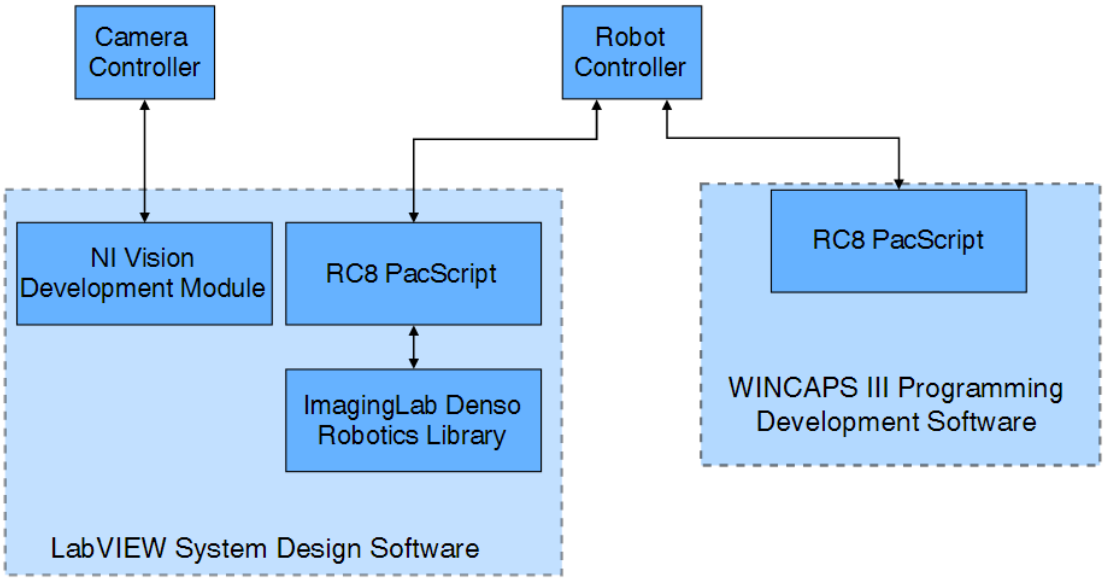

Figure 11: Programming software overview

2.3 End Effector

The particular end effector tooling for this research needed to be identified and mounted to perform the specific actions required, such as interaction with objects and visual inspection of multiple areas. A pneumatically powered type of gripper was selected to perform these tasks. Two gripper models had been previously acquired: the MHC2-20D and MHZ2-20D. The only major difference between the two models was that the MHC opened angularly, while the MHC opened linearly.

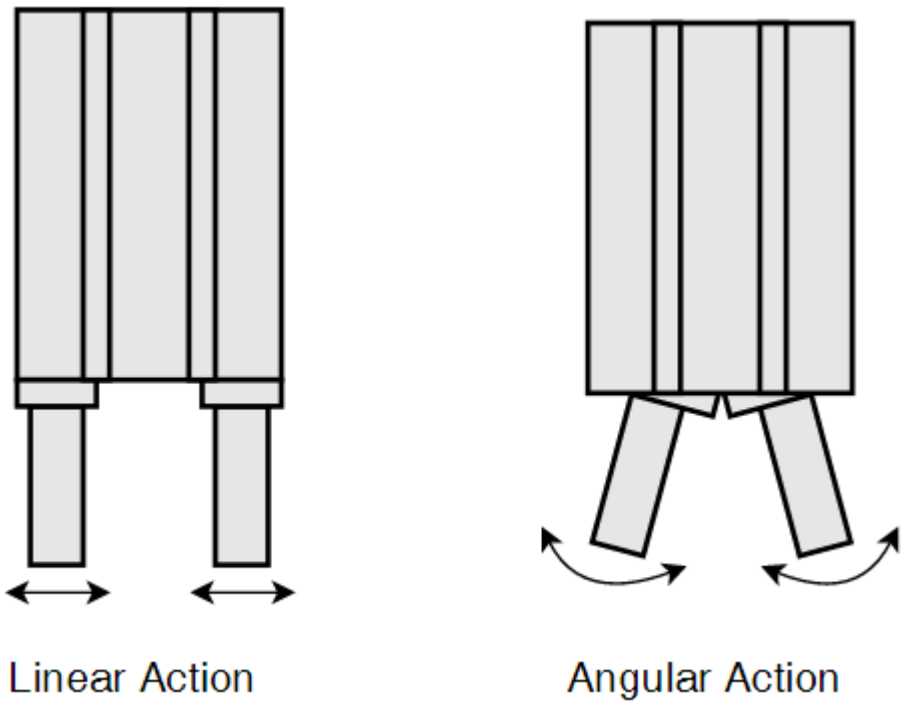

Figure 12: Parallel vs. angular gripper style

One other task requirement was the ability to obtain visual information from multiple viewing points for automation feedback. Due to a limited number of cameras, and for cost efficiency, it was determined that a single camera could accomplish multiple tasks if located directly on the end effector, allowing it to move to transition to multiple viewing positions or angles. In order to mount these devices to the robot, a bracket was designed to make use of the mounting holes on the J6-axis flange (Figure 13).

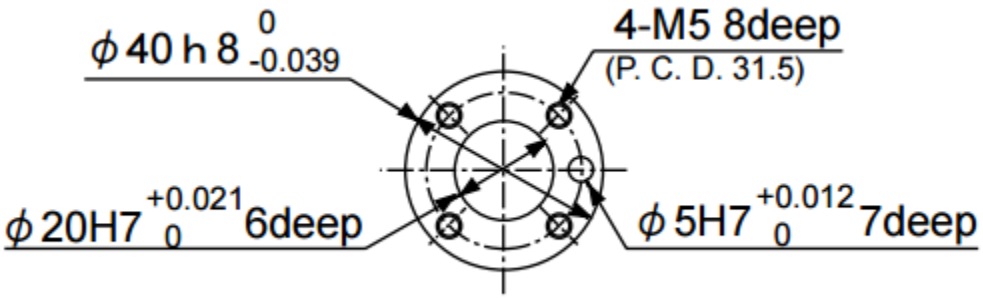

Figure 13: Robot flange mounting face [10]

Square aluminum tubing was chosen as the foundation of the bracket as it can be easily purchased, ensures consistent projection alignment, and provides structural rigidity to reduce movement errors. The tubing required only to be cut the appropriate length, and for holes to be drilled at precisely measured locations, resulting in a bracket made of only one single component to prevent compounding misalignment errors. An error occurred in the machining of this bracket, however, resulting in the camera and gripper mounting locations being incorrect. Since the offsets were still constant, they were accounted for in calibration rather than re-machining a new bracket.

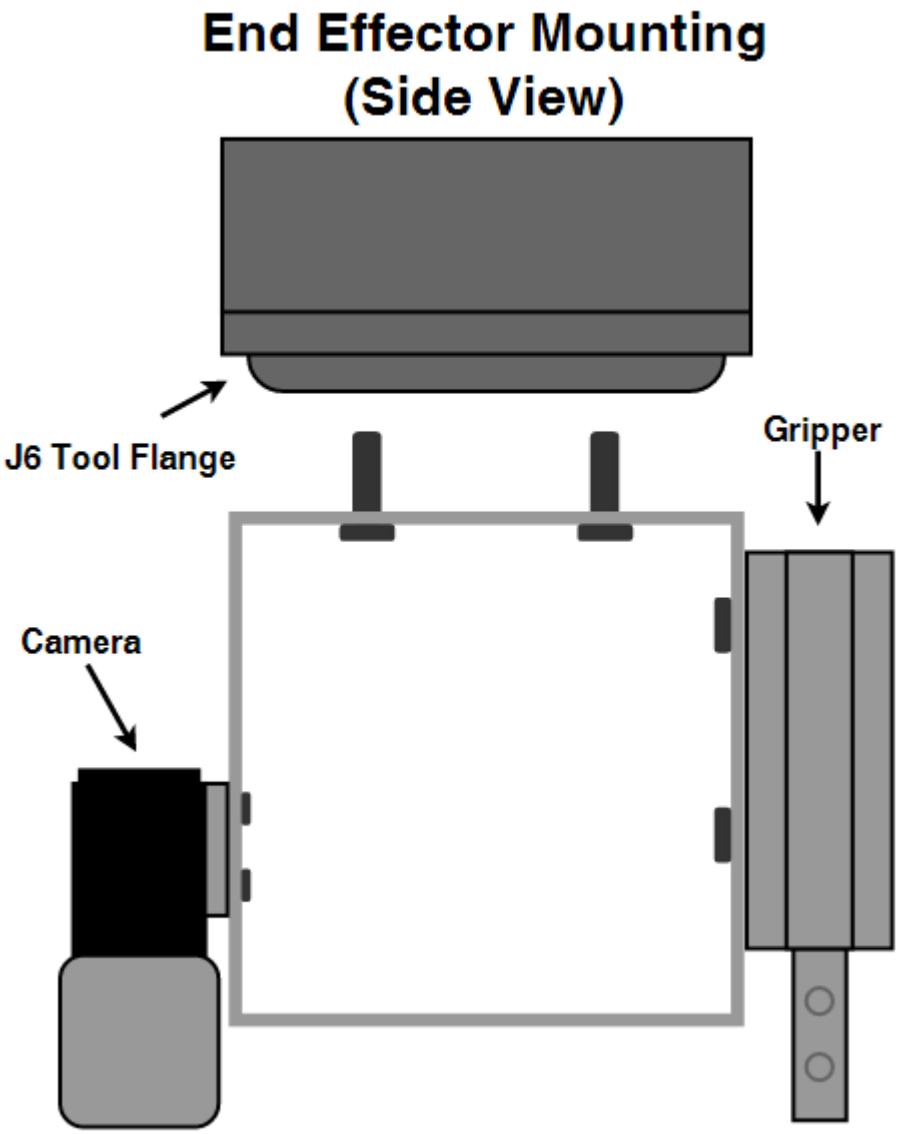

Figure 14: End effector mounting

The gripper assemblies provide mounting holes for attachments such as fingers, which were developed based upon the particular component grasping requirements, but the optimization of the fingers was not to be a primary focus of this research. The final finger design was created with considerations for minimizing implementation time.

Aluminum corner brackets were selected as the foundation of the fingers. The reasoning for the 90° angle was to allow a surface with a larger area that could be utilized for other functions, such as placement of fiducials to assist in visual localization, and this shape also allowed the material to be thinner and lighter while maintaining acceptably rigid characteristics. This was taken into consideration due to the moment of inertia produced at the greater distance extension, which can result in increased positioning time and reduced accuracy.

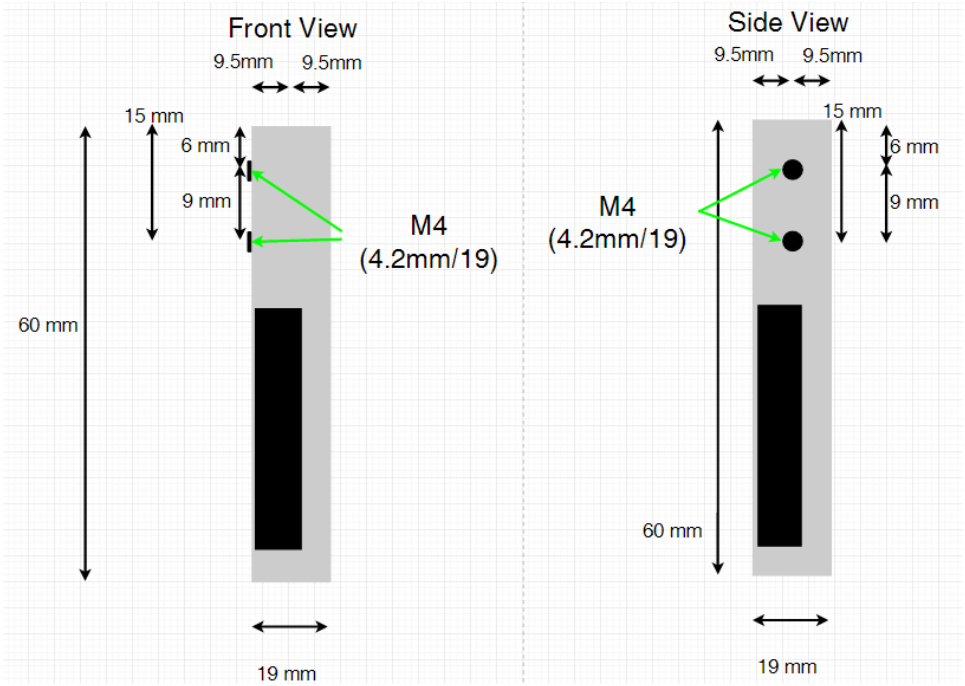

Figure 15: Final gripper finger design (front & side view)

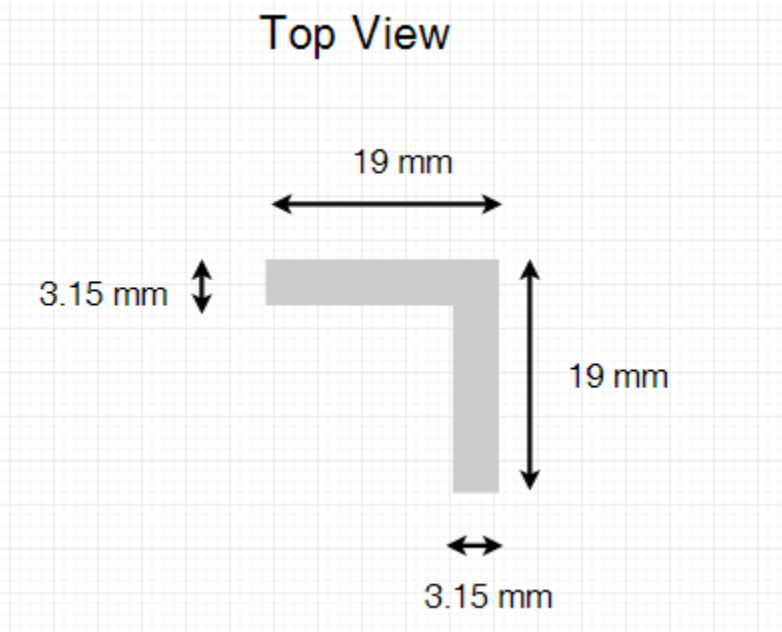

Figure 16: Final gripper finger design (top view)

The aluminum was cut to the appropriate lengths and a mill was used to ensure the alignment of the mounting holes was consistent. As a final measure to reduce compliance and alignment errors, shim stock was inserted between the gripper and the inner surface of the fingers. This design provided the precision and repeatability necessary for other aspects of the research.

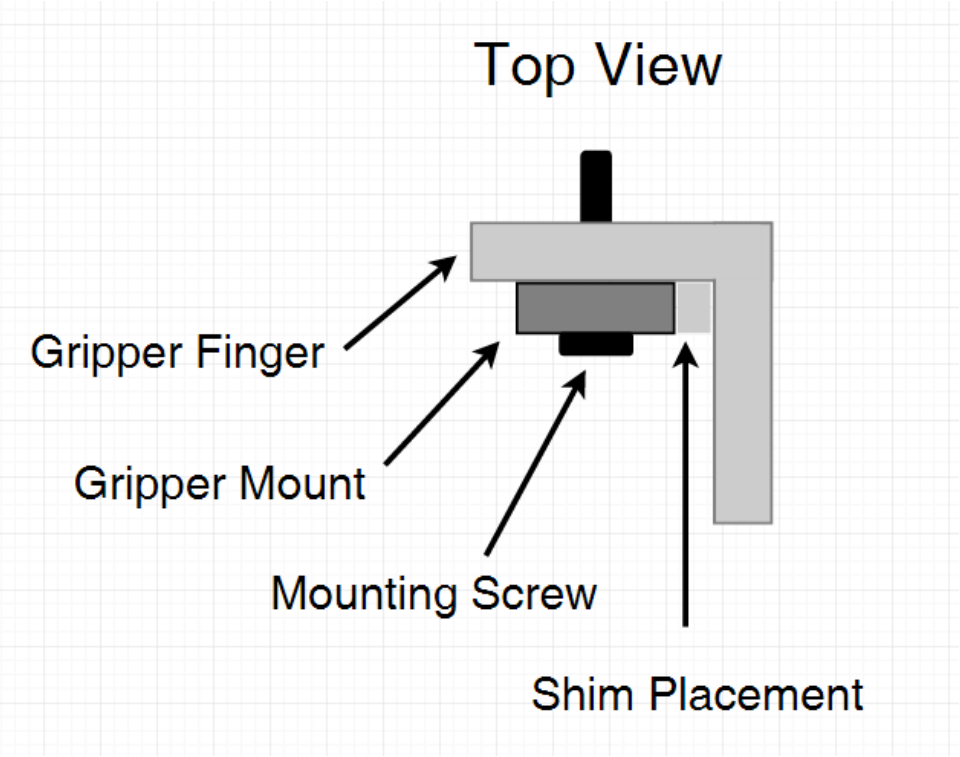

Figure 17: Gripper finger mounting alignment

The maximum load of the end effector was given in the specifications at 2.5 kg with the wrist direction facing down and 2 kg otherwise. However, the distance this load is applied results in a moment of inertia which also must be considered. The maximum inertial moment allowed around the J6-axis is 0.007 kgm^2. The total mass of the end effector was estimated to be approximately 1 kg and the total inertial moment of the end effector assembly was calculated to be less than 0.001 kgm^2 which was well within the robot's specifications. At these loads, the positioning time of the end effector can be estimated using the manufacturer provided graphs displayed below [10].

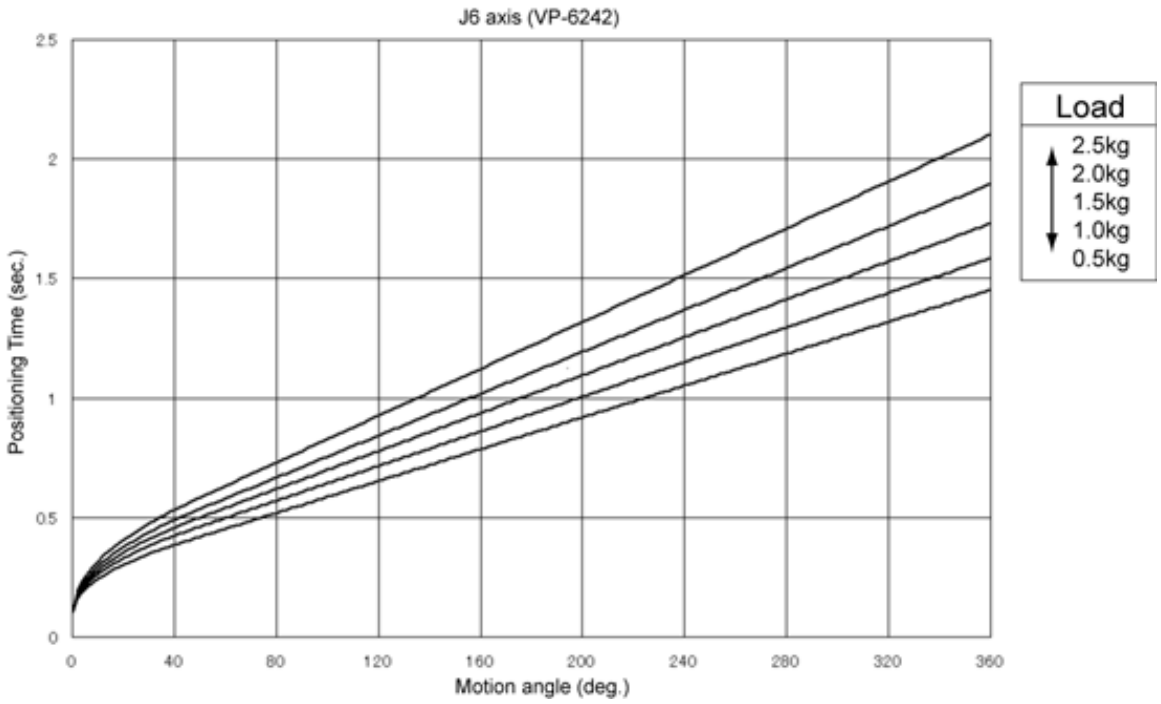

Figure 18: Robot J6-axis positioning time [10]

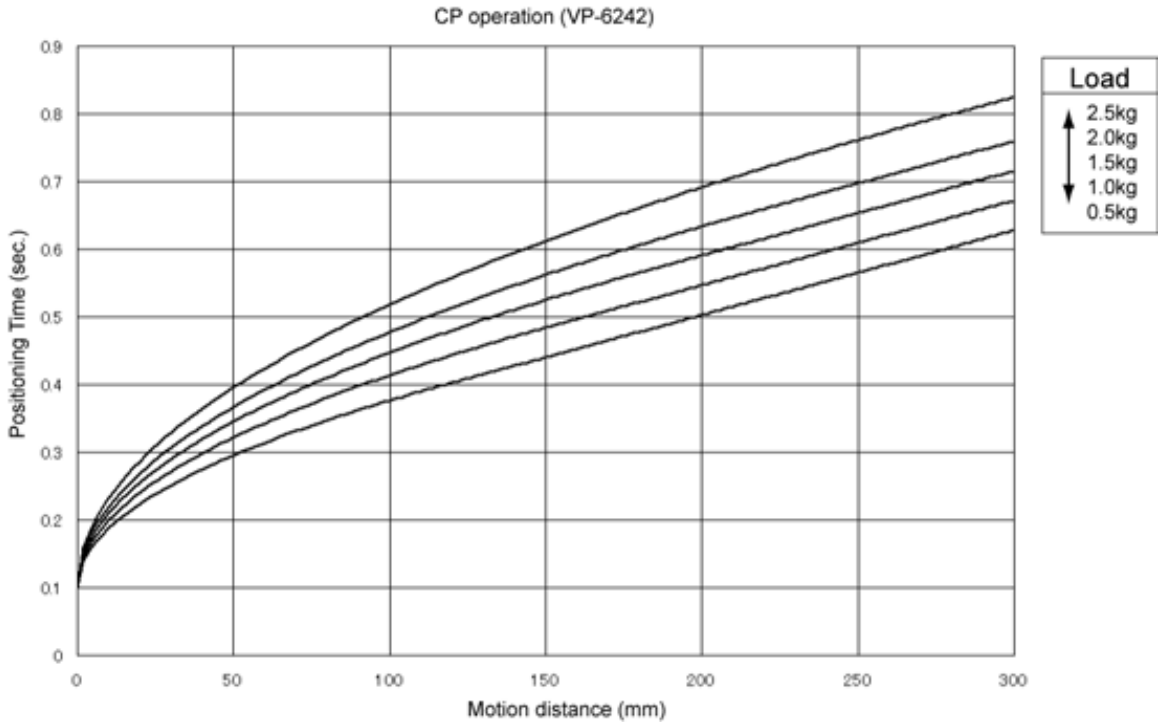

Figure 19: Robot continuous path positioning time [10]

The movements of the robot which limit the overall positioning time are the rotation of the J6-axis and the translation of the tool between positions. Large rotations require significantly more positioning time and may occur both when picking angled parts and during inspection, and can become a limiting factor in total positioning time. Assuming that no more than ±180° would be necessary to pick a component of any orientation, Figure 18 indicates a maximum positioning time under rotation of approximately 0.9 seconds at an end effector load of 1 kg. According to Figure 19, the positioning time becomes linear in relation to distance at approximately 100 mm taking 0.42 seconds. This information is useful for path planning in order to minimize the overall positioning time requirement.

