

FAIRFIELD UNIVERSITY School of Engineering and Computing

**Implementation, Development, and Evaluation of a Single-Camera Robotic
Pick-and-Place System for Dynamic Object Detection, Tracking, and Pickup**

A Thesis by

John Chiodo

Department of Mechanical Engineering

Submitted in partial fulfillment of the requirements for the degree of

Master of Science in Mechanical Engineering

Spring 2025

Advisors

Dr. Naser Haghbin, *Professor, Fairfield University*

Dr. Sriharsha Sundarram, *Professor, Fairfield University*

Dr. Andy Judge, *Professor, Fairfield University*

Dr. Shahrokh Etemad, *Professor, Mechanical Engineering Chair, Fairfield University*

ABSTRACT

This thesis investigates the performance and limitations of a simplified vision-guided robotic system designed to perform dynamic pick-and-place tasks using a single, end-mounted camera. In contrast to high-cost, multi-sensor industrial solutions, this approach emphasizes low-cost implementation, minimal calibration, and off-the-shelf components, making it a good candidate for small to mid-sized manufacturers seeking entry-level automation. The study explores whether such a system can achieve practical accuracy and reliability when operating in real-time environments, and what specific system parameters most critically influence performance.

The experimental setup includes a six-axis Epson VT6 robotic manipulator equipped with a parallel gripper and an industrial camera mounted directly to the end of the arm. All vision processing and motion control were implemented using the Epson RC+ software suite, with no external sensors or auxiliary computing. The camera captures part motion on a conveyor belt, and the robot attempts to intercept the part in motion using one of three predictive strategies: a method based on vertical descent time, a method that models full diagonal travel using a closed-form kinematic solution, and an ambush-style method that positions the robot in advance at a fixed location and waits for the part to arrive. Experiments were conducted to evaluate system accuracy across three key metrics: positional error in the planar motion, and rotational misalignment. Each predictive strategy was tested under controlled baseline conditions to determine which yielded the most accurate and repeatable results. In addition, a set of system-level parameters—including scanning height, conveyor belt speed, number of velocity estimation iterations, and robot motion constraints—were varied independently to assess their impact on performance.

Results show that the Hypotenuse Method produced the most accurate positional predictions in the x direction and the most consistent rotational alignment, with average errors of 14.57 mm and 5.91°, respectively. The Ambush Method outperformed others in y direction accuracy, averaging only 21.12 mm of error. Overall, the system maintained high trial success rates in baseline configurations but exhibited significant degradation in performance under extreme scanning heights, fast conveyor speeds, and low velocity estimation iterations. Compared to industry standards, the system’s accuracy falls short of high-end expectations, particularly in applications requiring sub-millimeter precision or sub-degree alignment. However, in less demanding environments—such as basic sorting, semi-structured part transfers, or part identification—the results suggest that this low-cost, single-camera approach can serve as a viable alternative. The data collected also provide a valuable baseline for future optimization.

In summary, this work presents a quantitative evaluation of a minimalist vision-guided robotic system, establishing its strengths, limitations, and key performance boundaries. It offers a reference framework for practitioners and researchers seeking cost-effective automation strategies using commercially available hardware and straightforward integration.

ACKNOWLEDGMENTS

I would like to express my sincere gratitude to my thesis advisor, Dr. Naser Hagbin, for his invaluable guidance, mentorship, and support throughout the duration of this project. His insight and encouragement were instrumental to the successful completion of this work. I am also deeply grateful to my thesis committee members, Dr. Sriharsha Sundarram and Dr. Andy Judge, for their feedback, technical expertise, and encouragement throughout the research process. This project was made possible in part through the generous support of the NASA Connecticut Space Grant Consortium (Award No. P-2331), whose funding was vital to the development of this work. I would also like to extend my thanks to Accurate Lock & Hardware (Stamford, CT) for supplying the Epson robot, camera, gripper, and associated hardware—resources without which this research would not have been feasible. Lastly, I would like to thank the team of dedicated undergraduate research assistants who contributed significantly to this project, Michaela Nanna, Jack McKenna, Nicole Reagan, and Joseph Maucieri. Their collaboration, enthusiasm, and hard work played an essential role in bringing this system to life.

NOMENCLATURE

Term or Symbol	Definition
x	Horizontal position of the part or gripper in the global frame
y	Horizontal position of the part or gripper in the global frame
z	Vertical height or elevation of the part or gripper in the global frame
u	Angular orientation of the part or gripper, around z-axis
T	Predicted time to intercept the part (s)
v_x	Estimated horizontal velocity of the part in the x-axis (mm/s)
v_y	Estimated horizontal velocity of the part in the y-axis (mm/s)
a	Maximum acceleration of the robotic arm (mm/s ²)
R	Maximum speed of the robotic arm (mm/s)
H	Camera scanning height above the conveyor (mm)
H_{pickup}	Pickup height or vertical position of the part in the z-axis (mm)
LiDAR	Light Detection and Ranging; a remote sensing method that uses laser pulses to measure distance to an object, commonly used in robotic systems for 3D mapping and obstacle detection.
ROS	Robot Operating System – middleware used for robotic control and sensor integration
IQR	Interquartile Range – used in outlier detection
APF	Artificial Potential Fields – a path planning algorithm for robotics
RRT	Rapidly-exploring Random Tree – a motion planning algorithm for high-dimensional spaces
RL	Reinforcement Learning – an AI method for training robots through experience
DQN	Deep Q-Network – a neural network architecture used in reinforcement learning
RC+	Epson’s proprietary software suite for robot and vision control
mm	Millimeters – unit of linear distance
mm/s	Millimeters per second – unit of velocity
mm/s ²	Millimeters per second squared – unit of acceleration
°	Degrees – unit for angular measurements

TABLE OF CONTENTS

ABSTRACT	2
ACKNOWLEDGMENTS	4
NOMENCLATURE.....	5
TABLE OF CONTENTS.....	6
LIST OF TABLES	9
LIST OF FIGURES	10
CHAPTER I: Introduction.....	11
CHAPTER II: Background and Literature Review	19
2.1 Introduction to Vision-Guided Robotic Systems	19
2.2 Advanced Multi-Sensor Robotic Systems	19
2.2.1 Multi-Camera and LiDAR Configurations.....	19
2.2.2 Robot Operating System (ROS) Frameworks	20
2.3 Robotic Motion Planning Algorithms	20
2.3.1 Rapidly-exploring Random Trees (RRT) and Artificial Potential Fields (APF).....	20
2.3.2 Reinforcement Learning Approaches	21
2.4 Simplified Single-Camera Systems	21
2.4.1 Monocular Vision Challenges and Benefits	21
2.4.2 Trajectory Prediction Models	22
Direct Kinematic Method	22
Hypotenuse Method.....	22
Ambush Method	23
2.5 Influence of System Parameters on Accuracy	24

2.5.1 Conveyor Speed Effects	24
2.5.2 Scanning Height Impact	24
2.5.3 Iterative Velocity Estimation	24
2.5.4 Robotic Motion Constraints.....	25
2.6 Summary and Identified Research Gap	25
CHAPTER III: DESIGN METHODOLOGY	27
3.0 Research Objectives and Associated Methods.....	27
Objective 1: Determine Maximum Achievable Accuracy	27
Objective 2: Numerically Compare System Accuracy to Industry Standards	27
Objective 3: Identify Practical Limitations of the Single-Camera Setup	28
3.1 System Development and Architecture.....	28
3.2 Experimental Testing Methodology	31
3.2.1 Predictive Pickup Methods	33
3.2.2 Additional Variable Changes	36
3.2.3 General Testing Procedure.....	37
3.2.4 Data Processing and Outlier Removal	38
CHAPTER IV: RESULTS	40
4.0 Results by Research Objectives	40
4.1 Maximum Achievable Accuracy	42
4.1.1 Direct Kinematic Method	44
4.1.2 Hypotenuse-Based Trajectory Method	45
4.1.3 Ambush Method	46
4.1.4 Comparative Summary of Pickup Methods.....	47

4.2 Comparison of System Accuracy to Industry Standards	52
4.3 Practical Limitations Identified	55
4.3.1 Scanning Height	56
4.3.2 Conveyor Belt Speed	60
4.3.3 Velocity Estimation Iterations.....	63
4.3.4 Robot Motion Constraints	66
4.3.5 Summary of Failed Attempts	69
4.4 Summary of Key Findings	71
Chapter V: Conclusion.....	75
5.0 Introduction.....	75
5.1 Summary of Key Results	76
5.2 Interpretation and Analysis of Results	78
5.2.1 Pickup Method.....	79
5.2.2 Scanning Height	81
5.2.4 Iteration Count.....	85
5.2.5 Robot Motion Limits	87
5.3 Feasibility, System-Specific Considerations, and Research Objective Reflection ..	90
5.4 Limitations and Future Work	93
5.5 Concluding Remarks.....	97
REFERENCES	100
Appendix A.....	A1
Appendix B.....	A13

LIST OF TABLES

Table 1: Overview of System Parameters and Their Functional Roles	17
Table 2: Summary of Pickup Method Performance	23
Table 3: Summary of Key System Parameter Performance	25
Table 4: Setup Components	29
Table 5: Key Variable Baseline Values	32
Table 6: Component Error and Detection	47
Table 7: Standards Comparison	55

LIST OF FIGURES

Figure 1: Core control logic of the vision-based system.	14
Figure 2: Visual comparison of the three pickup methods: Direct Kinematic, Hypotenuse, and Ambush.	15
Figure 3: Diagram of Experimental Setup	32
Figure 4: Experimental Setup	33
Figure 5: Error Visualization	44
Figure 6: Displacement Inaccuracy vs. Pickup Method	48
Figure 7: Normalized Error for the Pickup Methods	51
Figure 8: Displacement Inaccuracy vs. Scanning Height	58
Figure 9: Normalized Error vs. Scanning Height	59
Figure 10: Displacement Inaccuracy vs. Conveyor Speed	61
Figure 11: Normalized Error vs. Conveyor Speed	62
Figure 12: Displacement Inaccuracy vs. Iteration Count.....	65
Figure 13: Normalized Error vs. Iteration Count.....	66
Figure 14: Displacement Inaccuracy vs. Robot Motion Limits.....	68
Figure 15: Normalized Error vs. Robot Motion Limits	69

CHAPTER I: Introduction

Robotic pick-and-place systems are foundational elements in modern manufacturing and industrial automation. These systems allow robots to detect, grasp, and relocate objects with high reliability across diverse production environments. Traditionally, achieving the precision required for these tasks has involved the use of complex multi-sensor configurations, external vision networks, and/or sophisticated machine learning models. While effective, these systems present notable drawbacks, particularly for small to medium-sized manufacturers [1], [2]. These include high implementation and maintenance costs, the need for extensive and often time-consuming calibration procedures, and limited adaptability when production layouts or product types change. For larger industrial players with more infrastructure and dedicated engineering teams, such systems may be justified. However, for smaller-scale operations seeking cost-effective automation, these requirements can be significant barriers to adoption. As a result, there is growing interest in developing simplified, low-cost alternatives that can deliver reasonable accuracy without excessive system overhead [3].

This thesis evaluates one such alternative: a vision-guided robotic pick-and-place system that employs a single, end-mounted camera on a six-axis industrial robot. Unlike external vision setups or systems requiring intricate software pipelines, this design relies solely on commercially available, off-the-shelf components and the manufacturer's supplied control software. By eliminating the need for additional sensors or custom integration, the system minimizes overall complexity and cost [3], [4]. The central research objective of this work is to assess the practical limitations of this simplified vision-based approach in dynamic environments. Specifically, the study investigates how different

system parameters influence accuracy and seeks to characterize the trade-offs inherent in this single-camera design.

In this context, "accuracy limitations" are defined across three key dimensions: (1) positional error, or the difference between the predicted and actual position of the part at the moment of pickup; (2) angular misalignment, referring to discrepancies between the predicted and actual orientation of the part; and (3) repeatability, or the system's ability to deliver consistent results across repeated trials under the same conditions. These metrics reflect both the robot's precision and the vision system's reliability, which are crucial for successful pick-and-place operations.

Several system parameters were selected as experimental variables due to their anticipated impact on prediction performance. These included conveyor belt speed, scanning height of the camera, the maximum speed and acceleration constraints of the robot, and the number of velocity estimation iterations used to calculate part motion. By systematically varying these factors, the study aims to uncover their individual and collective effects on system performance. For instance, it was anticipated that faster conveyor belt speeds could introduce greater positional error, potentially caused by motion blur and reduced processing time [5]. It was also expected that changes in scanning height could influence image resolution and field of view, potentially impacting tracking reliability [6].

This kind of system is especially relevant to small and medium-sized manufacturers that may not have the financial or technical resources to implement complex automation solutions. For example, consider a small company with a single production line. This company may not be able to invest in advanced vision systems or custom robotic software,

but could still benefit greatly from a commercially available robotic arm equipped with an onboard camera. Even if the system does not achieve perfect accuracy in every instance, the cost savings and operational flexibility it offers may outweigh the marginal loss in performance [4]. Similarly, businesses with dynamic product lines, frequent part changes, or evolving layouts can leverage the system's mobility and minimal calibration needs to redeploy it efficiently across different tasks without incurring major setup costs.

The initial phase of this research involved designing the robotic control framework and assembling the test environment. A central goal was to isolate physical and algorithmic factors from software engineering complexities. This was achieved by implementing a minimal yet robust control program that enabled the robot to scan, track, and intercept moving parts on a linear conveyor. The robot continuously monitored parts using its mounted camera, calculated future part positions based on observed velocities, and executed pickup motions aimed at intercepting the part as it passed underneath. Figure 1 illustrates the core logic of this system. Once the robot is activated, it enters a feedback loop where it detects the part, estimates velocity, and incrementally moves toward the predicted intercept location. This cycle repeats for a predefined number of iterations (the “number of velocity estimation iterations” parameter) before the robot attempts the pickup. Accuracy was measured by the difference between the robot’s predicted pickup coordinates and the actual position of the part, at the moment of attempted pickup.