2.4 Lighting Control

The workbench used for this research was located in a laboratory environment with overhead fluorescent lighting fixtures. These lighting conditions were non-ideal for many of the machine vision tasks for a variety of reasons, including intensity variations, shadow casting, and specular reflections. The control of machine vision lighting can become a complex task in itself. While a variety of lighting techniques are able to solve some lighting challenges, no universal lighting solution exists. One technique known as dark field lighting is based on directional lighting which forms relatively low incident angles with the camera as shown in Figure 20 [11]. This method is highly effective for reducing glare on specular surfaces by directing reflected light away from the surface at the same incident angle, rather than towards the camera.

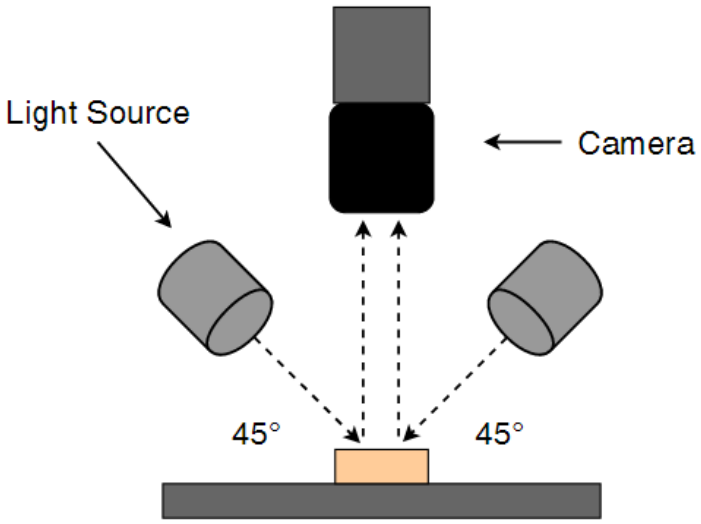

Figure 20: Dark field lighting concept

2.4.1 Lighting Setup

Nearly every electrical component has metallic features which are highly specular in nature (e.g. pins). Circuit boards also exhibit these characteristics from features such as mounting pads. To reduce glare while maintaining texture features and workspace interaction, a dark lighting approach was implemented; however, the optimization of the lighting conditions was not a research objective.

A large cubic frame enclosure, with four legs and a solid top, was placed on the work surface in order to block ambient light originating from directly above the system while allowing light to enter from the side openings, as shown in Figure 21. This frame needed sufficient vertical clearance for movement of the manipulator, which resulted in a higher than desirable incident angle of light in the workspace. Completely removing light from this direction was observed to unfavorably increase the presence of shadows, which decreased the effectiveness of many vision tasks. A compromise was made by using light diffusion cloth to reduce the coherence of the light while sustaining the approximate quantity. This allowed the average light intensity to remain higher while reducing the glaring effect caused by high incident light angles. In the event that a particular object required higher contrast, additional light fixtures were placed inside the enclosure frame which provided further diffused lighting. Linear fluorescent bulbs were chosen to distribute the light more uniformly than the point source of a lamp style bulb.

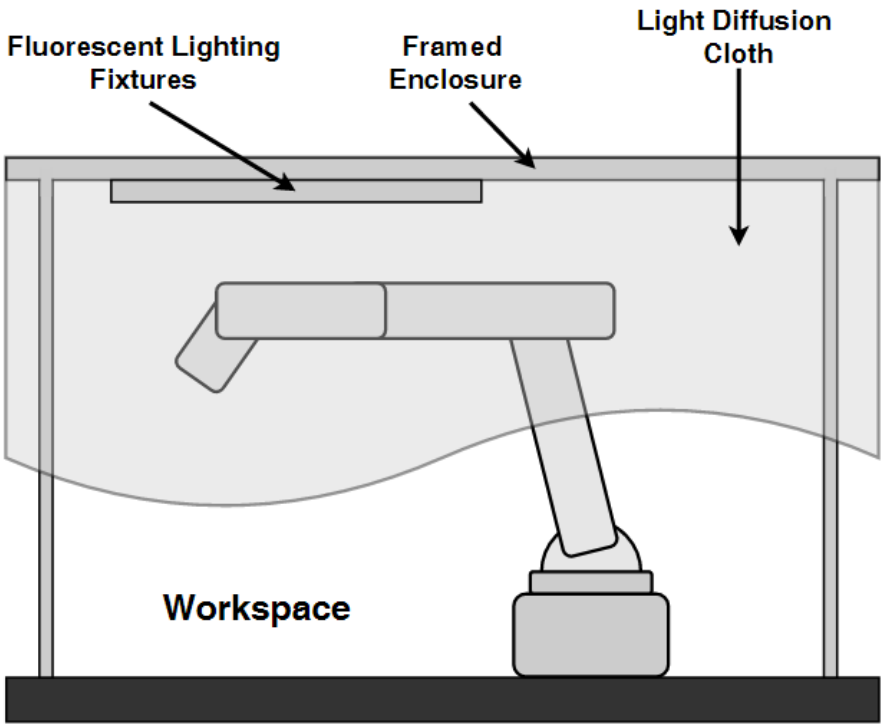

Figure 21: Lighting control system

2.5 Vision System

One of the main components of the research relied on the use of an imaging system to obtain visual information of the work area and objects of interest. This system was also used to perform real world object localization and measurement, and therefore needed to be of high enough quality to meet the precision requirements. Two camera and lens assemblies believed to meet these requirements were selected. Each camera was a Basler area scan machine vision camera coupled with Computar low distortion lenses that consisted of the following specifications:

Table 1: Camera hardware specifications

Camera	Sensor Type	Output Format	Resolution (Pixels)	Frame Rate (fps)	Lens Focal Length (mm)
acA1300-30uc	CCD	Color	1294x964	30	12
acA2500-14uc	CMOS	Color	2590x1942	14	8

The camera used to view the board and components benefitted from a larger FOV by allowing more components to be analyzed and placed on a larger board. The lens with the shorter focal length was paired with the higher resolution camera to allow a larger viewing area while mitigating the loss of feature resolution. Each camera communicated through a USB3 interface, which allowed the necessary throughput to operate in real time.

The information obtained from each camera needed to be transferred, processed, and analyzed in real time to minimize the process time. To do so, a National Instruments CVS-1459RT compact vision system was selected as the real time controller. This device was compatible with LabVIEW software as well as the Denso robotics library. A host machine was installed with all of the software and configuration files, which could be accessed from any other machine on the network.

CHAPTER 3: ROBOTIC MANIPULATOR CALIBRATION

3.1 Coordinate Systems

The manipulator position is expressed using a 3 dimensional Cartesian coordinate system referenced to the center of the flange surface located at the end of the J6-axis. This position can be defined through the use of unique coordinate systems, which are the Base, Work, and Tool coordinate systems [10]. Each of these coordinate systems are useful for determination and command of positional set points relative to various points of interest. The Base coordinate system is absolute and referenced to an origin located at the center of the robot base as shown in Figure 22. All other coordinate systems are established with respect to the base coordinates. The Work coordinate system can be defined through any translation or rotation of the Base coordinates, allowing the origin to be relocated to any point of interest within the workspace. This system can be dynamically redefined as needed, making programming movements simpler and faster.

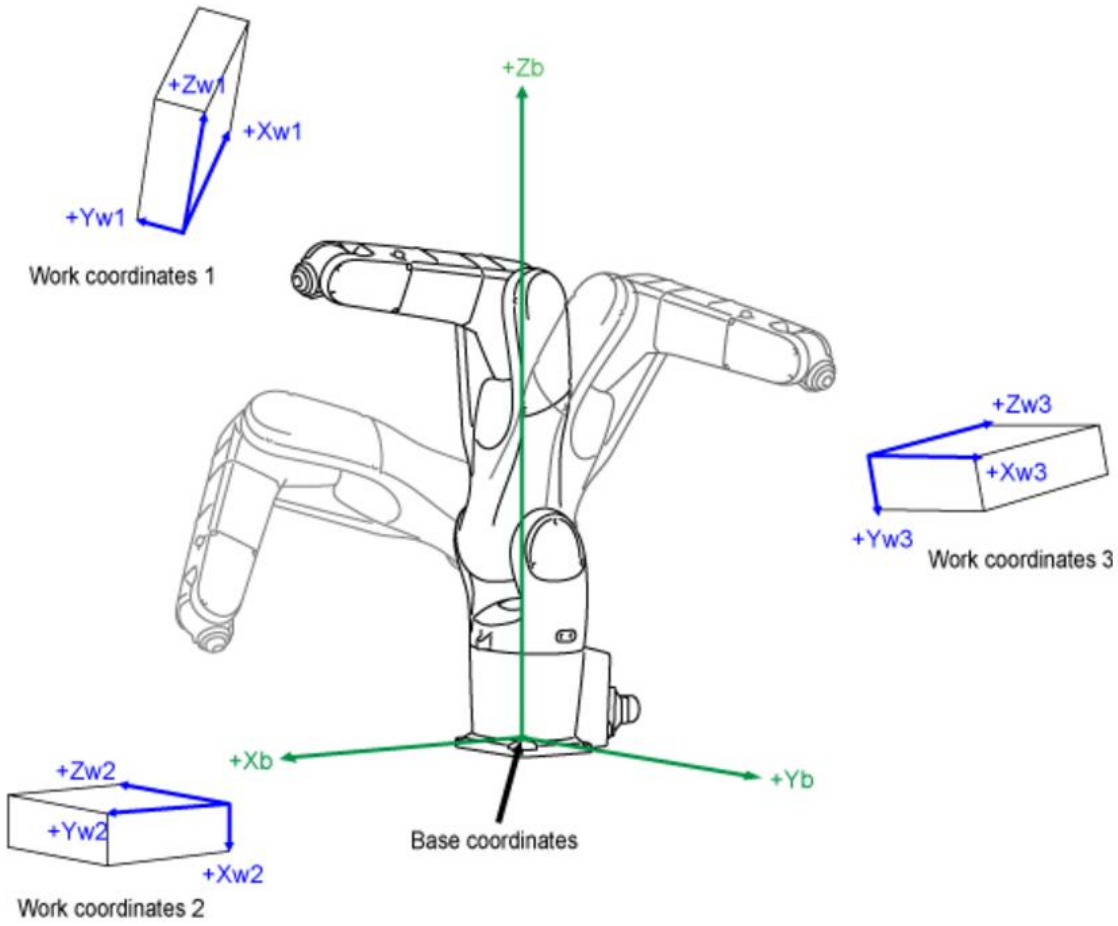

Figure 22: Robot base and work coordinate systems [10]

Before defining the Tool system, it is necessary to define the mechanical interface coordinates, which are a fixed coordinate system established with respect to the center of the flange surface on the J6-axis. As shown in Figure 23, The Zm-axis is normal to the center of the flange, with Ym pointing in the direction of the orientation key hole [10]. Xm passes through the center of the flange and is orthogonal to Ym and Zm and the intersection of these three axes is the point used to determine the robots position in relation to the Work or Base coordinates. The Tool coordinates can be established through rotation or translation of the mechanical interface coordinates as shown in Figure

24, similar to the process of defining Work coordinates. Using Tool coordinates allows the robot to be positioned relative to more useful points of interest, such as the center of the gripper, which enables more precise manipulation of components.

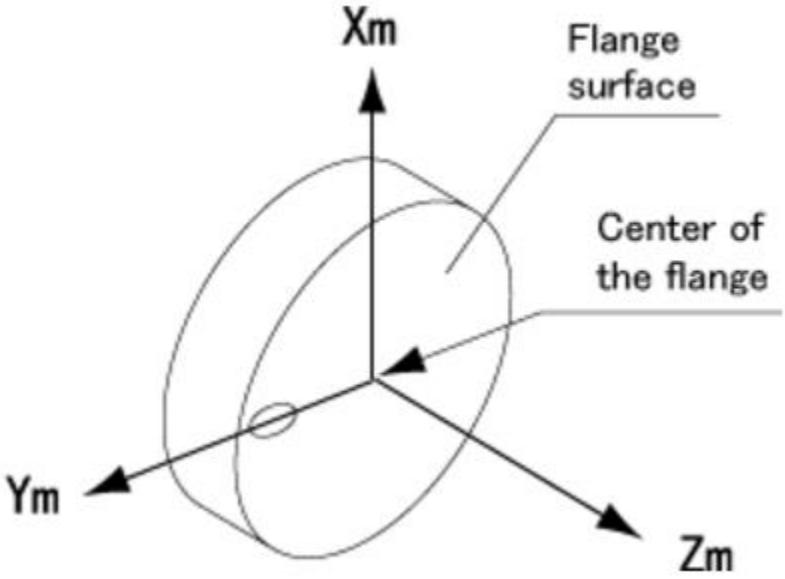

Figure 23: Robot flange coordinate system [10]

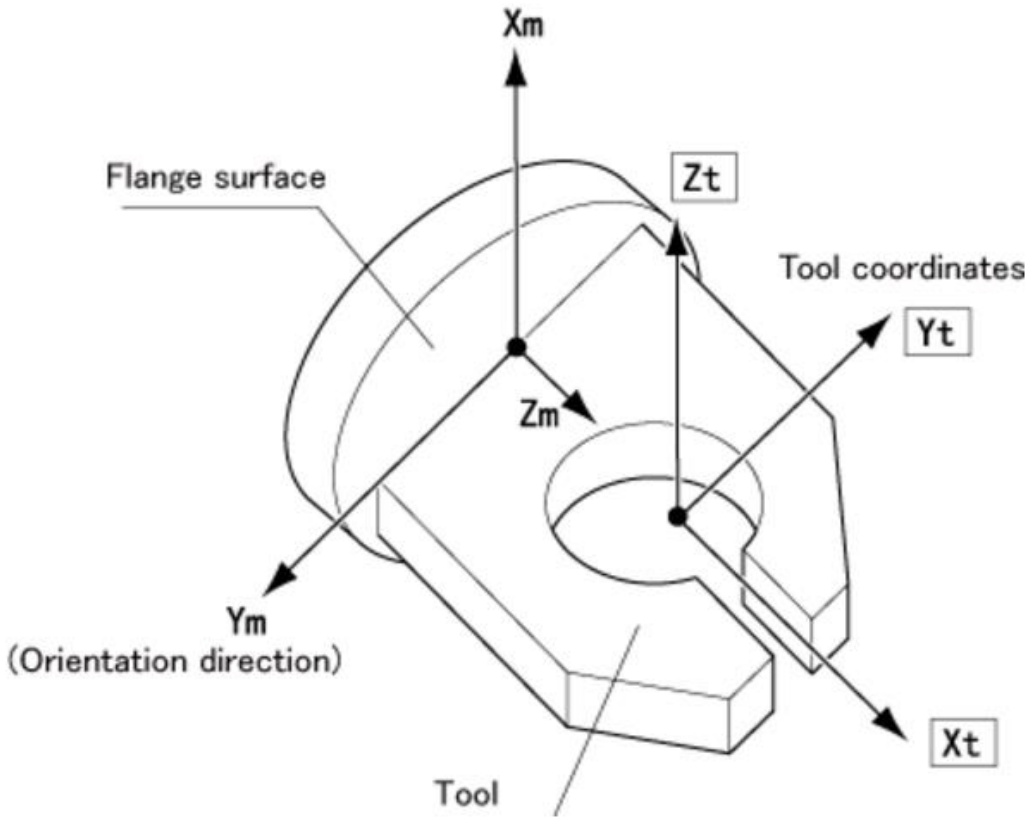

Figure 24: Robot tool coordinate system [10]

3.2 Tool Coordinate Offset Measurements

When an end effector is mounted on the mechanical interface, the resulting offsets of any particular point of interest must be precisely measured to ensure consistency between the modeled reference point and the physical reference point. A variety or error sources must be considered to achieve high precision requirements of this research. Seemingly small measurement errors can accumulate to produce non-linear errors in positioning.

Before attempting to calibrate the tool offsets, the orientation of the Base coordinates first needed to be determined as a reference. The most fundamental method to accomplish this is to identify workspace reference points that correspond to manipulator positions.

This was done by manually moving the arm along the x-axis and y-axis separately and tracing the resulting paths to determine the X and Y directions. In order for this method to be valid, it was assumed that the robot XY plane was approximately parallel to the work surface.

When a new tool was attached to the end effector, the coordinate offsets were determined through the determination of the tool tip with respect to known mechanical interface coordinates. A physical feature needed to be placed to identify the locations of robot position reference points on the work surface. To maximize the certainty of the true origin of a reference point, the size of the identifying features needed to be sufficiently small. This was accomplished by using a sharpened needle with a very fine tip. A soft but rigid foam board was placed on the work surface. The needle was firmly secured in the inner channel of each gripper finger with just the tip protruding from the bottom. For each (x,y) reference position the needle was inserted into the foam along the z-axis to create an indentation. The next step was to accurately measure distances between these points. The manipulator was chosen as the measurement device as it is capable of a repeatability of 0.02 mm in each axis, which is far superior to measurements done by hand. To perform a measurement, the needle tip was manually positioned from one position to another, tracking the displacement distance in each axis according to the following equation:

$$x_t = \frac{[x_{t0} - x_{t1}]}{2} \tag{1}$$

$$y_t = \frac{[y_{t1} - y_{t0}]}{2} \tag{2}$$

The tool was manually stepped visually until it could be re-inserted into the indentation of interests without causing any deformation. This method ensured the ground truth origin, or the point used to identify the reference point, was equal or less than the width of tip. The maximum error of the arm is rated at ±0.0283 mm in between two axes, therefore the maximum limiting error is described by the following equation and represented by Figure 25, where ε_{arm} is the maximum error of the arm and ε_{gt} is the maximum error of the ground truth origin defined by the radius of the needle tip.

$$\varepsilon_{max} = \varepsilon_{arm} + \varepsilon_{gt} \tag{3}$$

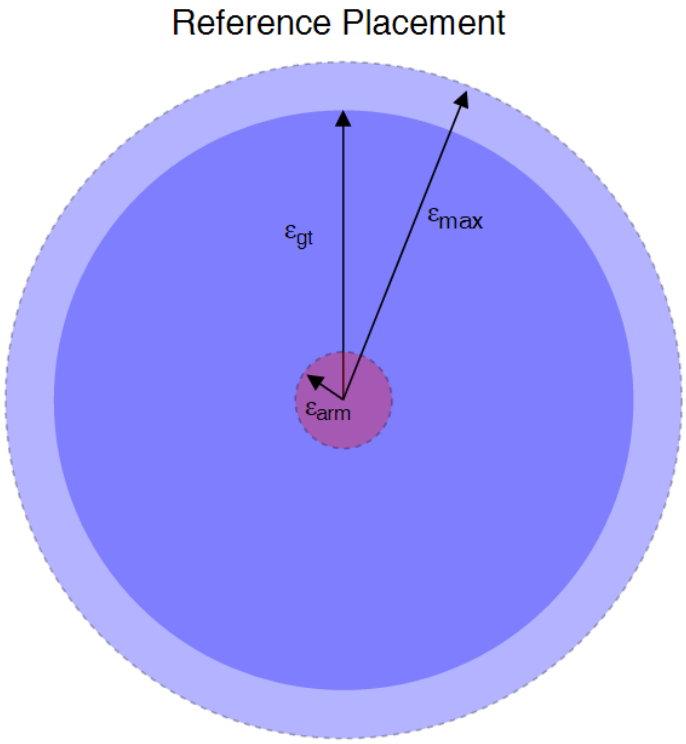

Figure 25: Reference position imprint error analysis

Using (4), the displacement measurement error can be described using the following equations, which indicates the maximum error using this method would be equal or less than $|2\varepsilon_{max}|$. For a sharpened point with a diameter at the tip of 0.05 mm, the maximum resulting offset error would be approximately 0.14 mm in each axis, but would likely be less than this value.

$$x_t = \Big((x_m \pm \varepsilon_{max}) - \big(x_{ref} \pm \varepsilon_{max}\big)\Big) = x_m - x_{ref} + |2\varepsilon_{max}| \tag{4}$$

$$y_t = \Big((y_m + \varepsilon_{max}) - \big(y_{ref} + \varepsilon_{max}\big)\Big) = y_m - y_{ref} + |2\varepsilon_{max}| \tag{5}$$

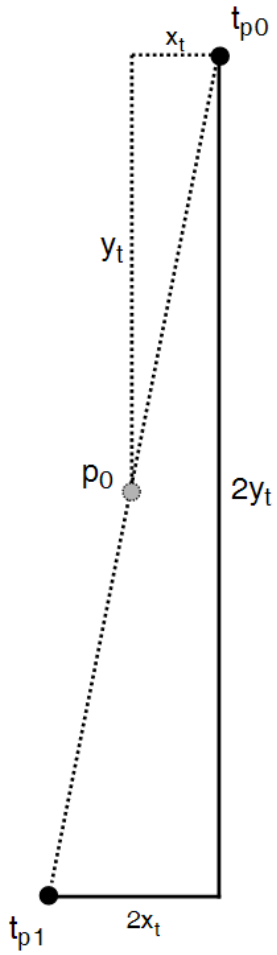

Figure 26: Determination of robot tool offsets

Once calibration offsets were obtained, the accuracy and precision of the calibration was tested. The base coordinate system previously traced represents the positions which should now correspond to the calibrated tool reference point. If there is any significant error, a recalibration is necessary. By rotating the tool around a given point, any calibration errors can also be measured as deviation from the set point as the tool rotates.