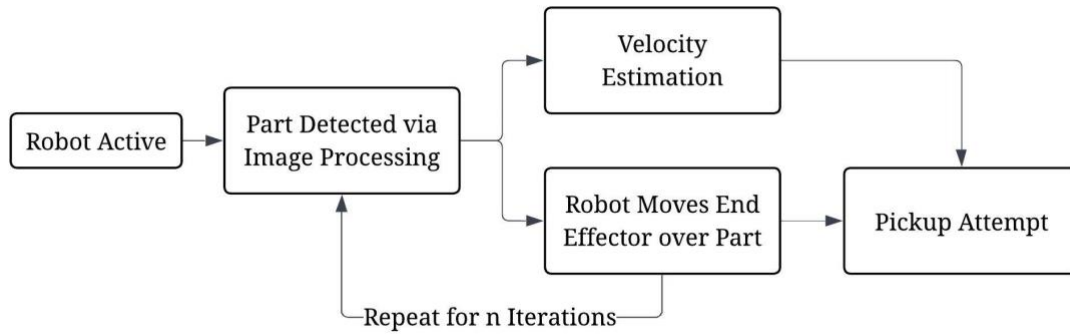


Figure 1: Core control logic of the vision-based system. This diagram shows how the program operates, using image processing, robot control, and velocity estimation.

A key focus of the experimental phase was the evaluation of three distinct pickup prediction methods. The first, the Direct Kinematic (Height-Offset) method, estimated pickup timing based solely on vertical descent time from the camera's scanning position to the conveyor surface, as illustrated in the left portion of the diagram in Figure 2. The second approach, the Hypotenuse Method, accounted for diagonal motion and solved for the total travel time using a quadratic equation. This method models the robot's actual diagonal trajectory toward the predicted part location and typically resulted in better alignment. The third strategy, the Ambush Method, bypassed motion prediction altogether. The gripper pre-positions itself above the conveyor, ahead of the part, and waits for the part to pass underneath. This eliminates trajectory estimation error but increases sensitivity to timing and part velocity fluctuations. Figure 2 below offers a visual comparison of these strategies and serves as a conceptual reference for understanding how each method attempts part interception.

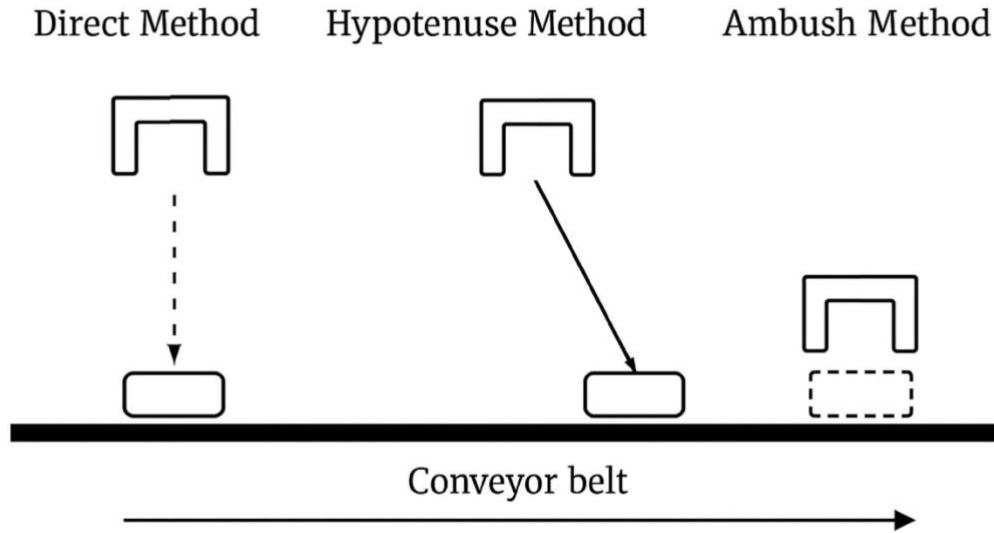


Figure 2: Visual comparison of the three pickup methods: Direct Kinematic, Hypotenuse, and Ambush. The part is shown on the conveyor, and the gripper is shown as an inverted “u” channel.

In addition to comparing prediction strategies, the study assessed how several key system parameters influenced the robot’s ability to accurately track and intercept moving parts. One such parameter was conveyor belt speed, which was varied to observe how the system performed when parts moved at different velocities. Faster conveyor speeds reduce the time available for the robot to detect, track, and respond to an object’s motion. Additionally, higher speeds may introduce motion blur in the captured images, making it more difficult for the vision system to accurately determine part location and trajectory. Testing across multiple speeds helped evaluate how well the system could maintain accuracy under more demanding, real-time constraints. Scanning height, the vertical distance between the camera and the conveyor belt, was another critical parameter. Lower scanning heights were expected to provide better image resolution and more detailed part

contours, which can improve detection accuracy. However, it was anticipated that a lower camera position would reduce the field of view and shorten the time the object remains visible to the camera. Conversely, higher scanning positions were expected to increase visibility across a wider portion of the conveyor, but potentially degrade image quality and introduce greater uncertainty in part detection. Robot motion constraints, the maximum speed and acceleration of the robotic arm, were also varied. These limits define how quickly the robot can move to reach a predicted pickup position. Slower motion settings were anticipated to offer smoother, more controlled movement but might reduce responsiveness, particularly for fast-moving objects [7]. Finally, the number of velocity estimation iterations—the number of camera frames used to calculate the object’s speed—was modified to study its impact on motion prediction. It was expected that using more iterations would yield a more reliable velocity estimate by averaging out random noise, but they also lengthen the observation period and delay the robot’s response. Using fewer iterations was anticipated to reduce latency but potentially lead to inaccurate or erratic predictions due to limited data. Each of these parameters represents a real-world trade-off between speed, accuracy, and system complexity. These parameters are shown in Table 1 below. They are explored in greater detail in Chapter III: Design Methodology, where their implementation and impact on system performance are systematically analyzed.

Table 1: Overview of System Parameters and Their Functional Roles

Key Parameter	Description
Conveyor Belt Speed	How fast the part moves, higher speeds reduce tracking time and introduce motion blur, lower speeds reduce accuracy.
Scanning Height	The vertical distance between the camera and the conveyor. Lower heights improve image resolution but reduce visibility time; higher heights increase field of view but reduce image clarity.
Robot Motion Limits	Refers to the robot's maximum speed and acceleration. Faster settings improve responsiveness but may introduce instability; slower settings offer smoother movement but reduce reactivity.
Velocity Estimation Iterations	Number of image frames used to estimate part velocity. More iterations improve accuracy through averaging but increase response time; fewer iterations reduce delay but risk noisy predictions.

Together, these tests allowed for a broad assessment of the system's operational envelope. The goal was not to optimize the system or create a comprehensive solution, but rather to document its limitations and evaluate its feasibility for real-world applications. In essence, the research seeks to answer the following overarching question: To what extent can a single-camera, end-mounted robotic system perform dynamic pick-and-place tasks in a cost-sensitive industrial setting, and what specific parameters govern its accuracy?