In order to determine if the calibration quality was within the maximum tolerance, the maximum allowable error needed to be determined. This error margin is limited by the

maximum error margin of the other components of the system during the task execution. Considerations needed to be made with respect to the tolerance of the components to be placed, spacing tolerance of the circuit board through holes, and the maximum feature resolution of the imaging system.

CHAPTER 4: MACHINE VISION

A critical component to automating the assembly procedure is the use of the vision system. The information available in images provides the capability to recognize and localize objects, obtain precise geometric measurements, and perform inspections. The LabVIEW vision development module provides powerful library tools to implement the necessary machine vision algorithms; an in-depth discussion of their operational concepts is provided in the following sections.

4.1 Vision Fundamentals

A digital camera is a device which uses an electronic sensor to convert light intensities into a digital representation. These sensors are typically a rectangular arrangement of phototransistors with dimensions (x,y), where the relationship of x/y is known as the aspect ratio, and the number of sensor elements along each dimension defines the maximum image resolution. When light hits one of these sensors, the energy is converted to an electrical signal proportional to the lights intensity which can then be transferred to an image file, where they are referred to as picture elements, or pixels. The maximum rate at which the full number of pixels can be transferred is the cameras frame rate. For a given amount of light from a scene is directed at a camera sensor, ideally the number of sensor elements should be maximized, but to keep the same amount of light reception, the sensor area should be minimized to reduce the signal to noise ratio (SNR) [12] [13].

Digital images are represented as either a two dimensional matrix for grayscale images, or a three dimensional matrix for color images where each of the RGB color scale has a unique two dimensional representation [12] [14]. Each element of the matrices is a spatial representation of a corresponding pixel location and its numerical value represents the luminous intensity at the time of the sensor exposure. Representing images this way allows them to be operated on using many conventional digital signal processing techniques.

One of the most basic objective of machine vision for this research is the identification and recognition of image features. According to Farinella et al. [15], a quality feature should be distinctive from spatial surroundings, invariant to irrelevant changes, robust to noise, comparable to similar content in other images, and highly relevant to the current application. Some of the fundamental categories of features include: intensity, edges, contours, shapes, and texture. The images acquired from camera sensors contains noise that often must be filtered before analysis in order to produce reliable results.

The fundamental operator of the filtering process is performed using convolution of a whole image or a region of interest (ROI) within an image. A 1D or 2D kernel is convolved through each pixel index in an image replacing the pixel value with the convolution results, as described using (6) [16]. The characteristics of the resulting image is determined by the kernel composition used in the filter.

$$I'(u,v) = \sum_{(i,j)\in R_H} I(u+i, v+j)H(i,j) \tag{6}$$

I = image function
H = convolution kernel

Smoothing filters are simple yet powerful techniques used for noise removal. The Gaussian filter is effective at smoothing images with Gaussian distributed noise and is based on the general form of Equation 7 [12] [13]. The effects of this filter can be observed in Figure 27.

$$g(x,y) = \frac{1}{2\pi\sigma^2} e^{\left(-\frac{x^2+y^2}{2\sigma^2}\right)} \tag{7}$$

x = x pixel
y = y pixel
σ = sample variance

This filter acts as a low pass filter by averaging the neighboring pixels of each origin pixel while applying a Gaussian distributed weighting to the kernel. This process is demonstrated in Figure 27, where the image on the left exhibits random pixel noise which is significantly reduced after the application of the smoothing filter as shown in the image on the right.

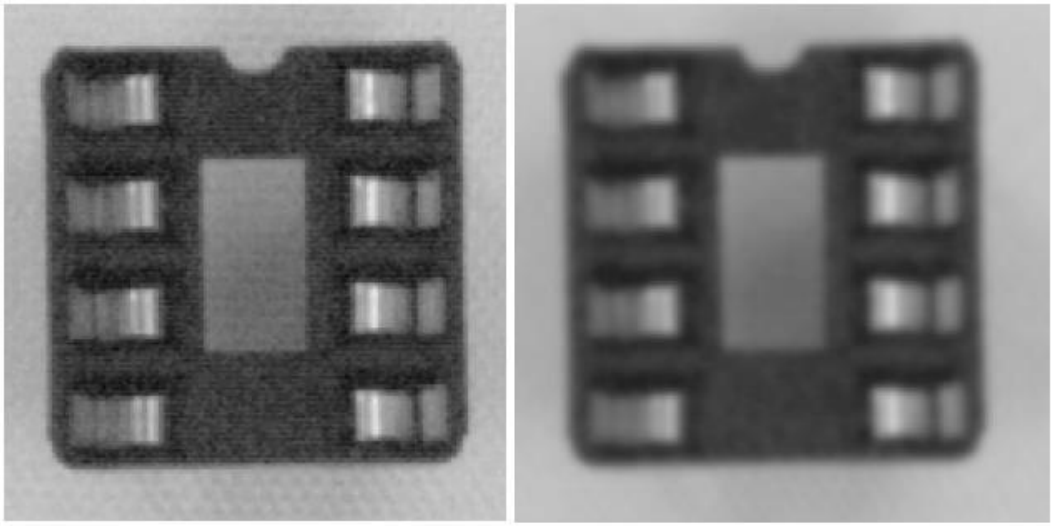

Figure 27: 5x5 Gaussian filtering example

Another useful technique is the identification of edges. Edges are a highly useful feature as they represent boundaries between objects or between regions within a single object. One method of determining an edge location is through analyzing the local rate of change between pixels. Equation (8) is the general expression for the rate of change between pixels, where f(x) is the intensity value of a pixel at the x-index [16].

$$\frac{df}{dx}(x) \approx \frac{f(x+1) - f(x-1)}{(x+1) - (x-1)} = \frac{f(x+1) - f(x-1)}{2} \tag{8}$$

This equation can be expanded to find the gradient of an image by taking the partial derivative with respect to each pixel axis as shown in Equation (9). A simple kernel to implement this function is the Sobel operator shown in Equations (10) and (11) where I is the original 2D image matrix. The functions S_x and S_y return the gradient values of the x-

axis and y-axis, respectively, where the gradient magnitude is calculated according to Equation (12) [14] [16].

$$\nabla I(x,y) = \begin{bmatrix} \frac{\partial I}{\partial x}(x,y) \\ \frac{\partial I}{\partial y}(x,y) \end{bmatrix} \tag{9}$$

$I =$ image function
$x =$ x-axis pixel
$y =$ y-axis pixel

$$S_x = \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix} * I \tag{10}$$

$S_x =$ x-axis Sobel operator
$I =$ image function

$$S_y = \begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix} * I \tag{11}$$

$$|\nabla I(x,y)| = \sqrt{S_x(x,y)^2 + S_y(x,y)^2} \tag{12}$$

Another area of image processing utilized was image compression, which is the resampling of an image to reduce the size of its representation. In many practical applications, an image contains more data than is needed which can result in unnecessary increases in processing time. Compressing the image can often remove much of this redundant information while maintaining the overall global image features desired. A basic form that was implemented was through interlacing, where alternating rows and columns are removed from the image. Another approach that is sometimes more effective

is to use Gaussian pyramids, which do not completely eliminate the information in the removed indices. This is accomplished by using a local Gaussian operator to determine the subsampled row and column values based on a pixels neighboring spatial information [12] [15] [13].

In many instances, after an image has been filtered, the specific pixel intensity values are no longer useful for providing additional information for a task such as edge detection for example. The location of the edge is the desired information while the intensity value of the edge is not directly useful. In these instances, image segmentation is used to establish boundary limits within an image, usually between the background and foreground, or to identify features in an area of interest. Once these boundaries are determined, the image can be reduced to binary values which correspond to background/foreground. This is accomplished through thresholding as shown in Equation 13, where I is the image function of indices x and y, and T is a calculated threshold value used to determine whether or not a pixel is of interest [13].

$$I(x,y) = \begin{cases} 0: \text{if } I(x,y) < T \\ 1: \text{otherwise} \end{cases} \tag{13}$$

Once an image is segmented, the connected regions may sometimes be referred to as particles. Further filtering techniques may be applied and are often referred to as particle filters, or binary filters [14]. Particle filters first analyze a segmented image and return characteristics of the connected regions which is then accepted or rejected based on a threshold for the returned analysis results. One useful analytical measurement used for

particle filtering is the Heywood circularity factor which represents the ratio of the perimeter of an enclosed region to the perimeter of a circle having the same area. This value can be determined using (14) and is effective for quantifying a shapes similarity to a circle [17].

$$circularity\ factor = \frac{p}{2\sqrt{\pi A}} \tag{14}$$

$p =$ particle perimeter
$A =$ particle area

4.2 Camera Calibration

In this research, it is necessary for the analytical results to be in world units, such as millimeters, which must be relationally defined. To translate the results accurately, the camera must be calibrated, which is the process of normalizing images into a geometrically accurate representation [13].

Ideally, more light received by a camera sensor results in more information about the scene. However, it is important that this light from one point in the scene is transferred to only one sensor element. A simple model can be represented through the concept of a pin-hole camera. As shown in Figure 28, the points of an image scene are linearly mapped onto the image plane according to a scaling factor determined by the focal length [16].

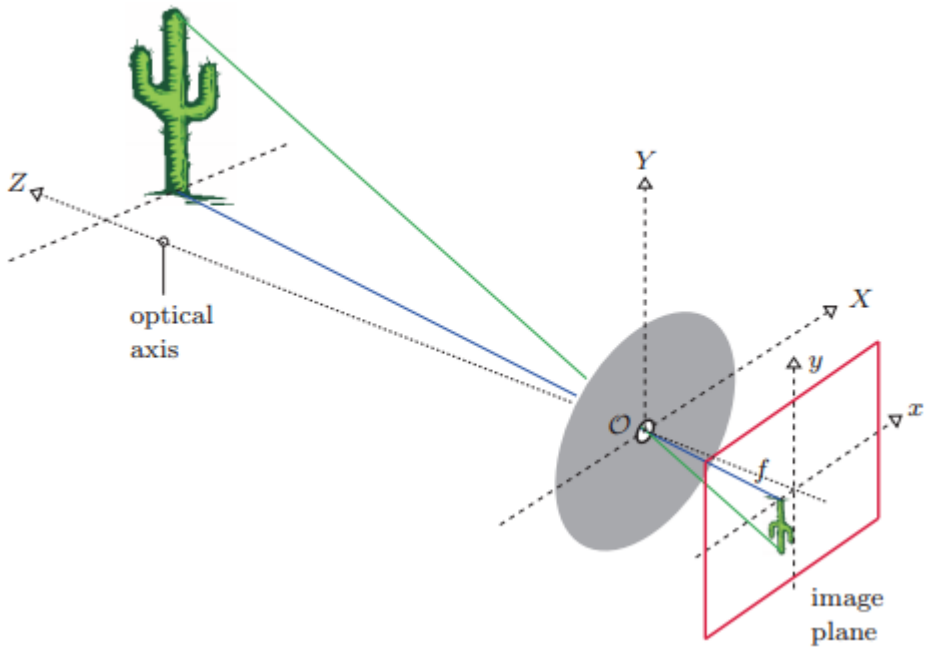

Figure 28: Pinhole camera model [16]

If all of the light passes through a single projection point, then an angular relationship forms between the scene and the sensor, so that only light from a single point is transferred to a single point on the sensor as described by the following equations [16]:

$$x = -f\frac{X}{Z} \tag{15}$$

x = image x-coordinate
f = image focal length
X = world x-coordinate
Z = optical axis

$$y = -f\frac{Y}{Z} \tag{16}$$

y = image y-coordinate
f = image focal length
Y = world y-coordinate
Z = optical axis

The pinhole camera model limits the amount of light available, which means reduced information content. This can be resolved by the use of a lens, which can focus a larger area of light to a single location, as shown in Figure 29 [12].

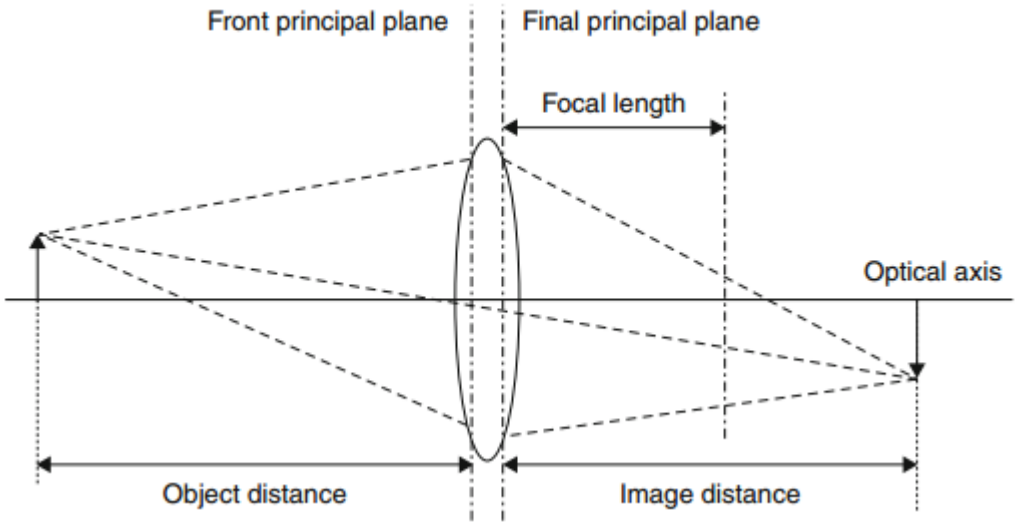

Figure 29: Lens description [12]

When lenses are introduced to a vision system, a variety of new error sources are also introduced. Therefore, the primary sources of error must be identified, modeled, and rectified to achieve geometrically accurate measurements. Some of the primary sources of error are [13]:

- World coordinate offset
- Central projection error
- Lens distortion
- Mapping sensor coordinates into image coordinates

To correct for these error sources, assume a single point $P_w = (X_w, Y_w, Z_w)$ in world coordinates can be described as camera coordinates $P_s = (X_s, Y_s, Z_s)$ by the affine transformation of the following equation [13]:

$$(X_s, Y_s, Z_s)^T = \mathbb{R} \cdot [(X_w, Y_w, Z_w)^T + \mathbb{T}] = \begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{bmatrix} \cdot \begin{bmatrix} X_w + t_1 \\ Y_w + t_2 \\ Z_w + t_3 \end{bmatrix} \quad (17)$$

$(X_s, Y_s, Z_s) =$ camera coordinates
$(X_w, Y_w, Z_w) =$ world coordinates
$\mathbb{R} =$ rotation matrix
$\mathbb{T} =$translation vector
$r =$ rotation coefficients
$t =$ translation coefficients

This can be further expressed as Equation 18 to account for error resulting from an offset image center [13].

$$\begin{bmatrix} x - c_x \\ y - c_y \\ f \end{bmatrix} = \begin{bmatrix} x_u \\ y_u \\ f \end{bmatrix} = f \begin{bmatrix} X_s/Z_s \\ Y_s/Z_s \\ 1 \end{bmatrix}$$

$$= f \begin{bmatrix} \dfrac{r_{11}(X_w + t_1) + r_{12}(Y_w + t_2) + r_{13}(Z_w + t_3)}{r_{31}(X_w + t_1) + r_{32}(Y_w + t_2) + r_{33}(Z_w + t_3)} \\ \dfrac{r_{21}(X_w + t_1) + r_{22}(Y_w + t_2) + r_{23}(Z_w + t_3)}{r_{31}(X_w + t_1) + r_{32}(Y_w + t_2) + r_{33}(Z_w + t_3)} \\ 1 \end{bmatrix} \quad (18)$$

If a distorted image point is given by Equation 19 which represents an undistorted image point of Equation 20, then Equations 21 and 22 may be used to represent the transformation of a distorted point to a calibrated undistorted point. Furthermore, for a principal point (c_x, $c_{y)}$, the magnitude of the radial distortion can be expressed as shown in Equation (23) [13, 12].

$$p_d = (x_d, y_d) \tag{19}$$

$$p_u = (x_u, y_u) \tag{20}$$

$$x_u = c_x + (x_d - c_x)(1 + k_1 r_d^2 + k_2 r_d^2) \tag{21}$$

x_u = undistorted coordinate
x_d = distorted coordinate
c_x = x-axis principal point
k = distortion coefficient

$$y_u = c_y + (y_d - c_y)(1 + k_1 r_d^2 + k_2 r_d^2) \tag{22}$$

y_u = undistorted coordinate
y_d = distorted coordinate
c_y = y-axis principal point
k = distortion coefficient

$$r_d = \sqrt{(x_d - c_x)^2 + (y_d - c_y)^2} \tag{23}$$

Equations (21) through (23) can readily be used to solve for the distortion coefficients using a reference point in world coordinates, its undistorted transformation to camera coordinates, and the resulting distorted location. The only input that must be provided is the reference point location within the world coordinate system. One method to accomplish this is through the creation of a grid system with known spacing. The world coordinate system origin can arbitrarily be set to zero during this process, then scaled later on. Once the reference points are located in an image, Equations (19)-(23) can be used to solve for the distortion modeling coefficients. These equations can then be used

to solve for an error map across the entire image stored in a look up table to decrease computation times during a live application.

In this research, the dot grid method was selected in order to take advantage of the LabVIEW vision libraries to handle the computation of the distortion model. An example the grid is shown in Figure 30. The determination of the reference positions using this method is based on the relative separation distances, and not an absolute known locations; careful procedures were followed to ensure a flat surface with even spacing to prevent additional model distortion. The dot grids were computer generated, then printed at 1200 dots per inch (dpi) to minimize resolution based errors. To reduce any perspective error caused by surface contours, a 1/8 in. steel plate was selected to provide a rigid surface. A thin layer of spray adhesive was used to secure the paper to the steel, which minimized any rippling of the paper.

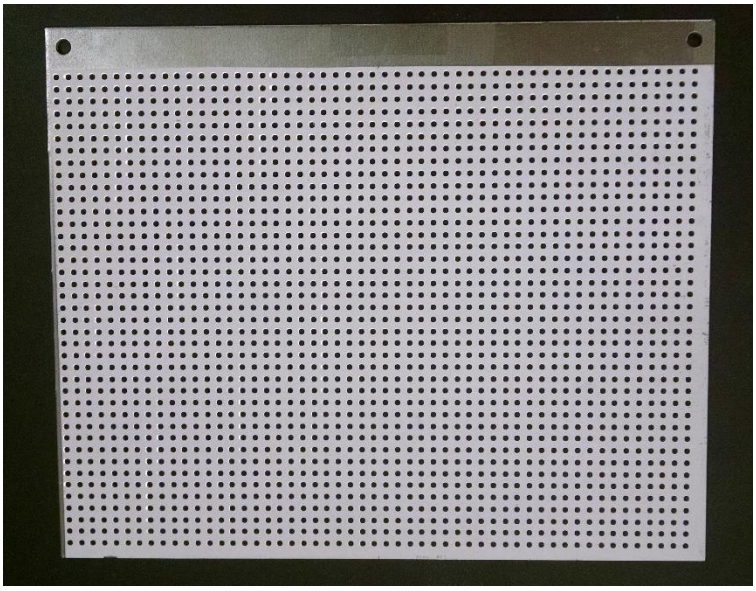

Figure 30: Dot grid used for camera calibrations

When performing the calibration, the camera was located in the position to be calibrated and the calibration grid was placed flat along the surface of the area to be mapped. Due to the affine transformation used by the calibration model, the results are only valid for real-world coordinates coplanar to the calibration grid reference plane, and images must be taken from the same camera pose. Several images were taken with the grid at various angles, which produced the reference coordinates used to determine the cameras intrinsic parameters. A local threshold was applied to segment the image using dots as the foreground and all other pixels as background. Noise frequently remained after this process which was subsequently eliminated through further particle filtering. Based on the average pixel area of each dot, an area filter was applied to remove any particles significantly smaller. Next a Haywood circularity filter was applied to remove any remaining noise that did not meet the threshold levels of edge circularity. The centroids of the resulting particles were calculated and used as the final reference point origins.

The lenses used with the cameras have variable focal points and apertures. Setting the aperture too low reduced the amount of light available, which decreased the image contrast, while setting it too high allowed too much light which can lead to saturation. Since lighting was controllable, it could be reduced to prevent light saturation, but doing so would degrade the SNR at the cost of image resolution. The ideal configuration would be a minimum aperture size with maximum light to increase the contrast. This is not easily implemented in practice, as it can cause specular reflections and can result in prominent shadowing, especially if the lighting is non-uniform. To find a balance sufficient enough for this research, the aperture was configured to its median setting and

additional lighting was supplemented as needed. To establish appropriate focal length setting, a live feed of the camera was monitored as the focus was adjusted until the image was visibly at its highest quality. Once these settings were found, set screws were tightened to prevent any further changes, as the calibration model is dependent on these specific parameters.

4.3 Cross Correlation

A common technique used to assess similarity between two signals is cross correlation. This method involves convolution of a search template through an image to produce correlation scores at each image pixel. The 2D formula for image correlation is shown in Equation (24) [17].

$$C(i,j) = \sum_{x=0}^{L-1} \sum_{y=0}^{K-1} w(x,y) f(x+i, y+j) \tag{24}$$

$C =$ correlation function
$f =$ image function
$w =$ matching template function
$i =$ pixel row index
$j =$ pixel column index
$L =$ template image width
$K =$ template image height
$x =$ template x index
$y =$ template y index

While correlation can reliably find a template within an images that share similar characteristics, the processing time is very high and rapidly increases with image size. Additionally, this methods effectiveness is highly sensitive to scale and rotation variations. Due to these shortcomings, more advanced methods were investigated.

4.4 Low Discrepancy Sampling

One method that can dramatically reduce the computation time of cross correlation is through the use of low discrepancy sequences (LDS). Sometimes referred to as quasi-random sequences, these patterns represent a continuous distribution of quasi-random numbers in a data set that has more homogeneous characteristics than random or pseudorandom distributions, especially with smaller sample sizes [18] [20]. Traditional correlation techniques utilize all of the information in an image during the matching process. However, it is often that all of the information is not necessary or useful resulting in unnecessary computations and time delays. LDS uses non-uniform sampling to extract only the dominant features of an image [18] [14] [20].

In numerical integration, an area of interest is typically reduced to a finite number of sample points which are used to evaluate a given function [19]. The number of samples required is dependent upon the nature of the analysis; for images, this number can easily reach numbers in the order of 10^6. The most basic approach is to use a random number generator to determine the sample locations, as shown in (Figure 31). However, this approach is not without disadvantages, including varying density through “clumping” [19] [20]. A second disadvantage results from non-deterministic characteristics. The two sample groups (left and right) were generated using the same parameters but resulted in a different set of sample locations. The same parameters were used to generate each plot, but resulted in different sample locations.

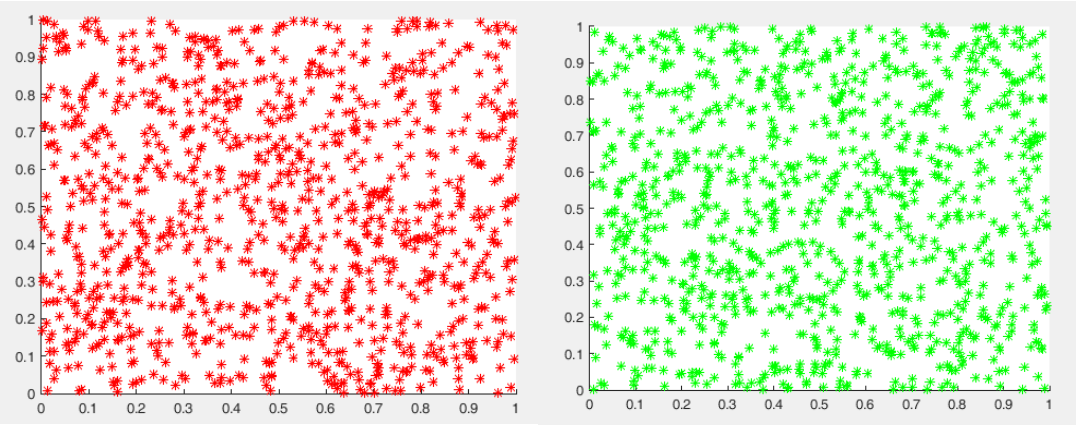

Figure 31: Spatial sampling comparison using pseudo-random number generator

Discrepancy is the terminology used to quantify the measure of uniformity in a sequence [20]. More specifically, it is the error produced by the difference in percentages of total volumetric sample points between subsets. In general, Equation (25), sometimes referred to as the star-discrepancy, can be used as a criteria to determine if a sequence can be described as low discrepancy [20].

$$D_N^* \approx \frac{(\log N)^s}{N} \tag{25}$$

$N =$ number of samples
$s =$ dimension order

Several types of sequences are frequently used which produce low discrepancy characteristics. These are often referred to as quasi-random sequences because they exhibit uniform distribution similar to random sequences, but they also produce characteristics of a lattice distribution [20]. One method used to generate such a distribution is the Halton sequence, which relies upon the p-adic expansion of integers

from a pseudo-random generator [20]. Equation (26) demonstrates a 1D uniformly distributed sequence where $(n)_p = a_k a_{k-1} \cdots a_0$ and $0 \leq a_i \leq p$ [20].

$$S_p(n) = \frac{a_0}{p} + \frac{a_1}{p^2} + \cdots + \frac{a_k}{p^{k+1}} \tag{26}$$

It follows that any s-dimensional application can be generalized by the following [20]:

$$\boldsymbol{x}_n = \left(S_{p_1}(n), \ldots, S_{p_s}(n)\right) \tag{27}$$

An example of a resulting sample distribution is shown in Figure 32, which is the first 512 elements of a Halton sequence. When compared to the random number generation method, it has notably more homogeneous characteristics and has a repeatable distribution pattern. An additional benefit is the capability to implement the sequence using additive recursion which allows samples to be incrementally added and can reduce the required number of samples evaluated before a match occurs [20].

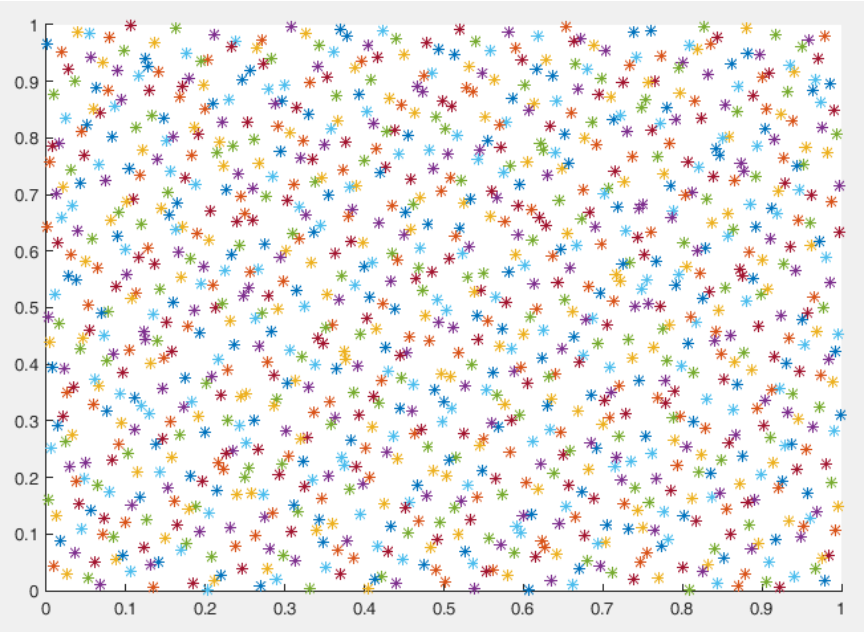

Figure 32: Low discrepancy sampling visualization using Halton sequence

The LabVIEW Vision Development Module has native libraries which allowed for rapid implementation of these techniques. To create a template image to use for matching in a search image, a reference image containing the desired object is acquired (Figure 33 top), preferably under consistent conditions as those during the search process. Using an image mask, the object can be sub-sampled and stored as a separate image. Further masking can be applied to prevent features from being used in the calculation of the correlation score (Figure 33 bottom left), which is especially useful if certain features exhibit high variability between search images. When a match occurs, it is often useful to identify an offset location to be used as the origin of the template, which is shown in Figure 33 (bottom right). Once all of the initial setup parameters are chosen, LabVIEW contains a library which initializes the low discrepancy sampling method and stores all of the template information as part of the image data.

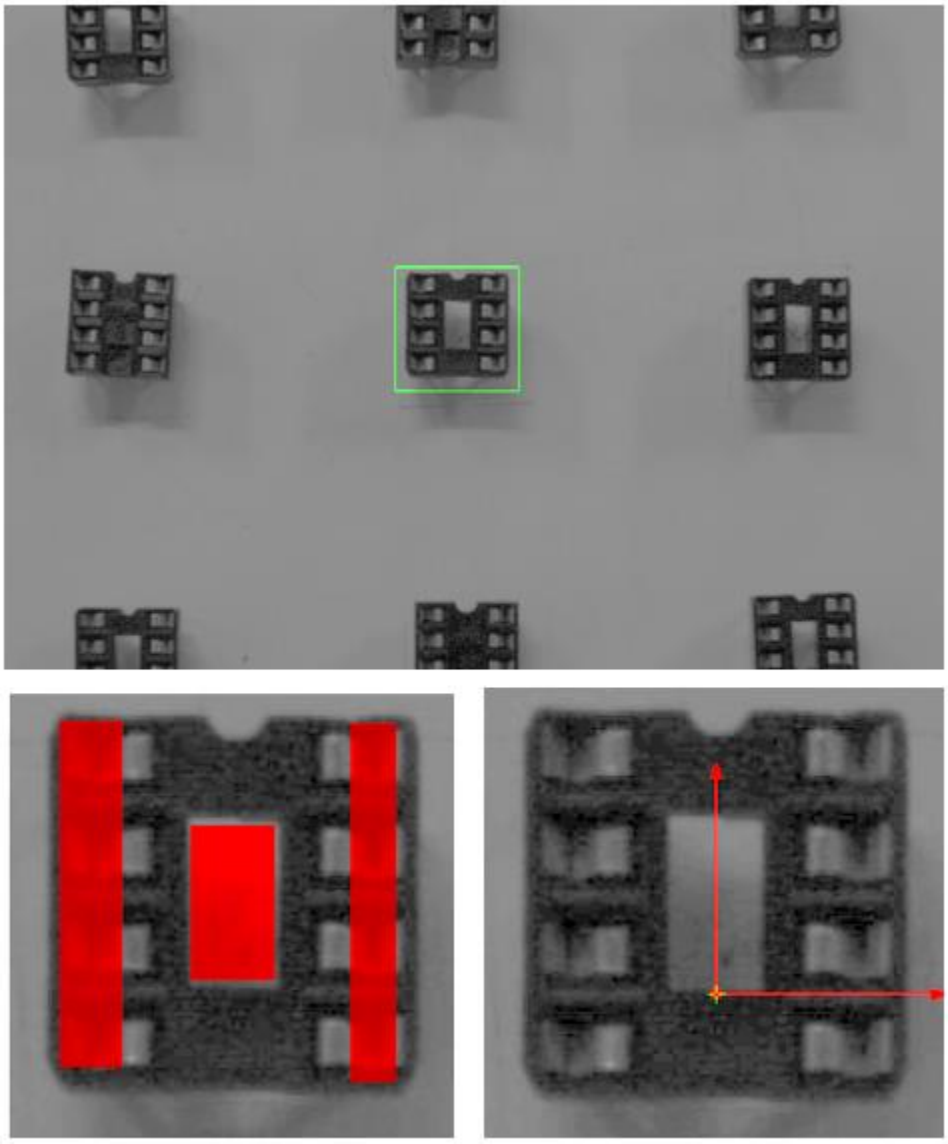

Figure 33: Object reference image (top); correlation mask (bottom left); match origin offset (bottom right)

In the matching process, convolution is used to produce correlation scores across the search image. Each resulting score is produced from the sub-sampled points produced by the low discrepancy series to reduce the processing time. The desired angular search range can be adjusted from 0-360°, and the incremental angular accuracy can be precisely

tuned as well. Rotated patterns can be searched for as well according to the desired angular range. This range can be limited, and the intermediate angular search angles can be adjusted to decrease search time. A correlation score from 0 through 1000 is computationally produced which is compared to a threshold to determine a successful match. In general, setting a threshold of 800 or higher produces a high confidence of a successful match. Once an approximate match occurs, interpolation is performed to achieve resolution potentially finer than a single pixel unit.

One primary advantage of LDS is its computational efficiency, requiring a relatively low number of samples to for accurate estimation of object location. This significantly reduces the computational complexity of a total cross correlation method, which exponentially increases with larger images. One potential disadvantage can arise when the template image is relatively small or has dense features intervals which can be excluded due to the small sampling numbers [14, 18].

4.5 Pyramid Template Matching

The grayscale value pyramid technique (GVP) uses the correlation score of a pre-existing image template model within the image of interest. Specifically, a template image is processed by down sampling the image at various pyramid levels. At each level, the features used for cross correlation are the normalized gray values of the image. During the matching process, both the image and template image are sampled to the highest available pyramid level, drastically reducing the computational complexity. A score is produced at each location in the search image and a threshold is applied to determine if an acceptable match has occurred. This is done at each pyramid level until a match occurs. When a match is found in a higher pyramid level, the search area in the

successively lower pyramids is approximately known, further reducing the complexity until the match is located in the full image size. This method is particularly useful for templates with complex textures and compact edges [18].

Another implementation of this pyramid matching is through gradient pyramid matching. The concept is similar, but instead of using each grayscale value, the image is first filtered to locate significant edges, while suppressing weak edges [14]. Much of the unnecessary information is removed in the resulting binary image which can decrease the execution time. However, if the image has a high feature density, the filtering process may consume more time than it saves. Additionally, this method can become less reliable in templates with minimal edge features [18].

CHAPTER 5: AUTOMATED WORK CELL METHODOLOGY

In order to evaluate each component of the research, an experimental procedure was designed and implemented to replicate the assembly process. This procedure was used to control the execution of various tasks and collect data for later analysis regarding certain process variables. This section will outline the development of both the procedure as well as the selection and measurement of process variables. Figure 34 and Figure 35 depict the primary system components of the work cell.

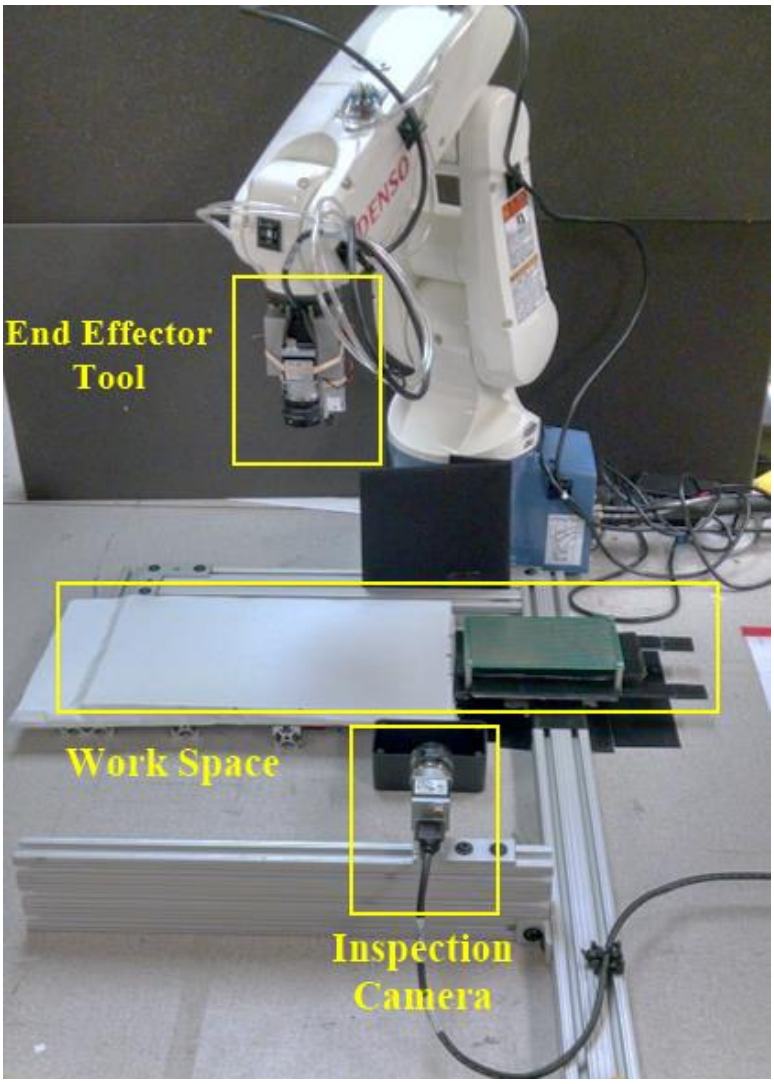

Figure 34: System component overview

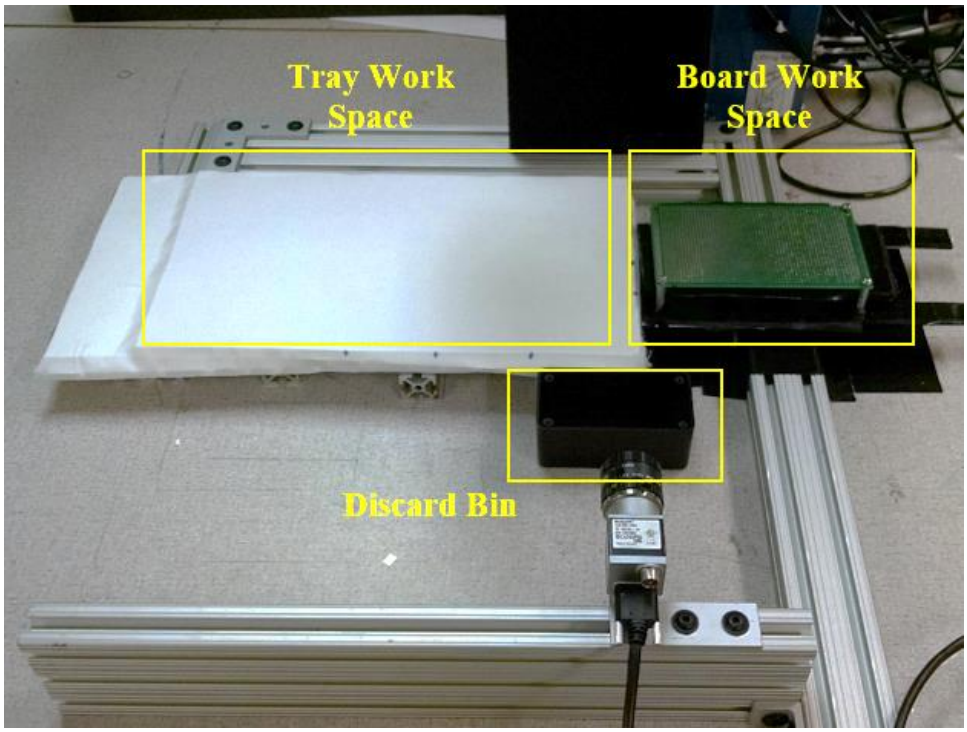

Figure 35: System work space overview

The individual steps in the testing process begin with calibrations. The end effector and camera calibration procedures previously described are only necessary to perform initially if a change in any of the hardware has occurred, such as changing gripper fingers or camera parameters. After the calibration procedures are complete the assembly project parameters are input into the experimental testing program. These parameters include the following:

- Image search templates
    - Component
    - Component pins
    - Circuit board
- Inspection parameters
    - Pin spacing tolerance range
    - Pin angle tolerance range
    - Pin quantity
    - Expected component quantity
    - Expected placement quantity
- Motion parameters
    - Inspection positions
    - Component pick clearance height
    - Placement clearance height

Once the setup is complete for an assembly project, a tray of the selected components and their corresponding target circuit board are placed in their respective workspaces. The automated process is then executed. The process used to conduct each experiment generally shared common task elements, which begin with the localization of the components and board placements within their work areas. This is displayed in Figure 36 where the robot moves to a predetermined position to acquire each image and processes the location and orientation data.