Three primary research objectives were defined to address this question:

1. Objective 1: Determine Achievable Accuracy of the System: Quantify the system's performance across various conditions using positional, angular, and repeatability metrics.

2. Objective 2: Numerically Compare System Accuracy to Industry Standards: Identify how the system's performance compares to industry standards with regards to robotics.
3. Objective 3: Identify Practical Limitations of the Single-Camera Setup: Test and evaluate the common approaches to pickup technique, and key system parameters to discover how they affect pickup accuracy and reliability.

These objectives give rise to specific research questions: What is the reasonable achievable accuracy for a single-camera, end-mounted robotic system? Which predictive method results in the least positional and angular error? How does each variable (e.g., scanning height, conveyor speed, robot motion constraints) affect the system's accuracy and repeatability? Could such a system be viable for use in real-world industrial applications with limited resources?

By generating controlled experimental data, this study contributes actionable insight to researchers and engineers working to develop low-cost, flexible automation systems. The findings help delineate the boundary between systems that are accurate enough for deployment and those requiring further refinement or alternative architectures. The remainder of this thesis provides a thorough literature review of related work in robotic vision and motion planning (Chapter II), a detailed breakdown of the system design and testing methodology (Chapter III), a presentation and analysis of experimental results (Chapter IV), and a concluding discussion on the implications of the findings and opportunities for future development (Chapter V).

CHAPTER II: Background and Literature Review

2.1 Introduction to Vision-Guided Robotic Systems

Vision-guided robotic systems have become a cornerstone of modern industrial automation, particularly in applications involving pick-and-place operations within dynamic environments. These systems combine robotic manipulators with visual sensing and motion planning to enable object recognition, localization, and interaction in real time. In traditional high-precision applications, such as automotive manufacturing, semiconductor assembly, and pharmaceutical packaging, these systems utilize a variety of visual and non-visual sensors to achieve robust performance across diverse conditions [1], [2].

2.2 Advanced Multi-Sensor Robotic Systems

2.2.1 Multi-Camera and LiDAR Configurations

Contemporary high-accuracy systems often employ multi-camera networks, stereo vision setups, or external LiDAR systems to provide rich spatial awareness. These configurations support advanced features such as three-dimensional object tracking, adaptive motion planning, and obstacle avoidance. For instance, Wang et al. implemented a stereo camera array to reconstruct 3D environments for robotic arms, enabling complex pick-and-place operations with sub-millimeter accuracy [8]. Similarly, Ge et al. integrated a vision system with external LiDAR to facilitate object tracking and spatial coordination in unpredictable environments [9]. These advanced systems, however, come at the cost of increased complexity. They require sophisticated calibration procedures, synchronized data acquisition pipelines, and custom software integration—most commonly through

middleware platforms such as the Robot Operating System (ROS). The computational overhead and financial cost of these setups limit their practical deployment in smaller-scale or budget-constrained industrial environments.

2.2.2 Robot Operating System (ROS) Frameworks

ROS remains a dominant architecture for robotic development due to its modularity and extensive open-source library. ROS-based systems benefit from a wide array of prebuilt packages for motion planning, perception, and hardware interfacing. However, some studies point out the significant overhead associated with deploying ROS, especially in systems where rapid prototyping and minimal configuration are essential. Guo et al. emphasize that ROS-based systems often require external computing units, careful dependency management, and multi-node communication frameworks, complicating deployment for applications intended to be mobile or quickly reconfigurable [10].

2.3 *Robotic Motion Planning Algorithms*

2.3.1 Rapidly-exploring Random Trees (RRT) and Artificial Potential Fields (APF)

A core technique in robotic motion planning is the Rapidly-exploring Random Tree (RRT) algorithm and its optimized derivatives. RRT is particularly well-suited for high-dimensional configuration spaces, enabling real-time planning in environments with dynamic obstacles. Ding et al. proposed an APF-RRT hybrid model that integrates artificial potential fields to guide tree growth while avoiding obstacles, significantly improving convergence rates and motion efficiency in simulated environments [11]. Despite their effectiveness, these algorithms require high processing power and are typically

implemented alongside ROS or other high-level control platforms, making them unsuitable for simpler, low-cost systems limited to a single vision input.

2.3.2 Reinforcement Learning Approaches

Reinforcement Learning (RL) has gained traction as a method for endowing robots with adaptive, experience-driven control policies. RL models, including Deep Q-Networks (DQNs), have been used in conjunction with multi-camera setups to train robots to respond to unpredictable trajectories and perform dynamic interception. Zhang et al. developed a vision-based RL system that improved grasping success in cluttered environments by 18% after repeated training cycles [12]. Another study by Lin et al. demonstrated that RL-enhanced trajectory planning increased interception rates by 22% compared to traditional methods [13]. However, RL models demand large training datasets, extended training periods, and substantial computing infrastructure, which poses challenges for implementation in compact, cost-sensitive platforms like those explored in this study.

2.4 Simplified Single-Camera Systems

2.4.1 Monocular Vision Challenges and Benefits

By contrast, some recent efforts have investigated single-camera, end-effector-mounted vision systems as an alternative to multi-sensor solutions. These monocular systems reduce hardware costs and streamline calibration requirements, offering an appealing solution for small-to-medium enterprises seeking robotic automation without the burden of complex infrastructure. For instance, Qin et al. implemented a single-camera system for object sorting on a conveyor, achieving a 92.5% classification success rate in a

controlled environment [3]. Another study by Li et al. applied monocular vision to detect and correct part alignment in basic assembly tasks, demonstrating consistent operation in low-precision applications [4]. While effective in constrained settings, these systems generally struggle with depth estimation and predictive motion tracking, particularly when object velocity varies or when the robot must coordinate timing with moving targets.

2.4.2 Trajectory Prediction Models

Direct Kinematic Method

Trajectory prediction remains a major challenge for single-camera systems. Traditional kinematic models assume linear, constant-velocity motion and rely on sampling to forecast object trajectories. Zhang et al. explored a kinematic method that estimated interception timing using vertical descent time and average object velocity, resulting in average errors of 14.2 mm in planar positioning during controlled trials [14]. While computationally efficient, this approach neglects simultaneous multi-axis motion and is sensitive to speed variation.

Hypotenuse Method

To improve prediction accuracy, several researchers have proposed more sophisticated interception strategies. Xu et al. explored a hypotenuse-based method that considers the robot's diagonal trajectory, including vertical and horizontal motion components. Their approach solved a closed-form quadratic equation to estimate interception time and achieved a 28% improvement in average pickup accuracy over

simpler vertical descent models [15]. These findings highlight the benefits of accounting for actual motion paths, especially in applications requiring dynamic interception.

Ambush Method

The Ambush Method represents an alternative interception strategy where the robot positions itself ahead of the part's projected path and waits for the object to arrive at a fixed grasping position. Huang et al. applied this strategy to a conveyor-based sorting system and reported consistent timing alignment with moving parts at up to 50 mm/s, achieving successful pickup in 95% of trials under fixed velocity conditions [16]. However, this method requires precise calibration of timing and placement to ensure consistent alignment. Table 2 summarizes the key performance characteristics of each pickup method discussed above. The performance values listed are representative examples taken from selected studies in the literature and are not necessarily the best or worst reported values.

Table 2: Summary of Pickup Method Performance

Pickup Method	Performance	Notes
Direct	14.2 mm planar error	Tested by Zhang et al. using constant-velocity parts; used a top-mounted camera and controlled environment, no diagonal motion modeling [14].
Hypotenuse	28% improvement over vertical descent models	Xu et al. used a diagonal motion prediction with closed-form trajectory solutions with enhanced modeling [15].
Ambush	95% pickup success rate at 50 mm/s	Huang et al. applied the method to fixed-velocity parts on a controlled belt [16].

2.5 Influence of System Parameters on Accuracy

2.5.1 Conveyor Speed Effects

Beyond motion prediction, several operational parameters influence the performance of vision-guided robotic systems. Conveyor belt speed directly affects observation windows and introduces variability in object tracking. According to Chen et al., increasing belt speed from 30 mm/s to 90 mm/s resulted in a 37% rise in average positional error in a monocular vision tracking system [5]. This emphasizes the difficulty of real-time prediction under reduced observation intervals.

2.5.2 Scanning Height Impact

Camera scanning height also plays a significant role. Wang et al. evaluated object recognition at different mounting heights and found that lower heights (600 mm) yielded an average detection accuracy of 95.1%, compared to 84.3% at 800 mm due to reduced pixel resolution and part size in the image frame [6]. This highlights the trade-off between field of view and image quality.

2.5.3 Iterative Velocity Estimation

Another important factor is the number of iterations used for velocity estimation. Hu et al. showed that increasing the number of sampling iterations from 5 to 10 improved prediction accuracy by approximately 12%, but beyond 12 iterations, performance gains diminished due to increased lag and system delay [17]. This demonstrates the importance of balancing temporal sampling with system responsiveness.

2.5.4 Robotic Motion Constraints

Finally, robot-specific parameters, including speed and acceleration limits, constrain interception precision. Tang et al. investigated the impact of robot acceleration on trajectory following accuracy and observed that limiting acceleration to below 750 mm/s^2 resulted in missed pickups in 20% of dynamic trials [7]. These findings underline the importance of tuning motion constraints to match prediction models. Table 3 provides a consolidated overview of the system parameter effects described above.

Table 3: Summary of Key System Parameter Performance

System Parameter	Performance	Notes
Conveyor Belt Speed	37% increase in positional error when increasing speed from 30 to 90 mm/s [5]	Chen et al. used a monocular vision system on a testbed with slower robot response [5].
Scanning Height	Detection accuracy dropped from 95.1% at 600 mm to 84.3% at 800 mm [6]	Wang et al. used high-res cameras and controlled lighting [6].
Velocity Estimation Iterations	12% improvement from 5 to 10 iterations; diminishing returns after 12 [17]	Hu et al. implemented prediction in MATLAB with pre-processed frames, decreasing processing times [17].
Robot Motion Constraints	20% missed pickups when acceleration limited below 750 mm/s^2 [7]	Tang et al. tested robotic arms in high-speed industrial setups with external sensing and more complex part motion [7].

2.6 Summary and Identified Research Gap

While prior literature demonstrates that complex multi-sensor and multi-algorithmic systems can achieve exceptional performance in dynamic pick-and-place tasks, there is comparatively little work investigating the potential and limitations of minimalist configurations. This thesis aims to address that gap by evaluating a single-camera, end-mounted vision system integrated with an off-the-shelf six-axis robot. The results offer

insight into the practical trade-offs between system simplicity and predictive accuracy, particularly for small-scale manufacturers seeking affordable, modular automation solutions.

CHAPTER III: DESIGN METHODOLOGY

3.0 Research Objectives and Associated Methods

This study is guided by three primary research objectives, each designed to evaluate different aspects of the single-camera robotic pick-and-place system and to support broader conclusions about its industrial applicability. This section outlines each objective, followed by a clear description of the corresponding methodology used to address it.

Objective 1: Determine Maximum Achievable Accuracy

To address this objective, the system was subjected to a series of tests using three distinct predictive pickup strategies: the Direct Kinematic Method, the Hypotenuse Method, and the Ambush Method. Each method represents a different approach to predicting the part's future location based on visual tracking and robot motion planning. System accuracy was quantified by comparing the predicted pickup position and orientation of a moving part to its actual position and orientation at the time of attempted grasp. The evaluation metrics included positional error in the x and y axes (in millimeters) and rotational error θ (in degrees). Each method was tested in controlled conditions to identify the most accurate configuration.

Objective 2: Numerically Compare System Accuracy to Industry Standards

This objective involved comparing the accuracy achieved in experimental trials to performance benchmarks reported in relevant literature and industry data. The evaluation criteria included average positional errors (x and y), rotational errors (θ), and reliability across multiple trials (expressed as a success rate). Results from this study were analyzed

in relation to accuracy levels commonly found in more complex vision-guided systems, allowing for a practical comparison between low-cost and high-end solutions.

Objective 3: Identify Practical Limitations of the Single-Camera Setup

To explore the inherent limitations of this system, additional variables were tested independently. These included conveyor belt speed, scanning height, velocity estimation iteration count, and robot motion limits (maximum speed and acceleration). Each of these parameters represents a potential constraint in real-world deployment, and varying them helped uncover the boundaries of the system's reliable performance. Methodologically, this involved isolating one variable at a time, holding others constant, and measuring the resulting changes in positional and rotational accuracy.

With these objectives defined and the methodological alignment established, the following sections provide a detailed description of the system's design and the experimental procedures used in this study.

3.1 System Development and Architecture

The robotic platform developed for this study was designed to emulate a realistic industrial pick-and-place cell, but with a focus on simplicity, modularity, and cost-effectiveness. The goal was to assess the accuracy of a vision-guided robot system that uses only one camera mounted directly on the robotic end-effector—a configuration that dramatically reduces system complexity.

To maintain accessibility and reproducibility, only off-the-shelf hardware components were used. The robotic arm chosen was the Epson VT6, a six-axis industrial manipulator known for its compact size and sufficient precision for light assembly tasks [19]. This arm was outfitted with an OnRobot 2FG7 parallel gripper [20], selected for its mechanical reliability and suitability for small-to-medium parts. A 5.0 Megapixel Epson GigE industrial camera [21] was securely mounted to the front of the gripper, ensuring that all vision data was collected from the same point as the pickup action. This eliminated the need for external cameras or multiple coordinate transformations. System control and programming were managed through the Epson RC+ 7.0 development environment [22], which integrates both vision processing and motion programming. This platform was ideal for minimizing software overhead while maintaining full control of all system components, which are shown in Table 1 below, and the setup schematic is shown in Fig. 3.