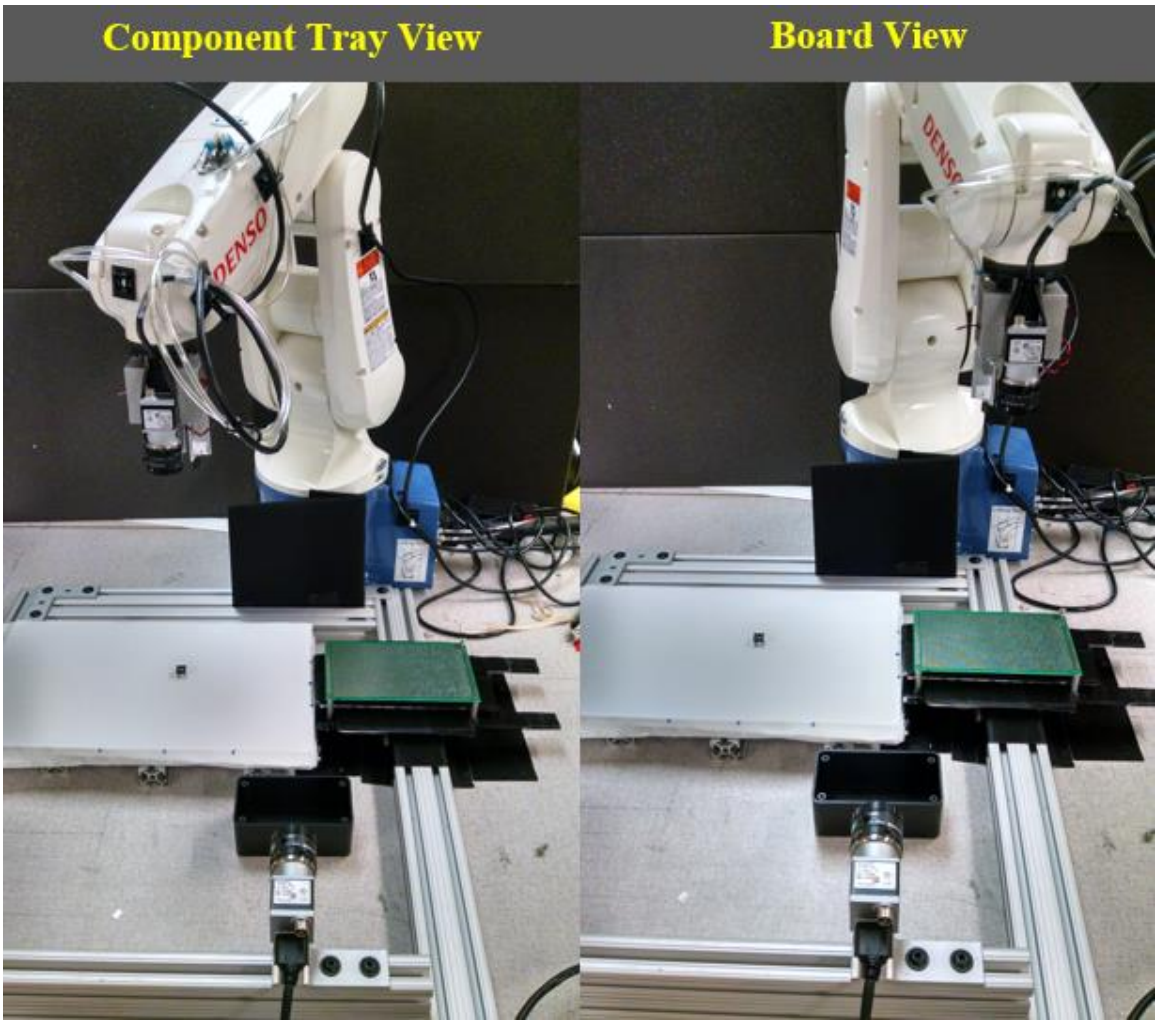

Figure 36: Robot positions for component/board localization

The selection of locations for the tray and board work areas were specifically chosen to minimize unnecessary movements of the end effector. Another limiting factor that was considered for the selection of these locations was the cameras field of view (FOV). The FOV increased as the cameras height increased, which allowed a greater number of components to be placed per iteration. This is beneficial because of the additional time duration required to capture and process the image of a new component/board set.

However, a limited space was accessible by the arms range of motion and increasing the FOV beyond this point did not provide any additional benefits. The same condition applied for the location of the board. However, since the camera lens has a fixed focal length and depth of field, both locations needed to be considered when selecting a FOV which was limited by the larger of the two areas.

In a typical assembly line, the number of an individual component on a given board varies widely based on the nature of the product requirements. A single board may use an individual component type only once, or it could have multiple placements. For this reason, tests were conducted where only one component placement was specified per board as well as multiple placement specifications per board. For automated processes such as this, the boards are typically placed on conveyors or other types of automated machines that maintain a constant flow of parts. The development or procurement of such systems was outside the scope of this research. Instead, manually placing the boards and component sets conceptually represented same action as if a machine had performed these tasks.

Once the component localization is complete, the program positions the tool at the calculated coordinates and closes the gripper (Figure 37). The orientation angle for the tool to grab the component is defined in the matching template coordinate system. The program executes a short movement upwards along the z-axis to prevent any pin deformation from contact with the tray surface.

Figure 37: Picking a component after localization

The component is translated to the inspection viewing position where it is analyzed in different poses for precision pin localization and tolerance (Figure 38). If it is determined the component is unfit for placement, it is discarded in the discard bin and the process iteration is terminated.

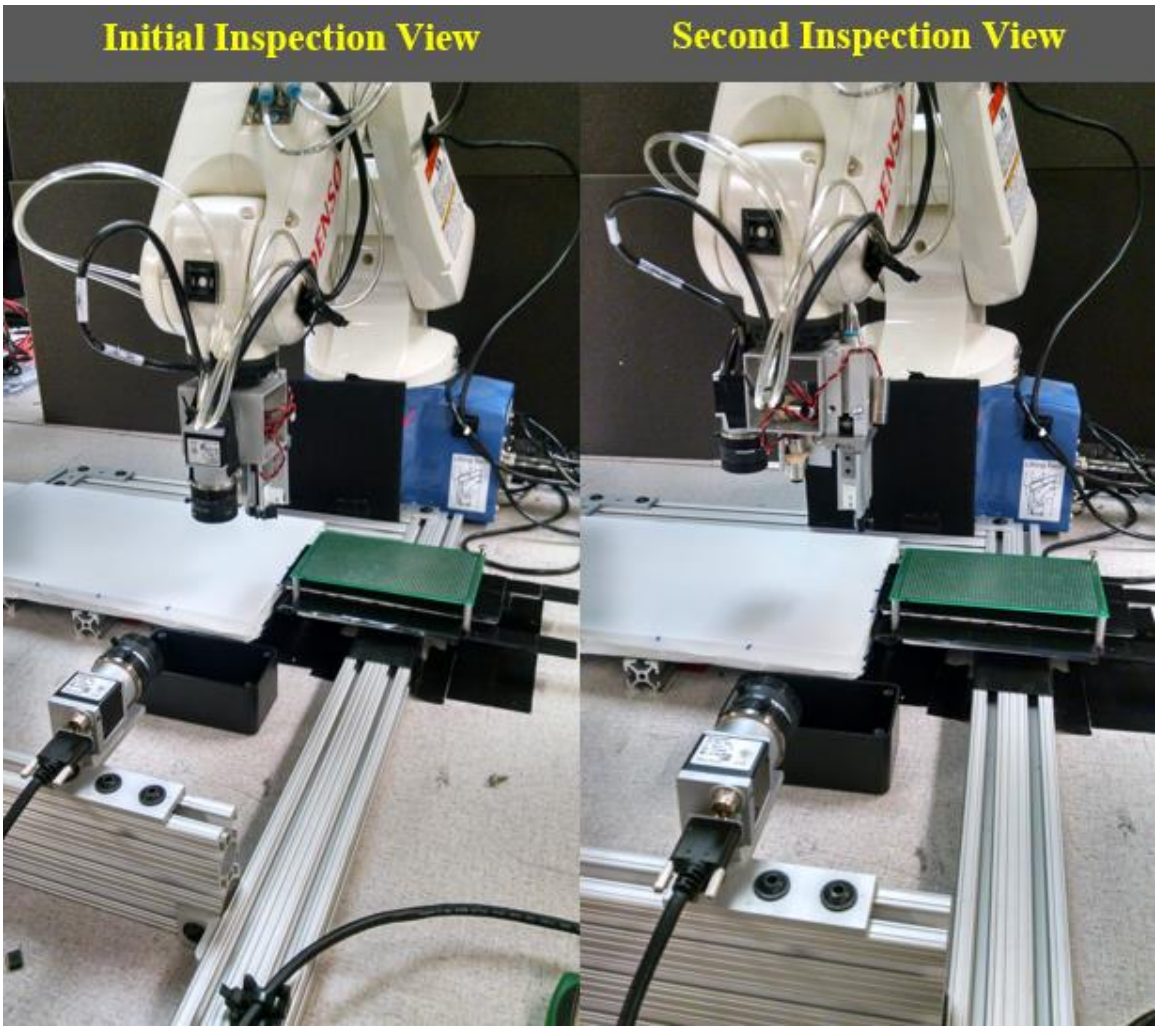

Figure 38: Component inspection positions

To determine the pin spacing, an image matching template is used to return the location and orientation of each pin (Figure 39). The template origin corresponds to the center of each pin tip where the difference between matches is used to determine separation distance. The measured location of a single pin with respect to the tool coordinates was used as the reference for the board placement. In all cases, the top left pin was used as the reference, which is consistent with the pin numbering convention of

many THT components. The component is then inserted into the circuit board to complete a single component placement cycle as shown in Figure 40.

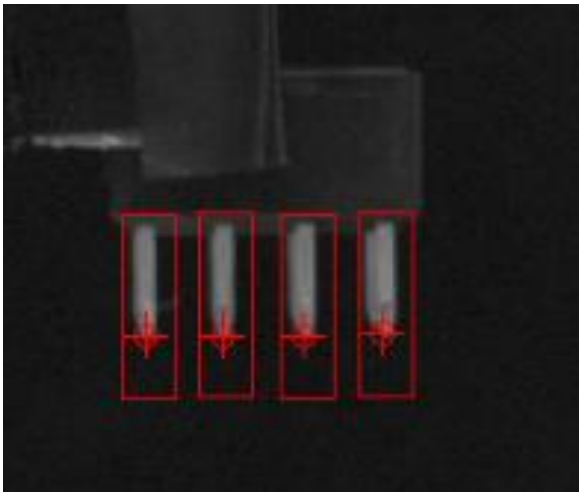

Figure 39: Pin reference matching

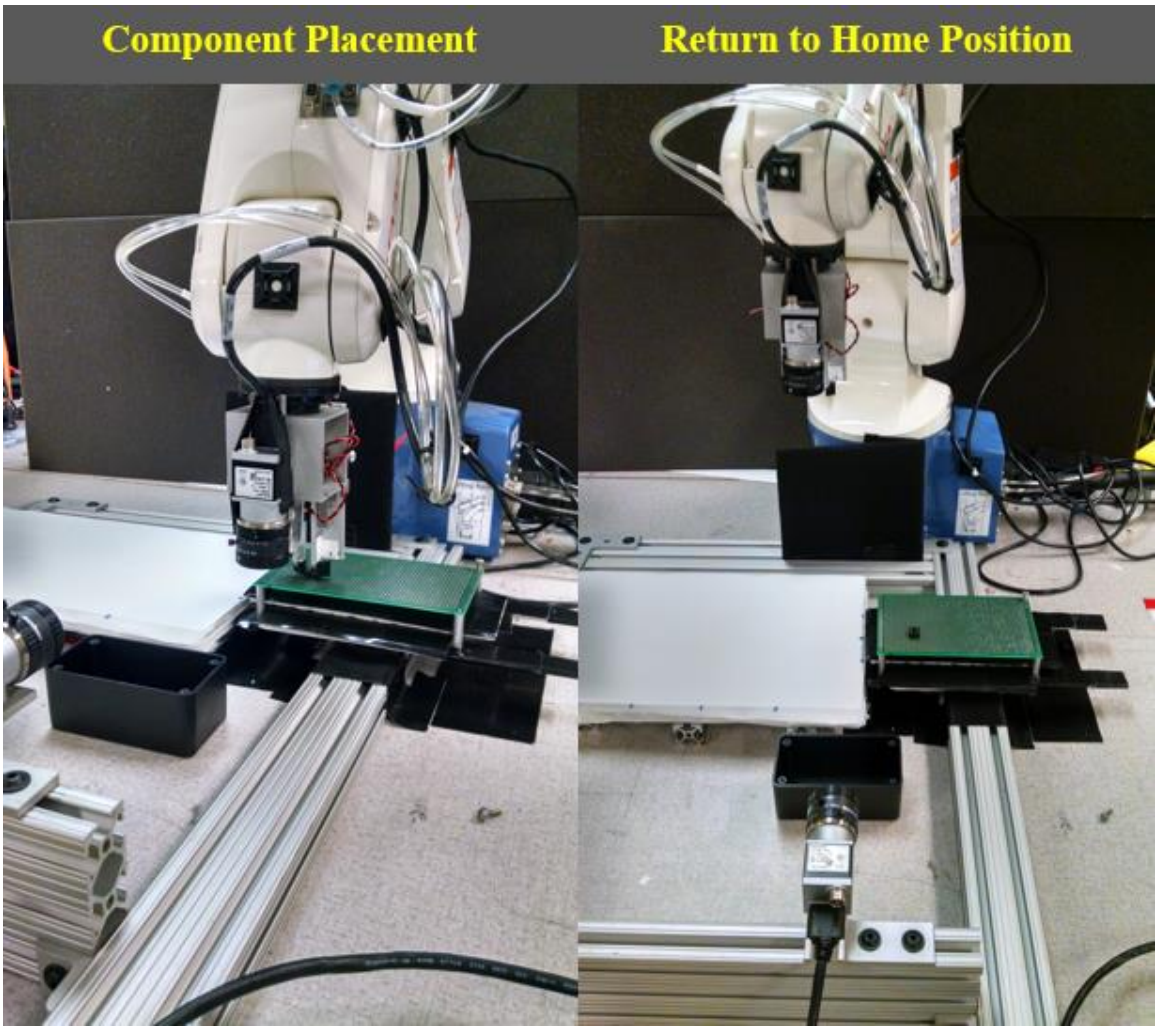

Figure 40: Component placement and process completion positions

Two primary approaches were utilized to determine the placement location for the reference pin of the component. If a board template is reasonably small, the entire board may be used to identify the placement location. Larger boards would result in searching for unnecessary information and produces higher processing durations. To reduce the processing time, unnecessary information in the template image was masked which was effective for boards requiring only a single component placement. When multiple

placements were needed for a single board, the processing time increased proportionally to the number of additional placements needed to be localized. A solution was implemented which used only 3 search templates that were made relatively small. As shown in Figure 41, these 3 templates were selected to identify coordinate positions for use in creating a board coordinate system. The distance between the reference template and the dx template was divided by the number of thru-holes to establish an array of locations corresponding to each thru-hole center. This was done similarly between the reference and dy templates. This approach created a board coordinate system that allowed any quantity of placements to be determined in approximately the same fixed time. The process of selecting placements was simplified by allowing integer values to index the desired row and column of the placement coordinate arrays.

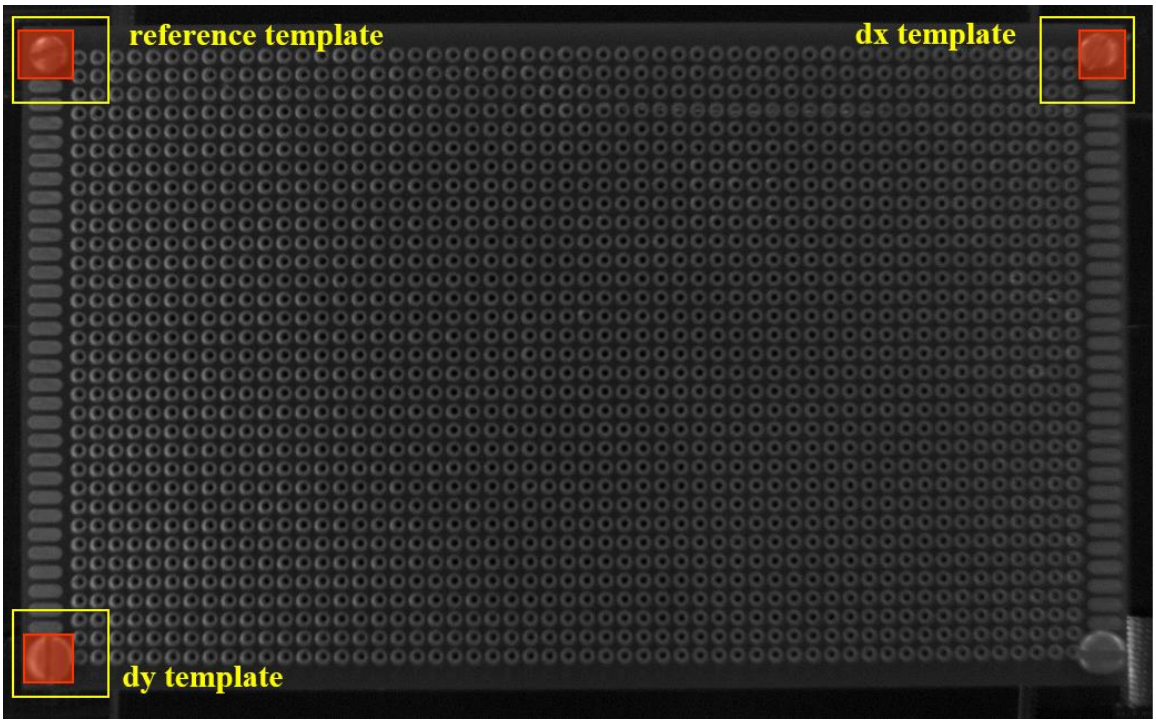

Figure 41: Board placement template overview

A summary of relevant definitions relating to the experimental testing procedure is located in Table 2. These parameters define the scope and limitations of the testing conducted, and also outline the specific performance goals and verification methods.

Table 2: Experimental testing definitions

Localization Time	Duration to localize all components on a single tray and all placements on a single board
Tray Cycle Time	Duration to pick and place all components on a tray
Component Cycle Time	Duration to pick and place a single component from a tray
Distributed Component Cycle Time	Equal distribution of tray cycle time among all components on a single tray
Process Error	An occurrence of one of the following: • Failed attempt to pick a component • Placing the component in the incorrect position, including misaligned placements and absent placements • Any action that results in damage to an electronic component, circuit board, or part of the work cell
Non-Standard Component	Non-standard, or odd components, are defined as those with the following characteristics: • Through hole placement (non SMT) • Non-conforming dimensions, form factors, or pin spacing • Are not packaged using a reel feeder • Cannot be readily used by existing dedicated systems

Performance Specifications	• Fully automated operation – no human interaction after issue of start command
	• Average error less than 1%
	  o Verified through visual inspection
	• Average component cycle time less than 5 seconds
	  o Verified through internal program timers

CHAPTER 6: EXPERIMENTAL RESULTS

This chapter contains the results of each experimental testing procedure conducted using the methodology previously outlined. Each experiment was performed using only one component type, while the quantity, location, and rotation characteristics were varied.

6.1 Coordinate System Identifications

The robot tool calibration procedure was performed once and produced the following tool offsets used in the subsequent experimentation:

Table 3: Robot tool calibration parameters

X	Y	Z	Rx	Ry	Rz
-11.45	51.7	128.5	0	0	0

6.2 Camera Calibration Results

The camera calibration procedure was performed for each camera to identify the modeling coefficients and camera projection matrices. The results are summarized below.

Table 4: Camera calibration results

Parameters		End Effector Camera	Inspection Camera
Radial Distortion Coefficients	k1	-0.103988	0.040119
	k2	0.0887567	-4.18614
	k3	0.381705	38.2996
Tangential Distortion Coefficients	k1	0.000377	-0.00253
	k2	0.001957	0.001515
Focal Length	fx (mm)	8.363	12.181
	fy (mm)	8.344	12.181
Optical Center	dx (pixel)	1270.27	686.696
	dy (pixel)	1079.21	438.546

End Effector Camera Projection Matrix:

$$\begin{bmatrix} 0.0680712 & 0.000230396 & -4.09087 \\ 0.000197397 & 0.0690134 & -2.01386 \\ -0.00000334581 & 0.00000644127 & 1.00001 \end{bmatrix}$$

Inspection Camera Projection Matrix:

$$\begin{bmatrix} -0.0731213 & -0.00147396 & 74.0162 \\ 0.000296107 & -0.0756875 & 140.32 \\ 0.00000528269 & -0.0000195153 & 1.03111 \end{bmatrix}$$

6.3 Component Matching Performance Results

An individual analysis was conducted to compare the performance of the image matching algorithms. The results are shown in Table 5 which were used to determine optimum matching approach.

Table 5: Template pattern matching performance comparison

Matching Method	Average Time (ms)	Average Correlation Score
LDS	62.1	938/1000
Grayscale Pyramid	71.9	946/1000
Gradient Pyramid	203.3	958/1000

6.4 Component 1: LM311 IC

The first component used for testing was an LM311 IC. The LM311 is available as either SMT or THT; the thru-hole model was used for testing. This component was selected for the initial testing to bridge the gap between a standard and non-standard component. This part is readily available on reels used in current assembly machines, but is also selectively available in other packing forms such as tubes or trays, which reduces a components compatibility with the high speed feeder's current systems use. This component is a DIP with 8 pins and has the specifications displayed in Table 6. Some of the challenges with this part occur when determining rotation as it has many symmetric features. The primary features useful for determining this information are the texture features printed on the surface and a small imprint keyed on one end.