Table 4: Setup Components

Device	Parameter
Robotic Arm	Six-axis
Camera	Monochromatic, 2560 x 1920 Resolution, 14 fps capture speed
End Effector: Gripper	Two finger parallel, 18 to 55 mm grip length
Control Software	Built-in manufacturer software in BASIC
Conveyor Belt	0 to 400 mm/s linear speed

Two main processes were developed within the software environment:

- VisionTask Function: Responsible for continuously capturing images of the conveyor and processing them to detect the part's position and orientation. The

vision tools included in Epson RC+ were used to compare each frame to a calibrated template of the part, allowing extraction of the part's center position in x and y, as well as its angular orientation (u).

- **RobotMotionControl Function:** This function controlled the motion of the robot based on visual input. It executed tracking, velocity estimation, and predictive calculation. The robot would move in synchronization with the part along the conveyor, maintaining a fixed scanning height until the pickup prediction was complete.

During operation, the robot iteratively collected position data from the VisionTask and stored it in dedicated arrays. By analyzing the time stamps between frames, it calculated instantaneous velocities in x and y. After a specified number of iterations, the robot computed the average velocity and used one of the three predictive methods to determine where the part would be by the time the robot reached it. To reduce early noise, the first three position readings in each trial were excluded from velocity averaging. This helped account for mechanical lag or error during initial robot movement. Importantly, the image processing was kept intentionally simple to reflect a realistic industrial scenario where high-performance systems may not be available. No machine learning or advanced computer vision algorithms were used. This ensured that the evaluation was focused solely on hardware constraints and basic predictive logic.

3.2 Experimental Testing Methodology

After constructing the physical and software infrastructure, testing proceeded to systematically evaluate the system's performance under a variety of controlled conditions. Each test was designed to isolate the effects of a specific variable while keeping others constant. The baseline configuration was selected based on practical observations during system development. This configuration offered reliable performance without being overly optimized. The key baseline values are summarized in Table 5. Each variable under study was varied independently while others were held at baseline. This allowed for clean attribution of performance changes to the variable being tested. A visual overview of the experimental setup is shown in Fig. 3., and Fig. 4. This includes the Epson VT6 robotic arm, the conveyor system, the gripper, and the end-mounted camera used throughout testing. Baseline values for each experimental variable were selected based on prior research findings, intuition, and practical engineering judgment. For most variables (such as conveyor speed, scanning height, velocity estimation iterations, and robot motion constraints), baseline settings were established before testing and were held constant throughout testing, even if subsequent results indicated that alternative settings might offer improved performance. However, for the pickup method variable, all methods were initially tested, and the Hypotenuse Method was selected as the baseline for subsequent experiments based on its superior performance relative to the other tested approaches. This selection helped ensure that the system's pickup strategy did not artificially limit the performance observed when varying other parameters.

Table 5: Key Variable Baseline Values

Key Variable	Baseline
Pickup Method	Hypotenuse
Iteration Count	9
Scanning Height	770 mm
Conveyor Speed	50 mm/s
Speed Limit	1000 mm/s
Acceleration Limit	1000 mm/s ²

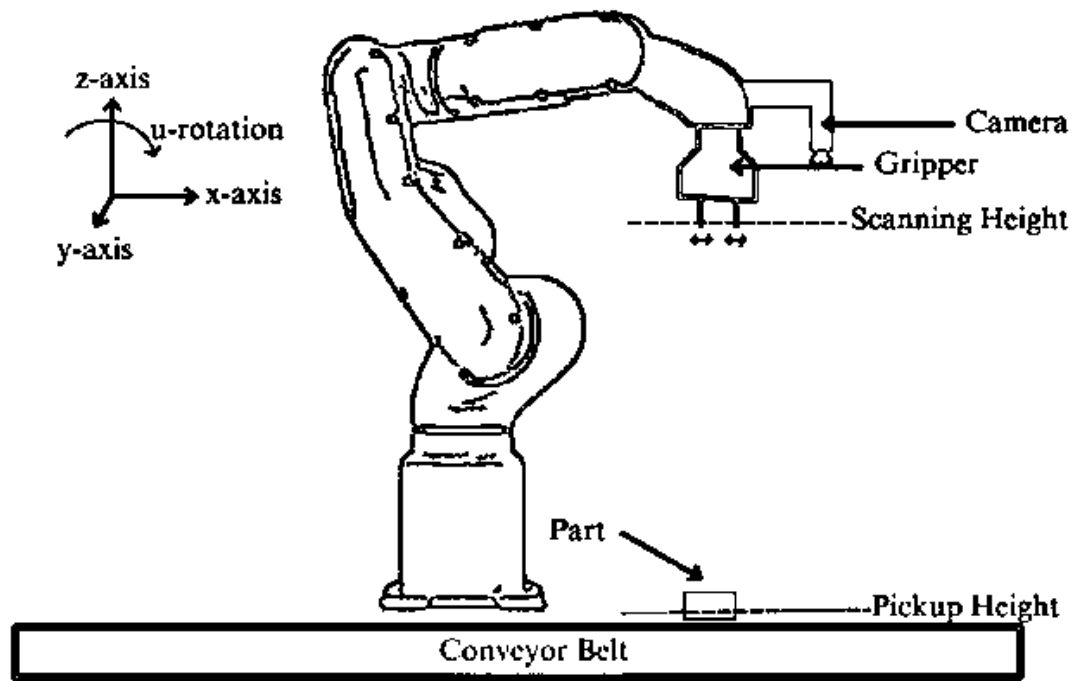


Figure 3: Diagram of Experimental Setup. Important components and variables are listed, such as the camera, gripper, part, conveyor belt, scanning and pickup height,