Figure 42: LM311 component (top view)

Table 6: LM311 component dimensions

Length (mm)	Width (mm)	Depth (mm)	Pin Spacing (mm)	Pin Diameter (mm)
10.1	6.5	8.26	2.54	0.6

The board to be used for placement is shown below. An unpopulated prototype board was selected due to ready availability. The use of an unpopulated board rather than populated should not result in any significant effects to the results; equivalent characteristics could be achieved by using an image mask on a partially populated board. This circuit board had a nominal thru-hole pitch spacing of 2.54 mm and a lead-hole diameter of 1 mm, resulting in approximately 0.4 mm of clearance for pin placement.

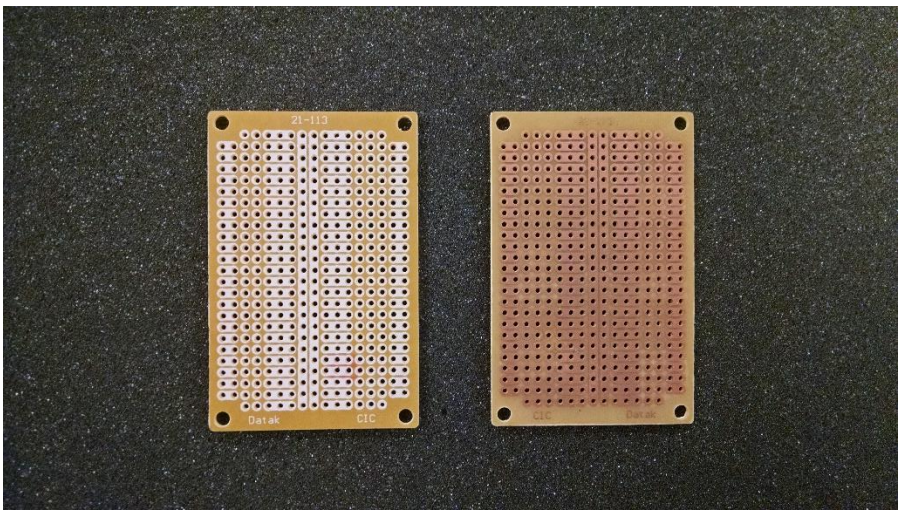

Figure 43: Unpopulated PCB (front & back)

To analyze the throughput rate and percent error, the assembly process was consecutively tested on 30 trays, each containing 3 LM311 components randomly distributed on.

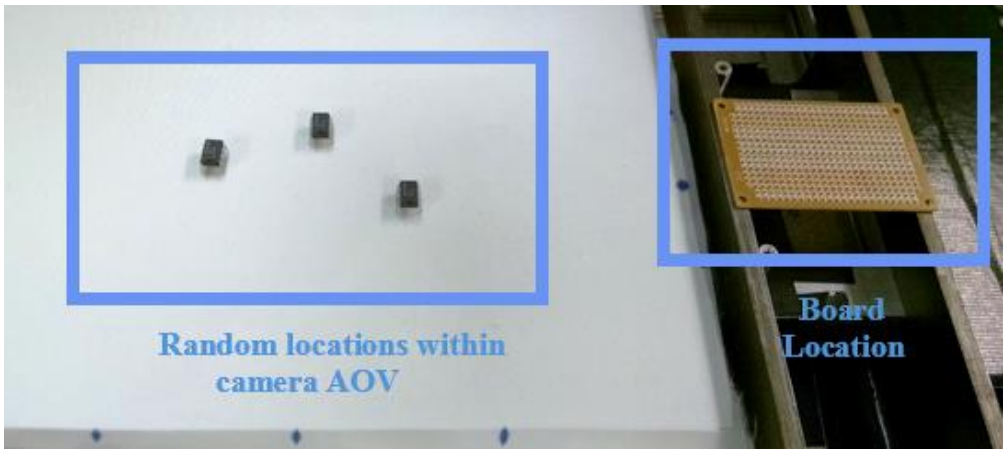

Figure 44: LM311 experimental component placement overview

The results are summarized in Table 7. The system was able to perform 30/30 placements successfully with an average placement time of 4.421 seconds and an average localization time of 3.423 seconds. Figure 45 through Figure 48 display graphs of the component cycle durations. The average calculated x-y placement locations for the ordered placements were respectively (-115.6, 309.64), (-115.5, 289.25), and (-115.4, 267.88).

Table 7: LM311 cycle time and success statistics

Component	Average, (s)	Standard Deviation, (s)	Max, (s)	Min, (s)	Successful Placements
LM311	4.42	0.13	4.64	4.19	30/30

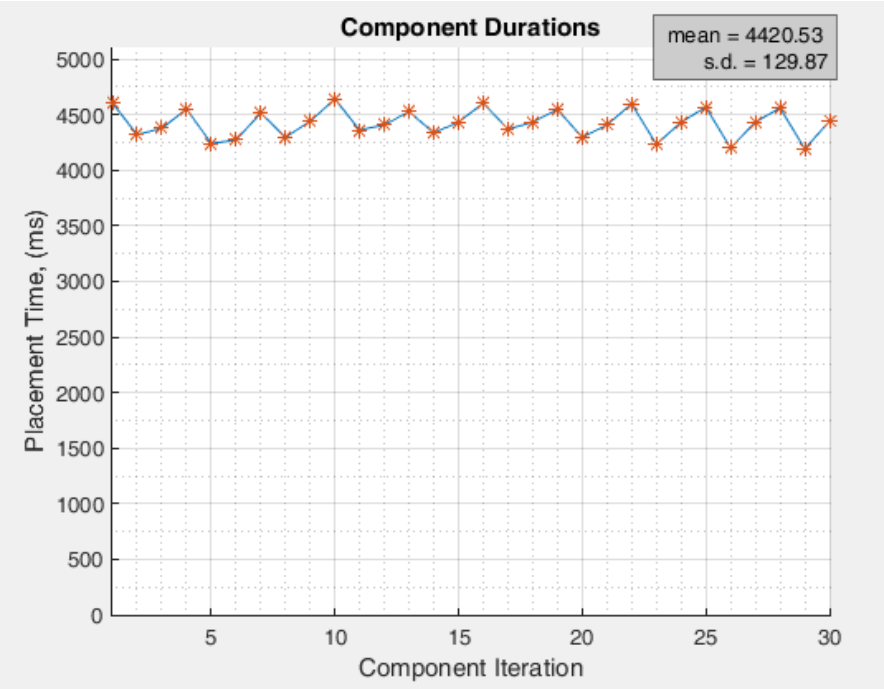

Figure 45: Combined component cycle times

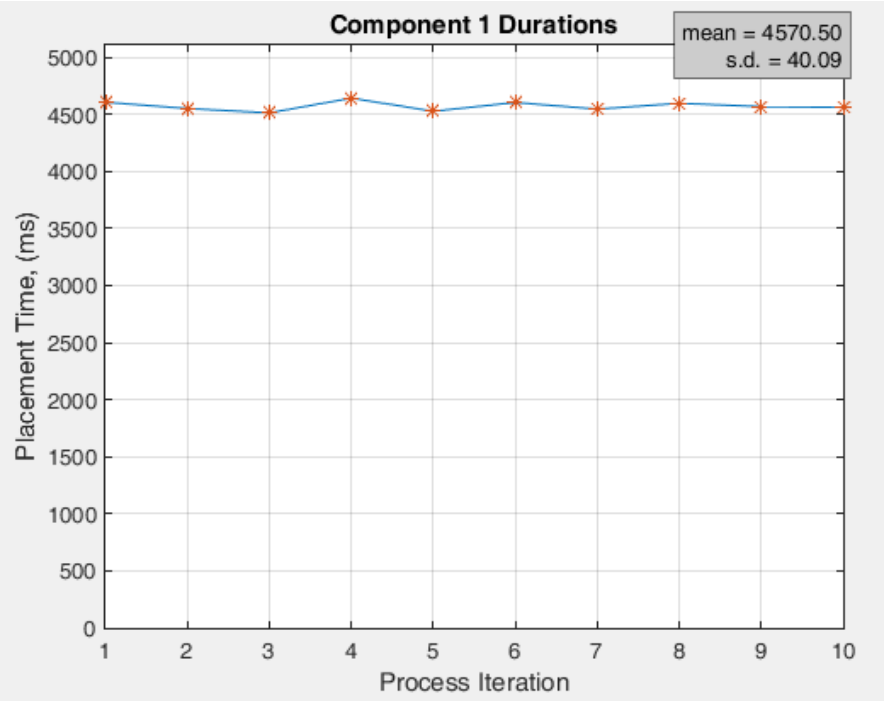

Figure 46: LM311 first component cycle time

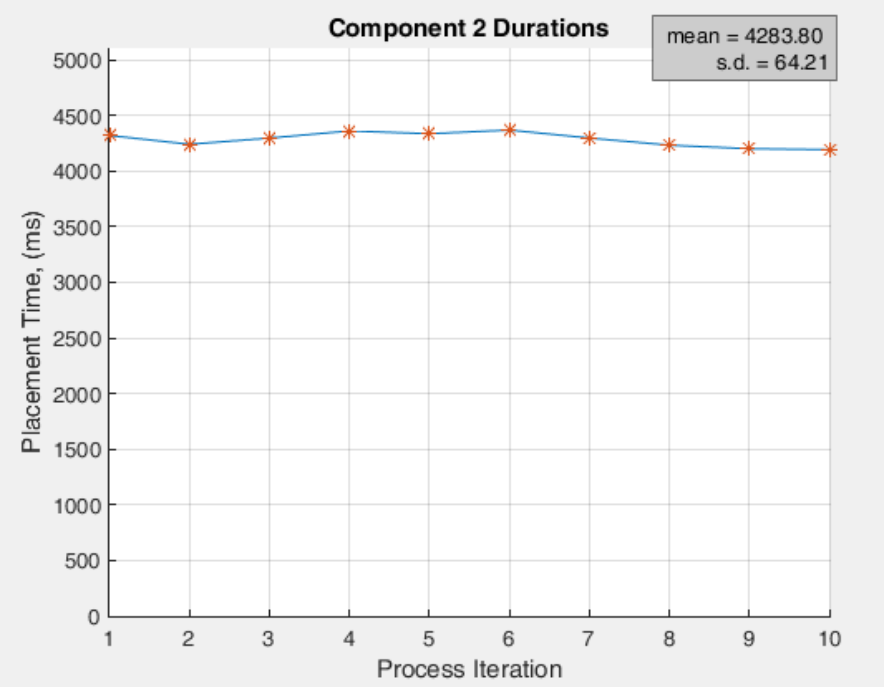

Figure 47: Lm311 second component cycle time

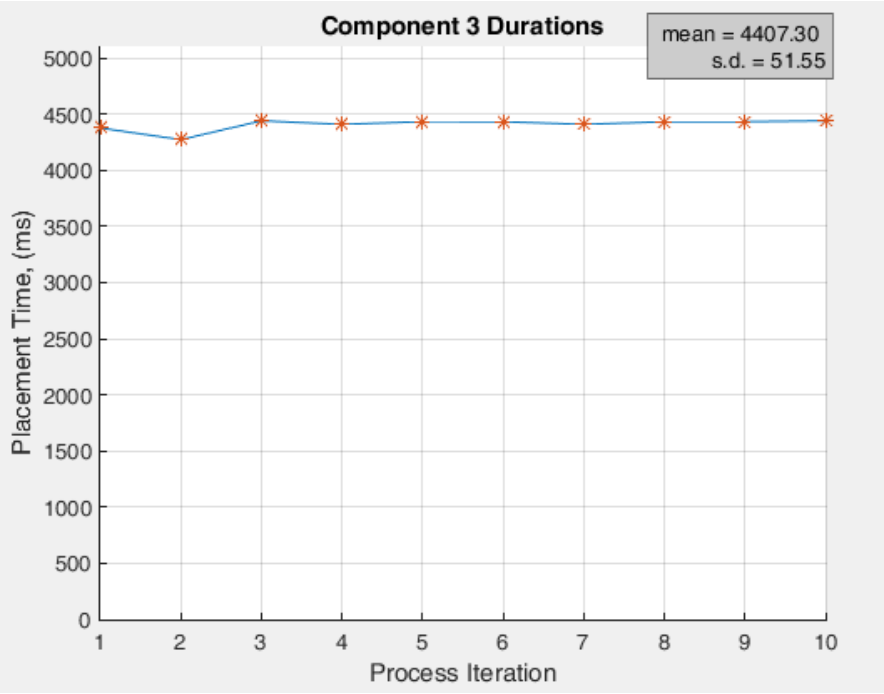

Figure 48: LM311 third component cycle time

6.5 Component 2: DIP Socket

The next component tested was an 8 pin DIP socket similar to the LM311, but has different package dimensions, and a non-uniform top surface. The package body has a much larger footprint and extends outward further than the pins. This component also had minimal features useful for determining orientation; a small indentation on one side was the only identifiable feature. Additional complications of this part were large curved specular surfaces on the top face. These features were curved in such a way that the reflections take on very different signatures dependent on the perspective position in the cameras FOV and the orientation of the component. These particular features were masked in the calculation of the correlation score.

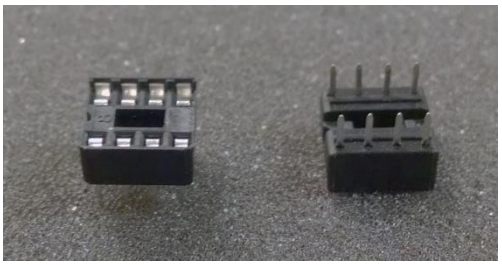

Figure 49: DIP socket component (top & bottom)

The board used for placement is shown in Figure 50. This board had a pitch of 2.54 mm and a rectangular arrangement of 54x33 lead holes each with a nominal 1.1 mm diameter. An additional feature difference from the previously tested board was double sided silver thru-hole pads as opposed to single sided copper pads which resulted in more prominent reflections. This board was also much larger, allowing a wider range of placements to be analyzed.

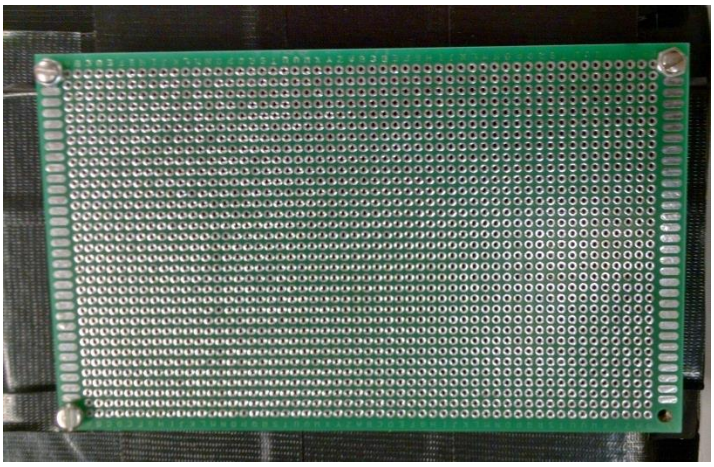

Figure 50: Unpopulated PCB used to place socket components

6.5.1 Experiment 1 Results

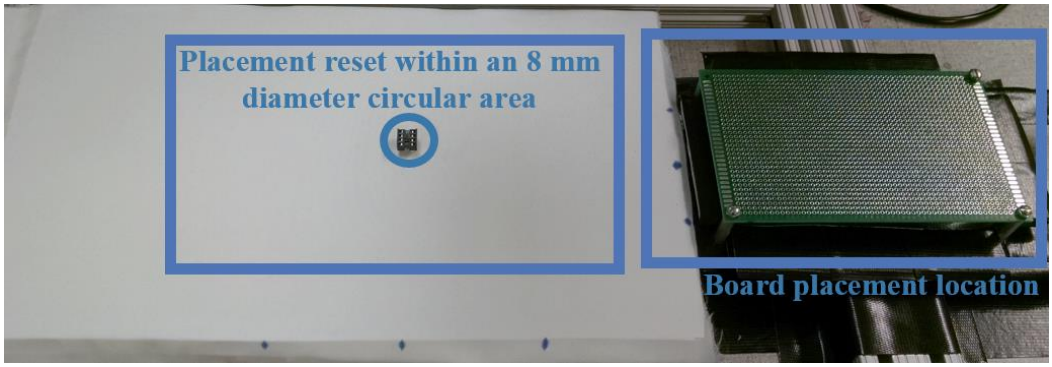

Figure 51: Overview of component & board reference positions

The initial procedure tested using this assembly combination involved a single component iteration per process. A circular area with an 8 mm diameter was randomly chosen and outlined towards the center of the camera FOV where the component was to be placed for each test iteration. A total of 40 consecutive process iterations were performed, which involved the following steps:

1. Place component and board at reference positions
2. Manually command automated sequence
3. System initialization
4. Component/board localization
5. Pick-and-place component
6. Automation sequence completion
7. Visual analysis of results quality
8. Removal of component from board
9. Replacement of component at reference position

The calculated locations of each component are shown in Figure 52 and are further summarized in Table 8. Using the same component for each placement cycle, the pin spacing was calculated within a maximum range of 0.06 mm with a standard deviation less than 0.025 mm.

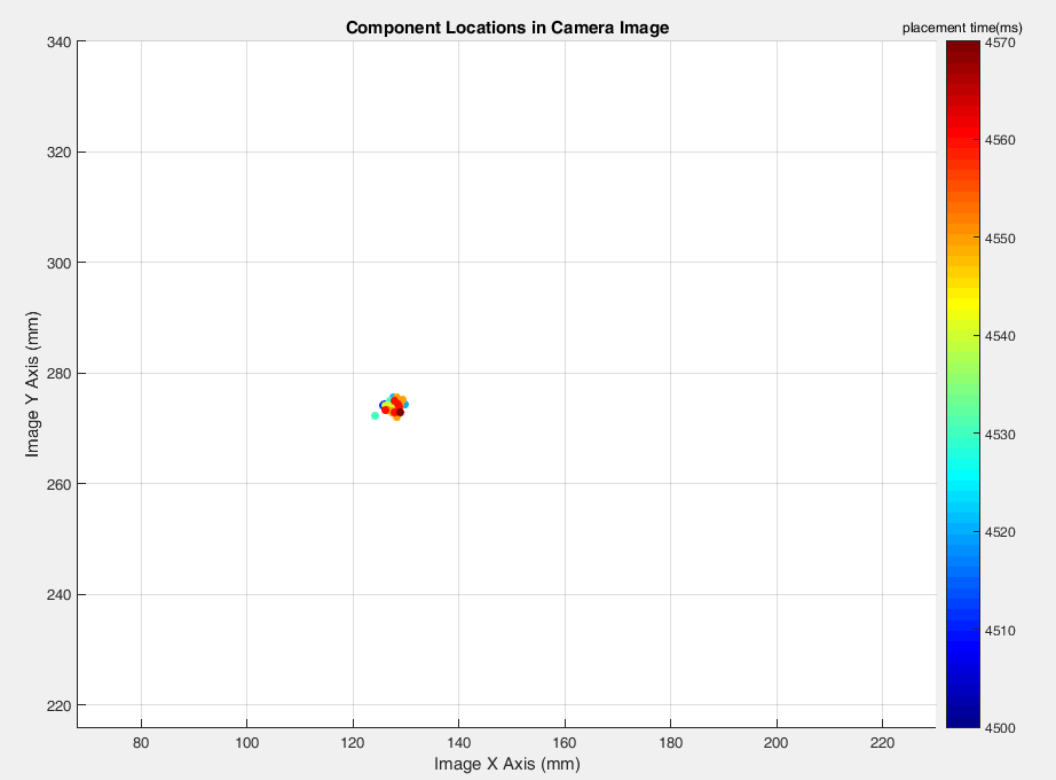

Figure 52: Results of component localization

Table 8: Localization statistics

Component Coordinates		Mean (mm)	S.D. (mm)	Max (mm)	Min (mm)	Range (mm)
Component	Pos X	127.665	1.13695274	129.75	124.13	5.62
	Pos. Y	274.01	0.86	275.7	272.19	3.51
	Angle	1.59625	1.44	4.32	-0.82	5.14
Board	Pos X	-147.19175	0.0090	-147.18	-147.23	0.05
	Pos. Y	313.512	0.0162	313.52	313.48	0.05
	Angle	0.31	0	0.31	0.31	0

The time duration of each process are shown in Figure 53. A statistical analysis was performed on this data and is summarized in Table 9.