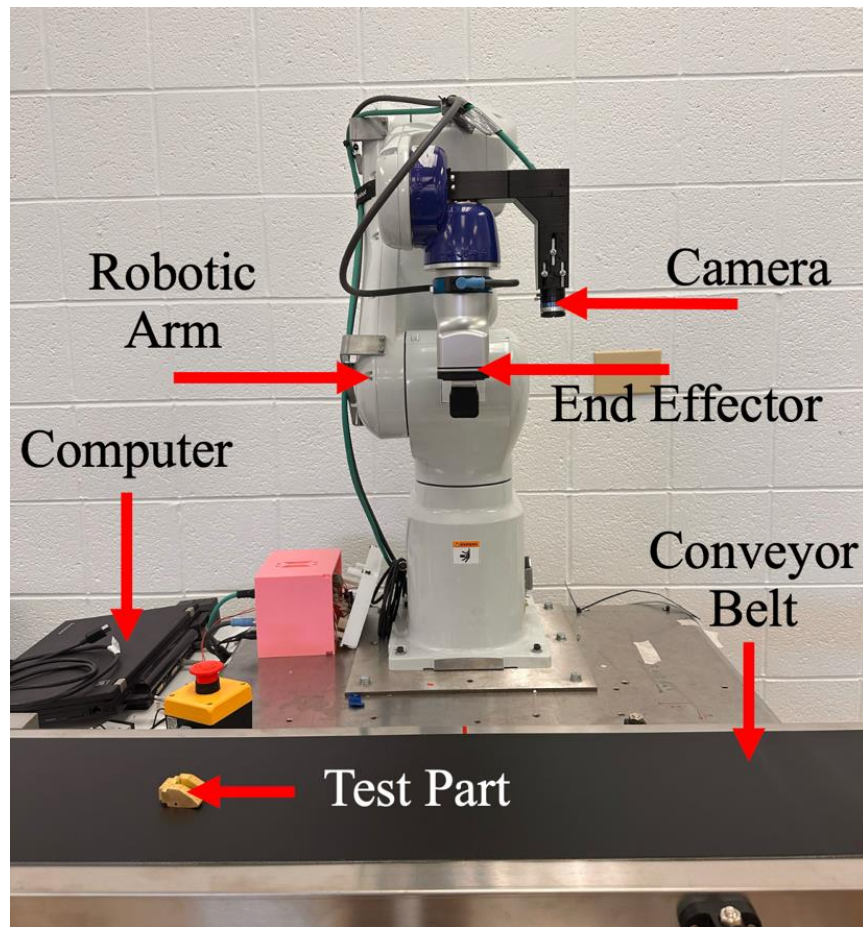


Figure 4: Experimental Setup. This photo shows the setup for all testing and development, all key components are shown.

3.2.1 Predictive Pickup Methods

Three pickup strategies were developed to compare how different motion assumptions affected system accuracy:

Method 1 – Direct Kinematic Approach:

The first method assumed that the primary motion limiting factor was the robot's vertical descent to the pickup height. The robot calculated the time required to traverse the

vertical distance, $H - H_{pickup}$, using known acceleration a , and maximum velocity R , following a trapezoidal motion profile:

$$T = \frac{2R}{a} + \frac{(H - H_{pickup})}{R} - \frac{R^2}{aR} \quad (1)$$

Where:

- H is the scanning height (shown in Fig. 3. as “Scanning Height”)
- H_{pickup} is the pickup height (shown in Fig. 3. as “Pickup Height”)
- R is the maximum robot velocity
- a is the robot’s maximum acceleration

This equation was derived using simple kinematic motion with the set acceleration and maximum velocity of the robot. The predicted pickup position in the x and y directions, x_{pred} and y_{pred} respectively, was calculated as:

$$x_{pred} = x_0 + v_x T, \quad y_{pred} = y_0 + v_y T \quad (2)$$

Where:

- x_0 / y_0 are the last known position of the part, as detected by the vision system
- v_x / v_y are the estimated velocities of the part in each direction
- T is the estimated time required for the end effector to reach the predicted pickup location.

Method 2 – Hypotenuse-Based Trajectory Model:

This refined method accounted for the fact that the robot moves along a diagonal (hypotenuse) path, rather than a purely vertical descent. The interception time T was calculated by solving the following quadratic:

$$\left(RT - \frac{R^2}{a}\right)^2 = H^2 + (x_0 + v_x T)^2 + (y_0 + v_y T)^2 \quad (3)$$

This equation was derived by modeling the robot's motion along the hypotenuse of a right triangle formed between the vertical scanning height offset and the predicted horizontal distance of the part. By treating the diagonal path as a single motion segment, the solution incorporates both spatial geometry and the trapezoidal motion profile described in Method 1 to estimate the total travel time. The quadratic was solved for T , yielding a more precise estimate of the interception time considering the true 3D path traveled.

$$T = \frac{-(2Bx_0v_x + 2By_0v_y) + \sqrt{(2Bx_0v_x + 2By_0v_y)^2 - 4(B^2 - v_x^2 - v_y^2)(A^2 - H^2 - x_0^2 - y_0^2)}}{2(B^2 - v_x^2 - v_y^2)} \quad (4)$$

Method 3 – Ambush (Wait-at-Location) Strategy:

The third method altered the pickup behavior entirely. Instead of predicting a moving interception point, the robot moved directly to a pre-determined ambush distance, x_{ambush} and y_{ambush} , ahead of the part's trajectory. Upon arrival, the robot waited until the part arrived under the gripper, then executed the pickup:

$$x_{ambush} = x_0 + v_x T_{fixed}, \quad y_{ambush} = y_0 + v_y T_{fixed} \quad (5)$$

Where T_{fixed} was a predefined wait time optimized for system response. This approach minimized error introduced by incorrect velocity predictions or sudden variations in part speed.

3.2.2 Additional Variable Changes

Beyond pickup strategies, other variables were tested to assess practical system limitations:

- Iteration Count: Number of scans used for velocity averaging. Tested from 8 to 14 iterations. Upper and lower limits were chosen due to performance drop off—lower values led to high noise; higher values increased delay and risk of missing the part.
- Scanning Height: Distance between the camera and conveyor. Tested from 650 mm to 800 mm in 30 mm increments. Lower heights improved image quality but shortened observation window. Higher heights offered more coverage but less detail.
- Conveyor Speed: Set between ~5% and ~20% of max speed. Higher speeds made it difficult for the robot to intercept. Lower speeds caused motor instability.
- Robot Motion Limits: Robot max speed and acceleration were varied from 500 to 2000 mm/s and mm/s², in increments of 500. At low values, the robot could not reach the pickup point in time. At high values, system accuracy decreased due to

mechanical limitations. Each of these variables was tested using the same pickup protocol described in the next section.

3.2.3 General Testing Procedure

Each trial began by placing a test part on the conveyor at a randomized orientation. The system began tracking the part using its vision tools and performed several scan iterations to collect position and velocity data. Once enough data was collected, the robot computed a predicted pickup point using the selected method. It then moved to this location and executed the pickup maneuver. The predicted pickup position for each trial was generated by the vision-processing algorithm and displayed to the user through the program's interface. The x, y, and u coordinates provided by the system were recorded. At the moment of pickup, the conveyor was stopped, and the actual part position was recorded. To record the actual part position, the robot was manually jogged using the Epson RC+ interface until the center of the gripper was visually aligned with the center of the part. Once aligned, the robot's displayed position values were recorded directly from the control panel, providing the ground truth x, y, and u coordinates for each trial. Positional accuracy was defined as the absolute difference between predicted and actual part locations in x and y (in mm). Rotational accuracy was the angular error in u (in degrees). Each configuration was tested over 10 trials. Data was averaged, and standard deviations were calculated to assess consistency. Average data represents the average of the x and y positional errors only. Rotational error (u) is excluded from this calculation to ensure consistent unit handling and reflect pure displacement accuracy. This procedure was repeated for every

pickup method and every variable range. This ensured a fair and consistent comparison across all test conditions.