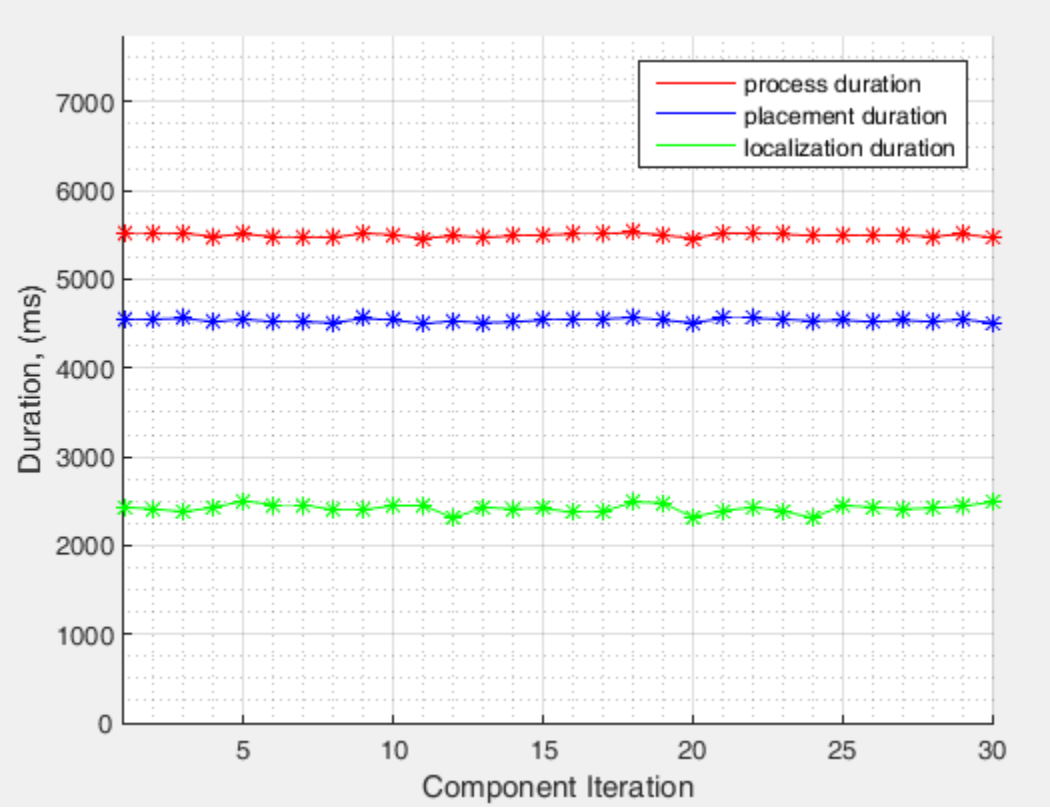

Figure 53: Graph of process durations for each component iteration

Table 9: Statistical analysis of experiment 1 durations

Parameter	Mean, (ms)	S.D., (ms)	Max, (ms)	Min, (ms)	Range, (ms)
Initialization Duration	4260	100	4450	4090	360
Localization Duration	2418	47	2500	2310	190
Pick-and-Place Duration	4537	19	4570	4500	70
Successful Component Iterations	30/30				

6.5.2 Experiment 2 Results

The next experiment was conducted using multiple components being placed in different locations on the same board. Each tray contained 10 components which were

placed across a range of approximately 111 mm in the x-axis and 69 mm in the y-axis. A total of 120 component cycles were sampled, which are graphed according to the placement cycle time in Figure 54.

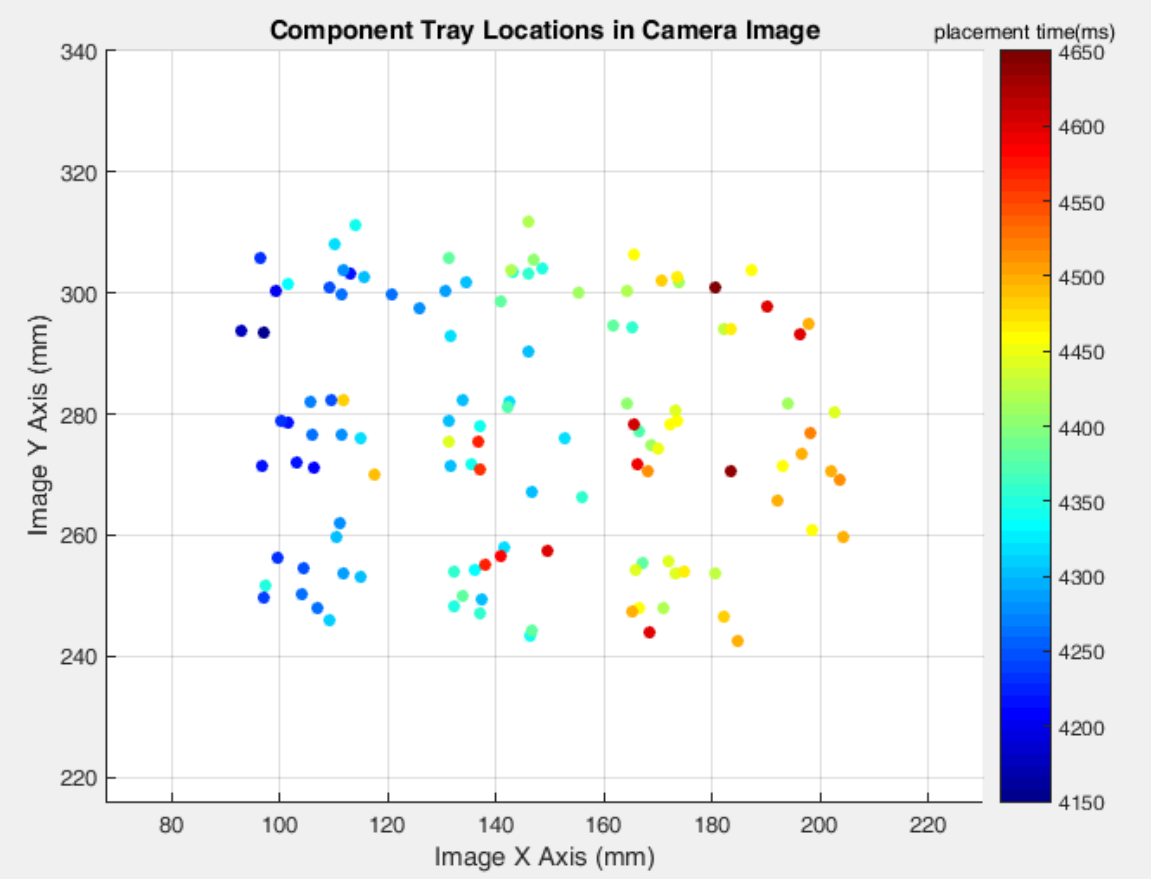

Figure 54: Localized component position distribution

Table 10: Analysis of localized board position data

Board Ref	Mean (mm)	S.D. (mm)	Max (mm)	Min (mm)	Range (mm)
Pos X	-147.196	0.01502	-147.19	-147.23	0.04
Pos. Y	313.4985	0.02075	313.52	313.48	0.04
Angle	0.31	0	0.31	0.31	0

Table 11: Analysis of process task durations

Timing	Mean (ms)	S.D. (ms)	Max (ms)	Min (ms)	Range (ms)
Localization Duration	2668	70	2800	2540	260
Component Cycle Duration	4384	112	4650	4150	500

6.5.3 Experiment 3 Results

In order to further explore the effects of placement location, an experiment was conducted using a single component per tray. The localization process was performed followed by the pick-and-place sequence. However, to allow larger data samples to be collected, the command to close the gripper was removed from the sequence so the component was not moved from its fixed reference position. Upon arrival at the inspection position, the sequence was terminated and the arm returned to the tray viewing position to begin the next process iteration. The following conditions were applied to this testing:

- Total consecutive tray cycles: 270
- Tray cycles at each reference location: 30
- Component cycles per tray: 1

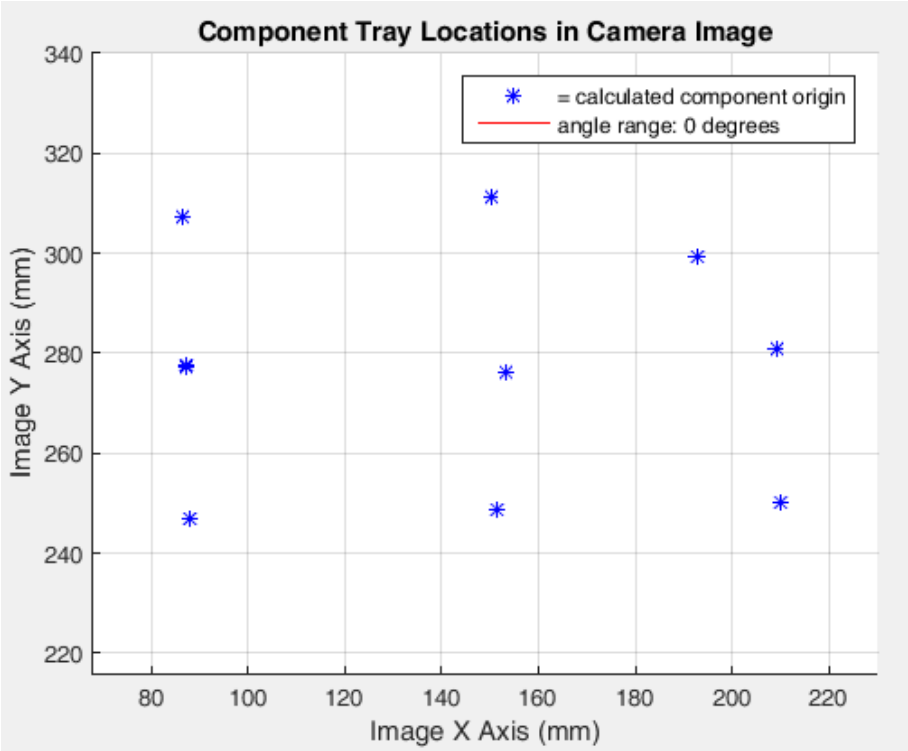

Figure 55: Distribution of localized component positions

Table 12: Pick duration comparison at various reference positions

Component Reference Position	XY Displacement Distance (mm)	Mean Pick Duration (ms)	Mean Inspection Travel Duration (ms)
Top Left	69.89	1289	573
Top Center	133.72	1340	654
Top Right	175.92	1361	688
Center Left	73.78	1289	536
Center	138.42	1279	646
Center Right	193.13	1344	703
Bottom Left	88.72	1201	598
Bottom Center	143.76	1267	659
Bottom Right	199.30	1290	722

A similar approach was implemented next to analyze the effects of component rotation. Using 2 nominal reference locations, the component was analyzed under various degrees of rotation and the following conditions were applied to the process:

- Total consecutive tray cycles: 300
- Tray cycles per reference angle: 30
- Components cycles per tray: 1

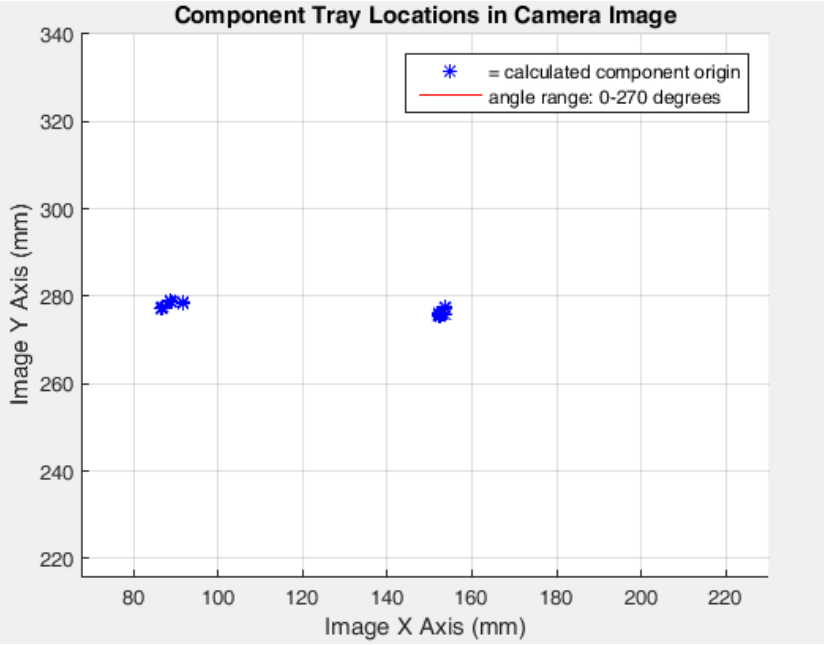

Figure 56: Distribution of localized component positions

Table 13: Pick duration results at various degrees of component rotation

Component Reference Position		Mean Calculated Angle (°)	Mean Pick Duration (ms)	Mean Inspection Travel Duration (ms)
Angle	Position			
0°	Center	1.2	1223	651
	Left	0.4	1189	541
90°	Center	90.1	1472	1110
	Left	90.2	1419	1046
180°	Center	179.7	1415	1280
	Left	179.19	1476	1337
270°	Center	270.6	1207	846
	Left	270.2	1164	899

6.5.4 Experiment 4 Results

The next experiment was designed to analyze the effects of multiple placements on a single board after improving the process based on the results of the previous experiments. The following testing characteristics were applied:

- Total consecutive tray cycles: 120
- Components cycles per tray: 8

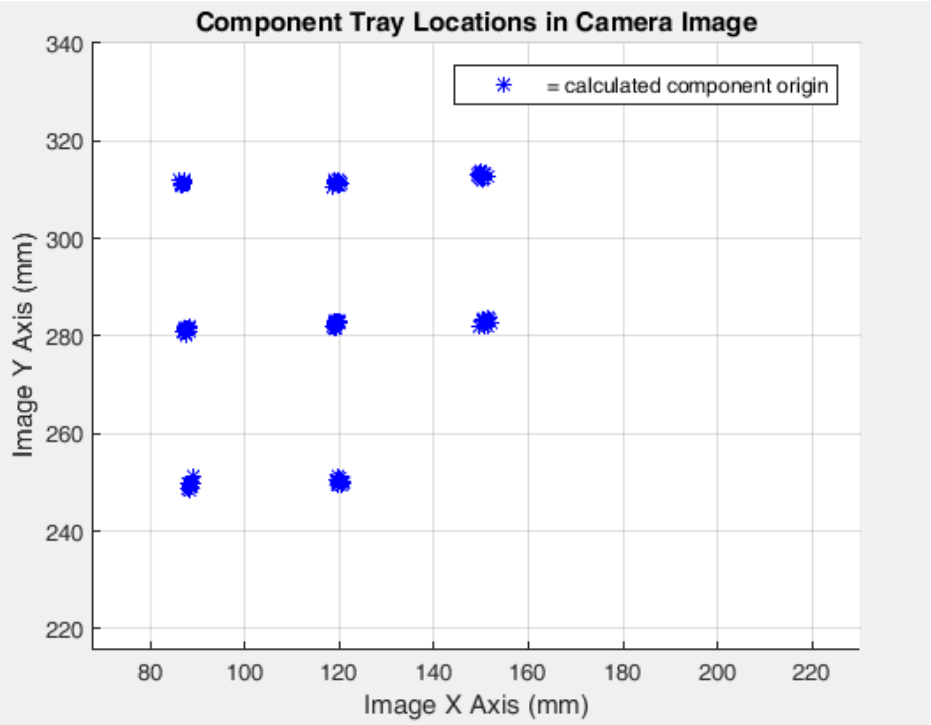

Figure 57: Distribution of localized component positions

Table 14: Resulting durations of combined process iterations

Parameter	Average (ms)	St. Dev. (ms)
Component Localization Duration	930	43.0
Component Position Movement Duration	13.7	11.5
Placement Localization Duration	167.7	16.7
Board Position Movement Duration	838.5	1.6
Total Localization Duration	1132.5	35.5
Individual Pick-and-Place Duration	3996.6	131.4
Distributed Component Cycle Duration	4098.25	28
Successful Placements	120/120	

6.5.5 Experiment 5 Results

This experiment was performed to analyze the effects of rotation on the overall process time using the following conditions:

- Total consecutive tray cycles: 24
- Components cycles per tray: 5
- Component cycles per reference angle: 20
- Total component cycles: 120

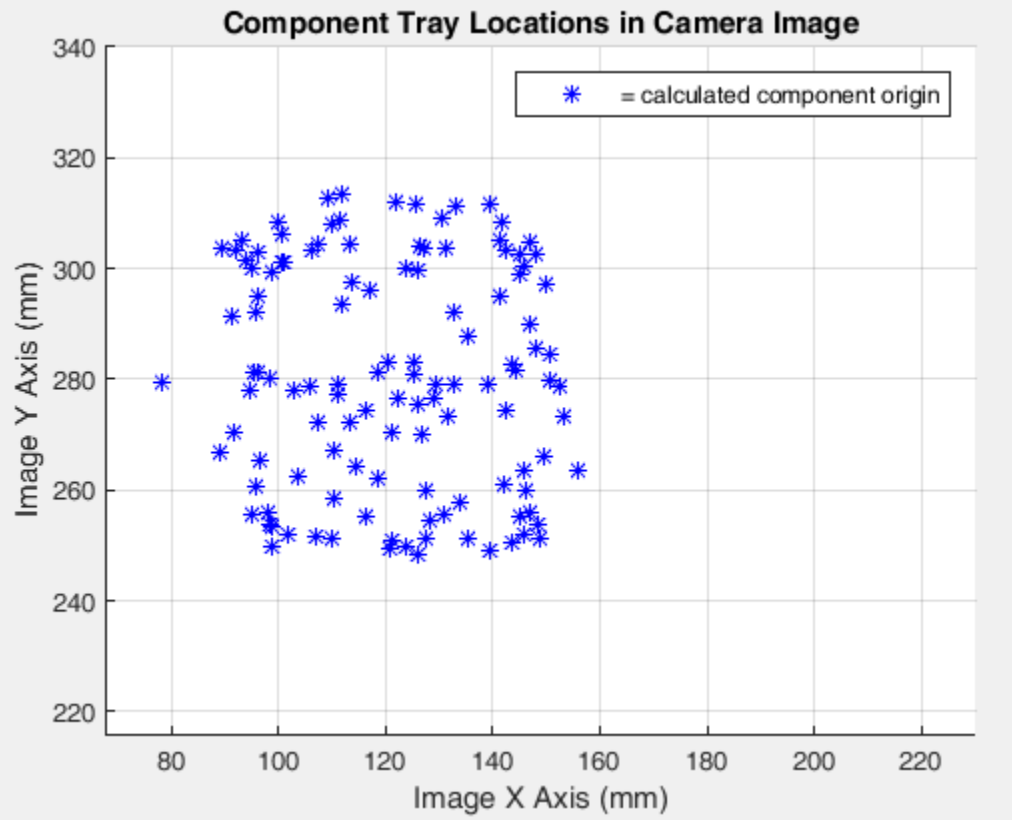

Figure 58: Distribution of localized components

Table 15: Process results of components at various degrees of rotation

Component Reference Angle	Mean Calculated Angle (°)	Mean Pick Duration (ms)	Successful Placements
-170°	-166.8	5251	20/20
-90°	-89.3	4539	20/20
-45°	-46.0	4278	20/20
45°	48.4	4607	20/20
90°	90.7	5069	20/20
170°	170.1	5480	20/20
Total	-	4870	-

6.5.6 Experiment 6 Results

The next experiment was designed to analyze effects of reducing the number of components being placed on a single board. The process was slightly modified to account for a board that requires only one component placement. Initially, a tray of 5 components was localized along with the board placement. After a single placement, the board was removed and a replacement was inserted. The system then localized the new board while maintaining the previously acquired component information to reduce unnecessary process time. Each step of the process was analyzed separately to identify potential areas of optimization. A total of 6 tray cycles were performed with 5 component per tray resulting in a total of 30 component cycle samples. All 30 of the components were successfully placed with an average process time of approximately 4.8 seconds per component.

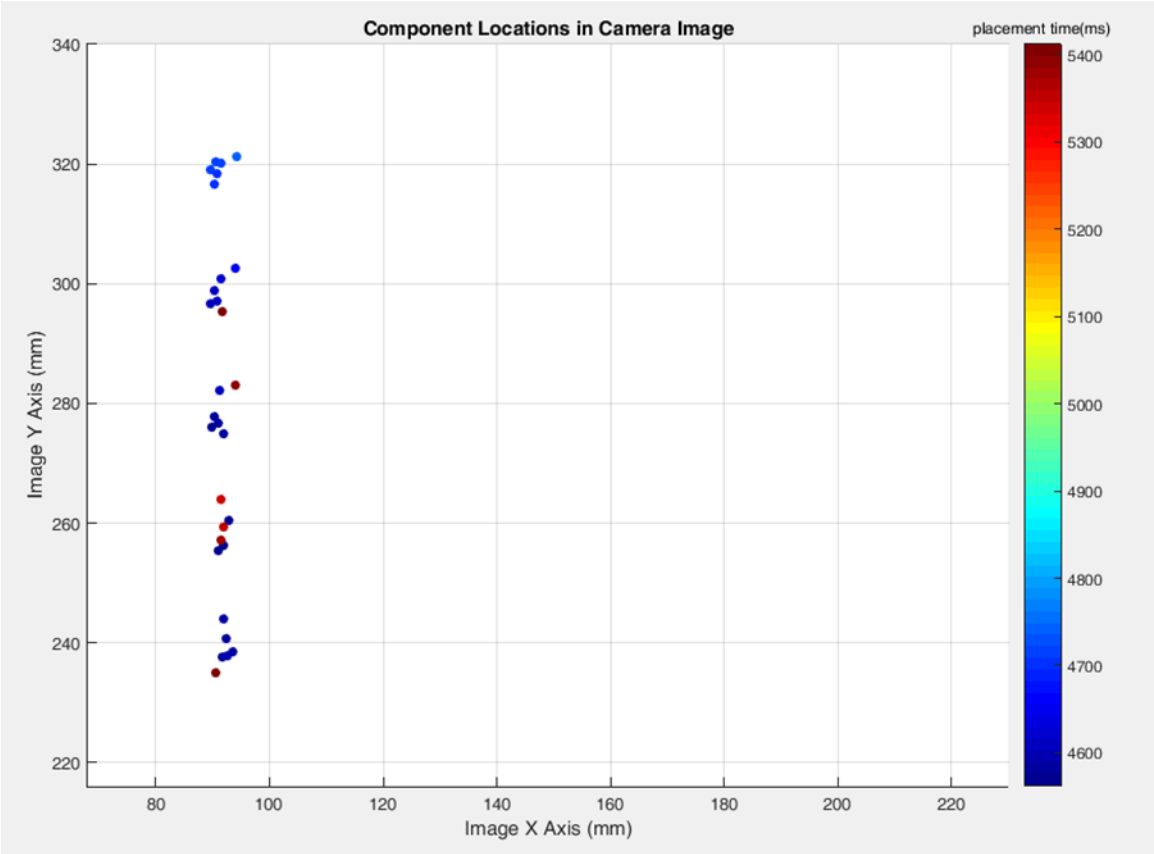

Figure 59: Distribution of localized components

Table 16: Process results summary

Parameter	Average (ms)	St. Dev. (ms)
Component Localization Duration	531	10
Placement Localization Duration	119	0.5
Board Position Movement Duration	843	6
Pick Duration	981	45
Movement to inspection pos. duration	585	30
Inspection Duration	127	68
Movement To Second Inspection Pos. Duration	942	3
Second Inspection Duration	101	4
Placement Duration	848	2
Return to Board View Position	917	2
Average Overall Process Time per Component	4775	312
Successful Placements	30/30	

6.6 Component 3: JQX-118F Electromechanical Relay

The third component tested was a mechanical relay with the part number JQX-118F. This part had several non-standard features such as a larger lead pitch of 3.2 mm and an asymmetrical pin arrangement. From above, printed part information is the only visual feature useful for calculating orientation. Due to its larger lead pitch, a custom board was created to use for placement testing. A blank PCB was machined according to the manufacturer's specifications which recommended a lead-hole diameter of 1.3 mm.

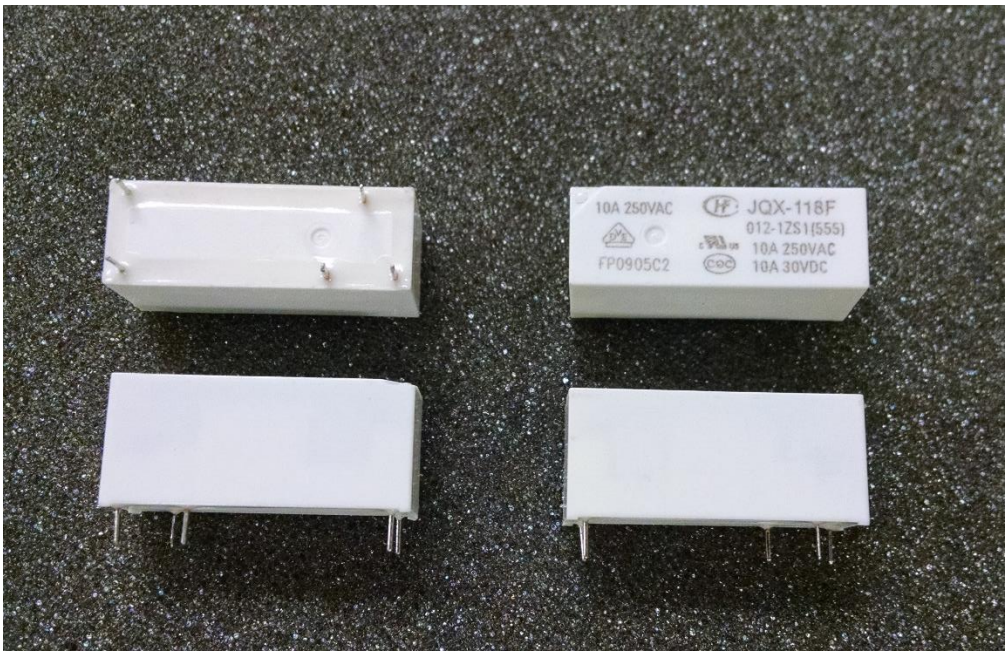

Figure 60: JQX-118F component features

Figure 61: Custom machined circuit board for JQX-118F component

The primary purpose of the experimentation with this component was to support the consistency of the results between different component types. As a worst case scenario, this experiment was performed using a single component localization and placement to provide data corresponding to the maximum process duration. Based upon observations made from the previous preliminary experimentation, several optimizations were implemented. Due to the relationship of the total arm displacement to the process duration, the distance between the component locations and the board placement was reduced. Addition parallel processing was utilized so that image analysis was primarily performed during arm movements. The results of the experiment are summarized below and demonstrate a pick-and-place duration of approximately 4.1 seconds with all attempted placements successfully performed.

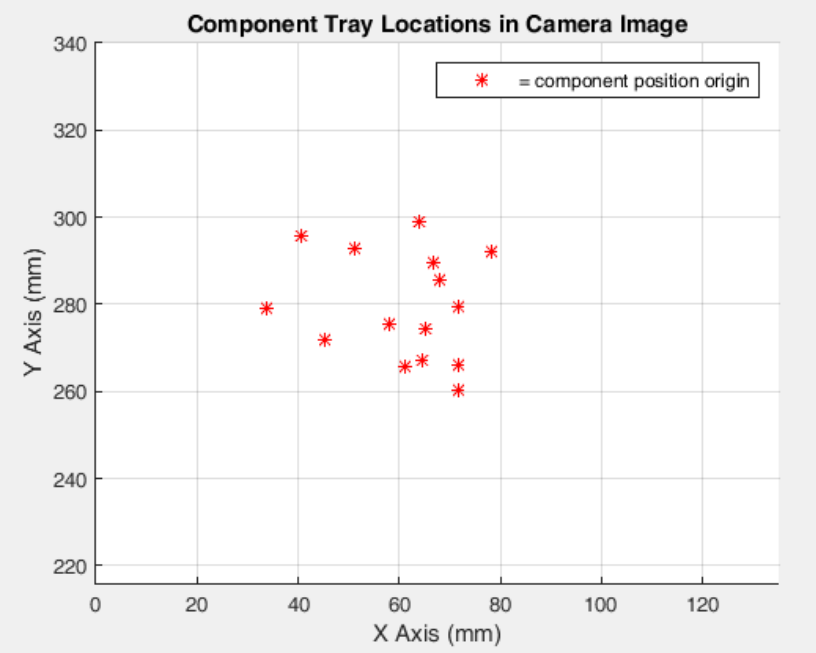

Figure 62: Distribution of localized JQX-118F component positions

Table 17: Summary of testing results using JQX-118F component

Parameter	Average (ms)	St. Dev. (ms)
Component Localization Duration	760	28
Component Position Movement Duration	8	2
Placement Localization Duration	92	5
Board Position Movement Duration	712	3
Total Localization Duration	886	26
Individual Pick-and-Place Duration	4059	106
Total Process Duration per Component	4927	109
Successful Placements	15/15	

CHAPTER 7: DISCUSSION AND CONCLUSIONS

7.1 Template Matching Analysis

The LDS method exhibited the shortest processing time compared to the other methods tested, and also experienced smaller processing time increases when parameters were adjusted to search for more components and wider angle ranges. The other methods were able to produce marginally higher match confidence, especially under rotation and when an object position approaches the image perimeter where perspective distortion is the highest. However background features could be feasibly be reduced in the workspace so that accepting lower scores would not likely result it false positive matches. Due to these reasons, the LDS method was chosen for all of the object recognition tasks.

7.2 Experimental Data Analysis

The data collected from the experimental procedures was analyzed and assessed with respect to the initial performance requirements. The two primary categories of analysis were the throughput cycle time of the pick and place process and the observed failure rate in the samples collected. The analysis of these results for each of the components tested are described in the following sections.

7.2.1 Pick-and-Place Throughput Cycle Time

The cycle time results recorded for the LM311 component exhibited repeatable data between process iterations. Using a tray of 3 components to be inserted to a single board, an average component cycle time of 4.42 s was calculated with a standard deviation of

130 ms. A repeatable pattern was observable from the graph in Figure 45 which was suspected to be a result of the displacement of the arm position for each component. Movement of the arm that originates from the board inspection position results in the largest travel time but only occurs for the first component pick of a tray. When the process was categorized based on the component cycle order, the resulting data confirmed the initial component cycle yielded the highest time of 4.57 seconds, while the subsequent cycles produced cycle times of 4.28 s. and 4.41 s respectively. The discrepancy between the cycle time of the 2nd and 3rd component can be reasonably attributed to the additional displacement determined by the board placement position. The average calculated placement positions in the y-axis of the tool coordinate system were approximately 289 mm and 268 mm which resulted in an additional 21 mm of travel for the third component placement to account for the increased placement time. A maximum cycle time standard deviation of 64 ms indicates that these results are repeatable.

Testing performed using the DIP socket component was conducted in greater depth to analyze the effects of parameter deviations. Data from the first experiment involved a single component per tray to allow consistent conditions to be used for comparison. After a total of 40 component cycles, the average placement time per component was 4.54 seconds with a standard deviation of approximately 19 ms. This high consistency between cycles with similar characteristics is significant in showing the results are repeatable and deterministic. The average time to localize the component and board was 2.42 seconds with a standard deviation of 47 ms. This can potentially increase overhead time before the pick-and-place task can begin, but is only performed once per tray. Therefore, higher quantities of components will equally distribute this constant value and

reduce the effective throughput cycle time. An experiment was performed using 10 components per tray which produced an average component cycle time of 4.38 seconds and a localization time of 2.67 seconds. Several important observations were noted with these results which were consistent with the expectations. First, the localization time was approximately 0.25 seconds longer due to the higher number of requested component matches. The selection of the low discrepancy sampling method proved beneficial because of minimizing the additional processing time. Secondly, the average cycle time was approximately 140 ms lower than the individual component testing primarily due to the reduced time of the 9 subsequent cycles compared to the 1st cycle which has the highest displacement. The third observation was that the higher quantity of components distributed the localization time among each cycle to produce an average distributed cycle time of approximately 4.65 seconds.

Experiment 3 was conducted to further analyze the characteristics of each task in the picking process, which included the individual localization tasks and each separate movement command. These results showed that the component localization took an average of approximately 265 ms and placement localization took an average of 157 ms. The movement time between localization positions was approximately 139 ms with a relative standard deviation of 0.03%. After localization, the pick time was dependent on the distance of the component from the arm, which was 1.3 seconds on average. The time to bring the component to the inspection position was dependent on the same characteristics and had an average of 642 ms. The low time deviation between similar component locations indicates the system’s level of repeatability.

The effects of rotation on the picking process was analyzed by varying the component orientation angle at two reference locations; one in the center and one on the left. The average component localization time was approximately 423 ms which was an increase from the previous experiment due to the higher rotation of the matching template. The average pick time was 1.35 seconds at the center position and 1.34 at the left position. The precise difference between the two averages was measured to be 2.33 ms which was a result of the positioning limitation time of the end effector rotations.

It is a common occurrence for some circuit board designs to have only one of a specific type of component, which would require the localization process for each component cycle. Upon discovering the specific composition of the localization time, a variety of optimizations were implemented to further reduce the overall process time. A tray of 5 components were localized once per process iteration; a new board was localized before each component placement cycle. The viewing positions were tuned to minimize unnecessary arm movements and the distance between the components and board was reduced. Additional parallel processing was implemented so that each of the localization tasks were performed during robot movements. These improvements produced an average individual component cycle time of approximately 4.78 seconds with a maximum of 4.892 seconds, which included the distribution of the localization times. These results meet the performance specification of less than 5 second pick-and-place cycles, and would improve with the addition of more components per tray.

The JQX-118F was tested using the optimizations developed from the previous experiments. To establish a maximum baseline cycle time, the experiment was performed using a single component per tray. The component location-per-cycle ranged from

approximately 79 mm to 121 mm with a localized angle range from -31° to 44°. The testing procedure according to these parameters resulted in an average placement time of less than 4.1 seconds; the localization time was effectively reduced to 886 ms, which yielded an average cycle rate of 4.93 seconds. These results represent the upper limits of the cycle time, and would be incrementally lowered with the addition of more components per tray as demonstrated in the previous testing.

7.2.2 Failure Occurrences

A failure, as previously defined, is any event which results in an incorrect component insertion into the circuit board, which primarily includes following:

- Localization error
- Missed component grasp or incorrect grasp angle
- Pin offset measurement error
- Component placement is missing or misaligned

Since the tool was precisely calibrated in each workspace, a component grasp failure would likely only occur as a result of a localization error. To prevent this occurrence, each component template was designed to emphasize features which indicate orientation angle. Feature masks were applied to suppress symmetrical features as well as features known to experience high variations such as specular surfaces. Perspective error occurs as the component position in an image radially deviates from the center, which was partially compensated by bounding the component placements on the tray to keep their locations away from the image borders. Additional masking was implemented along object edges where perspective occlusions are significant. Each template was tested at different locations and orientations to ensure the minimum correlation score was above a

general confidence threshold of 800/1000 before it was determined to be reliable for experimentation. This process was successful at preventing localization errors and grasping errors in all sample groups tested. A total of 1599 localization cycles were recorded, which resulted in no failed component grasps.

In the testing of the LM311 component, a total of 30 component cycles were analyzed, all of which resulted in a successful placement. The quality of the placements can be attributed to the precision of the calibration models and the performance of the pattern matching algorithm. The DIP socket component was exposed to a much wider range of parameter variations that allowed for detailed analysis of how the sample success rates were achieved. The first experiment included a sample size of 40 component cycles, each of which was successfully localized and placed. The localized placement position between all runs had a standard deviation of less than 0.04 mm in each of the X and Y axes, which is a strong indication of the effectiveness and repeatability of the placement localization process. The same component was used for each cycle and the pin spacing was measured within a total range of 0.06 mm. These precision characteristics were well within the required clearance radius of 0.2 mm to ensure a successful placement. Further evidence to support the achieved precision quality is provided by the additional experiments which involved many parameter variations such as modification of component quantity, location, and orientation. In total, 996 component cycles were tested using this component which resulted in success of all attempted circuit board insertion in each cycle. In the majority of tests, the component leads were formed to the correct alignment to ensure they were suitable for placement, which kept discards sparse. A total of 17/996 components were discarded due to

components not meeting the minimum measurement specifications. An additional 15 samples were tested using the JQX component to support the theorized flexibility of the methodology to handle different components and boards. All of the samples were successfully placed within the same range of precision and required only changing the matching templates and tool approach distances to handle the larger component.

7.3 Conclusions

The research conducted in this thesis was intended to propose a methodology to automate the pick-and-place assembly of non-standard components into printed circuit boards. The proposed methodology involved a 6-axis robotic manipulator guided by a precision calibrated vision system. The main performance requirements included improvement of the current manual component cycle rate of 5 seconds and a reduction of the current error rate of 1%. The experimental procedure was conducted using three separate components, each of which was able to satisfy the research goals. Each component was able to perform the pick and place process in less than 5 seconds, while the fastest observed times were less than 4 seconds. An analysis of the compositional tasks involved in the process indicated high repeatability of this data which was consistent between different components. Out 1,041 component cycles, a total of 17 components were determined unsuitable for placement and discarded, while the remaining 1,024 components were placed successfully. This repeatability was accomplished through ensuring the combined error from calibration and image localization was within the tolerance needed to ensure successful placement. This initial data strongly supports the claim that precise calibration and high confidence template localization methods are capable of achieving error rates lower than the 1% requirement.

The visual features used to identify and localize image objects were customized for each template to achieve the highest quality matching, but this process is not limited to the characteristics of only the components tested. Rather, the process of creating each image matching template can be applied to the feature sets of nearly any part type, which would result in a robust assembly system capable of handling a much wider diversity of electronic components than existing implementations.

7.4 Future Work

Several areas of optimization were determined through conducting this research that would allow throughput rates to be reduced even further. The positioning time of the VP-6242 was a key limitation to achieving faster cycle times. An example replacement could be the DENSO VS-068 robot unit, which has a rated cycle time of 0.33 seconds compared to the 0.99 second cycle time of the VP-6242; an increase by a factor of 3. Considering the JQX component which had an average combined positioning duration per cycle of 4.63 seconds, the VS series could potentially reduce the total positioning duration per component to as low as 1.55 seconds. Optimized path planning, minimizing the distance between points of interest, and minimizing unnecessary movements would further improve this number. One relatively simple approach to minimize extra movements would involve the use of additional camera units. A dedicated camera for localization would eliminate the positioning time needed to raise the arm to each viewing location. Similarly, multiple cameras for inspection would eliminate the extensive rotation time currently required to inspect each side of the component. Using the JQX as reference, this could potentially eliminate 600 ms or more from the total positioning time, resulting in throughput cycle rates less than 1 second at no sacrifice to repeatability. The

implementation of these improvements would only involve upgraded equipment and tuning of current motion parameters, neither of which would fundamentally alter any of the operational concepts which were utilized in these experiments.

REFERENCES

[1] L. Halbo and P. Ohlckers, Electronic Components, Packaging and Production, Oslo Universitet, 1995.

[2] S. Liebes, et al., "FLAIR - Robotic Printed Circuit Board Assembly Workcell," in *Robotics and Automation, IEEE International Conference on*, Philadelphia, 1988.

[3] S. Liebes, et al., "FLAIR - Robotic Printed Circuit Board Assembly Workcell," *Robotics and Automation, 1988. Proceedings., 1988 IEEE International Conference on*, vol. 3, pp. 1594-1600, 1989.

[4] K. Ono, "Detection, localization and picking up of coil springs from a pile," in *2014 IEEE International Conference on Robotics and Automation (ICRA)*, Hong Kong, 2014.

[5] A. Pochyly, T. Kubela, M. Kozak and P. Cihak, "Robotic Vision for Bin-Picking Applications of Various Objects," in *Robotics (ISR), 2010 41st International Symposium on and 2010 6th German Conference on Robotics (ROBOTIK)*, Munich, 2010.

[6] L. Lin, Y. Yang, Y. Song, B. Nemec, A. Ude, J. Rytz and A. Buch, "Peg-in-Hole assembly under uncertain pose estimation," in *Intelligent Control and Automation (WCICA), 2014 11th World Congress on*, Shenyang, 2014.

[7] R. Patel, M. Hedelind and P. Lozan-Villegas, "Enabling robots in small-part assembly lines: The "ROSETTA approach" - an industrial perspective," in *Robotics; Proceedings of ROBOTIK 2012; 7th German Conference on*, Munich, 2012.

[8] E. Gonzalez-Galvan, et al., "Precise industrial robot positioning and path-tracking over large surfaces using non-calibrated vision," in *Robotics and Automation (ICRA), 2011 IEEE International Conference on*, Shanghai, 2011.

[9] K. Rahardja and A. Kosaka, "Vision-based bin-picking: recognition and localization of multiple complex objects using simple visual cues," *Intelligent Robots and Systems '96, IROS 96, Proceedings of the 1996 IEEE/RSJ International Conference on*, vol. 3, pp. 1448-1457, 1996.

[10] Denso Wave, "Denso Robot User Manuals," 2016. [Online]. Available: densorobotics.com/content/user_manuals/19/index.html. [Accessed 8 July 2016].

[11] National Instruments, "A Practical Guide to Machine Vision Lighting," 14 Oct 2015. [Online]. Available: ni.com/white-paper/6903/en/. [Accessed 8 July 2016].

[12] B. Batchelor, Machine Vision Handbook, London: Springer, 2012.

[13] R. Klette, Concise Computer Vision, London: Springer, 2014.

[14] K.-S. Kwon and S. Ready, Practical Guide to Machine Vision Software: An Introduction with LabVIEW, Weinheim: Wiley, 2014.

[15] G. Farinella, S. Battiato and S. Singh, Advanced Topics in Computer Vision, London: Springer-Verlag, 2013.

[16] W. Burger and M. Burge, "Principles of Digital Image Processing; Fundamental Techniques," Springer, London, 2009.

[17] National Instruments, "IMAQ Vision Concepts Manual," NI, Austin, Texas, 2000.

[18] National Instruments, "NI Vision Concepts Help," NI, June 2015.

[19] N. Metropolis, A. ROSENBLUTH, M. ROSENBLUTH and A. TELLER, "Equation of State Calculations by Fast Computing Machines," *The Journal of Chemical Physics,* vol. 21, no. 6, 1953.

[20] W. Morokoff and R. Caflisch, "QUASI-RANDOM SEQUENCES AND THEIR DISCREPANCIES," *Siam J. Sci. Comput.,* vol. 15, no. 6, pp. 1251-1279, 1994.

[21] K. R. Haberle and R. J. Graves, "Cycle time estimation for printed circuit board assemblies," *IEEE Transactions on Electronics Packaging Manufacturing,* vol. 24, no. 3, pp. 188-194, 2001.

[22] D. Rosso, "Global Semiconductor Sales Up Slightly in May," Semiconductor Industry Association, 2016.

[23] A. Pochyly, et al, "3D vision systems for industrial bin-picking applications," in *MECHATRONIKA, 2012 15th International Symposium*, Prague, 2012.