3.2.4 Data Processing and Outlier Removal

During the experimental testing described above, several individual trials across multiple configurations produced anomalous data points that deviated significantly from the rest of the collected values. These outliers—characterized by unusually high or low error measurements—risked skewing the averages, variances, and graphical representations of system accuracy. To improve the clarity, consistency, and interpretability of the results, an outlier detection and removal process was implemented prior to final data analysis. The method selected for outlier identification was the box plot method, a statistical approach commonly used for identifying outliers in small or non-normally distributed datasets. A box plot distributes the data based on quarters. The key component used in this analysis is the interquartile range (IQR), which is calculated as the difference between the third quartile (Q3) and the first quartile (Q1). This range captures the middle 50% of the data. Any point that lies below $Q1 - 1.5 \times \text{IQR}$ or above $Q3 + 1.5 \times \text{IQR}$ is flagged as a potential outlier. This method is particularly effective for the dataset used in this study, where each configuration included only 10 trials and the data distributions could not be assumed to be Gaussian. Across the full dataset of 230 trials, 47 trials were identified and excluded as outliers based on the established criteria. This corresponds to approximately 2.25 outliers per configuration, with each configuration consisting of 10 trials. While some configurations exhibited no outliers, most configurations saw between two and three outliers removed. For each tested

configuration—specifically, each pickup method, each value of scanning height, conveyor belt speed, velocity estimation iteration count, and robot motion constraints—the full set of ten trial results was first copied into a new analysis sheet to preserve the raw data (available in the Appendix). The box plot method was then applied independently to the x, y, and u error values associated with each configuration. Any data point falling outside the calculated IQR bounds for a given error axis was excluded from final calculations of average, minimum, maximum, and standard deviation, as well as the plots. This allowed the analysis to focus on consistent system behavior while minimizing the influence of rare anomalies.

Additionally, to compare the error across variables with different units (such as distance in millimeters and rotation in degrees), a normalized error metric was employed. The normalized error is calculated by dividing the error in each direction (x, y, and u) by the maximum observed error value for that specific variable across all tests. This results in a unitless value between 0 and 1, enabling direct comparison. The normalized error for each test case is then computed as the average of the normalized x, y, and u errors. This approach is necessary because averaging errors with differing units directly would yield misleading results. Normalizing each error by its maximum observed value allows for the aggregation of these metrics while maintaining proportionality and comparability. The comparison of normalized error between test configurations was important for evaluating the overall accuracy of the system, as necessary for Objective 1.

CHAPTER IV: RESULTS

4.0 Results by Research Objectives

This chapter presents the results of the experimental evaluation of the vision-guided robotic system, organized around the three primary research objectives outlined in Chapter I and restated in Chapter III. This structure is intended to ensure that the experimental findings are interpreted in the context of the original goals of the study. Each section is dedicated to one objective and includes the data, observations, and analysis that directly address the corresponding research question. It is important to note that no data was deleted from the project archive. All original values, including those classified as outliers, have been retained and are included in the appendix. The results presented in Chapter IV are based exclusively on the filtered dataset, with all statistical metrics (including average, range, and standard deviation) calculated from the outlier-excluded trials.

The first research objective was to determine the maximum achievable accuracy of the system using only a single, end-mounted camera as the sole sensor. This objective was pursued by evaluating the performance of three unique predictive pickup methodologies under controlled conditions. Each method—Direct Kinematic, Hypotenuse, and Ambush—was subjected to a consistent testing protocol, and the accuracy was quantified using three key metrics: the difference between the predicted and actual position in the x direction, the difference in the y direction, and the rotational error θ in degrees. These results are thoroughly analyzed in Section 4.1. By examining each of these predictive approaches under identical conditions, this objective aimed to establish a baseline understanding of the system's optimal performance capabilities, while also highlighting the relative strengths and weaknesses of each motion model.

The second objective was to compare the system’s accuracy with relevant industry benchmarks and published results from other vision-guided robotic systems. While this study does not aim to create a production-ready system that competes directly with advanced multi-sensor robotic platforms, a key goal was to assess how a simplified setup performs in relation to the broader landscape of industrial automation. Section 4.2 explores this objective by drawing comparisons between the positional and angular accuracy values achieved during testing and the typical tolerances or performance metrics reported in existing literature. These comparisons are intended to contextualize the system’s performance, not as a direct competitor to high-end systems, but as a cost-effective solution for environments where ultra-high precision may not be necessary, and where flexibility, affordability, and ease of deployment are prioritized.

The third and final objective was to identify practical limitations of a single-camera vision-guided robotic system under realistic dynamic conditions. To accomplish this, a series of additional tests were conducted to assess the impact of various independent system parameters. These variables—scanning height, conveyor belt speed, the number of tracking iterations, and the robot’s acceleration and velocity limits—were methodically varied, with all other conditions held constant. The goal was to isolate each variable’s contribution to the overall accuracy of the system and, in doing so, reveal the most critical constraints imposed by the simplified design. Section 4.3 presents these results, focusing specifically on observed performance drops, failure modes, and outlier behavior. This analysis also identifies which parameters were most sensitive to change and which ones offered the most improvement when tuned correctly.

Across all experiments, a consistent evaluation framework was applied. Each trial began with the robot tracking a part moving at a steady speed along a linear conveyor belt, calculating the part's instantaneous velocity using a fixed number of scan iterations, and then executing one of the three pickup strategies. The predicted pickup location—both in terms of position and orientation—was then compared to the actual location of the part at the time of the attempted grasp. Absolute errors were recorded for each trial and aggregated across multiple runs to produce meaningful average values, standard deviations, and observed ranges. In addition to these quantitative results, notable failure modes and edge-case behaviors were recorded, including trials where the robot failed to complete the pickup due to incorrect predictions or timing misalignments.

By organizing the results around the core research objectives, this chapter ensures that each experimental outcome directly contributes to answering the overarching research questions posed in this study. The following sections detail these findings, beginning with the evaluation of the three predictive pickup methods and their corresponding performance metrics under baseline conditions.

4.1 Maximum Achievable Accuracy

This section presents the measured performance of the robotic system across the three tested predictive pickup methodologies: the Direct Kinematic Method, the Hypotenuse-Based Trajectory method, and the Ambush Method. All three methods were evaluated under identical baseline conditions, which included a fixed scanning height of 770 mm, a conveyor belt speed of 50 mm/s, a robot velocity limit of 1000 mm/s, an acceleration limit of 1000 mm/s², and a velocity estimation iteration count of 9. These

parameters were chosen based on iterative refinement during the development phase and represent a stable configuration for evaluating and comparing predictive pickup accuracy across all three strategies. Each method was subjected to ten independent trials. For each trial, the part's actual position at the time of the robotic gripper's attempted pickup was compared to the system's predicted location. The system's prediction was compared against the actual observed location of the part to determine the error in three components: linear displacement in the x and y directions, and angular displacement about the u-axis. These three values represent the difference between the predicted and actual position and orientation of the part at the time of pickup. Figure 5 provides a visual representation of these error components, illustrating a top-down view of the conveyor belt system. As the part moves along the belt, the vision system predicts its future location and orientation. The diagram highlights the discrepancy between this predicted location and the part's actual position when the robot attempts to pick it up, with each error component labeled accordingly. In the error visualization shown in Fig. 5, the error components are denoted as E_x , E_y , and E_u , representing the errors in the x direction, y direction, and rotational orientation, respectively. Moving forward in the results, these components are collectively referred to as 'displacement inaccuracy,' where E_x corresponds to error in x, E_y to error in y, and E_u to rotational error. These variables are plotted as the dependent variables on the vertical axes of the subsequent results graphs, consistently reflecting the measured deviation between predicted and actual pickup locations.

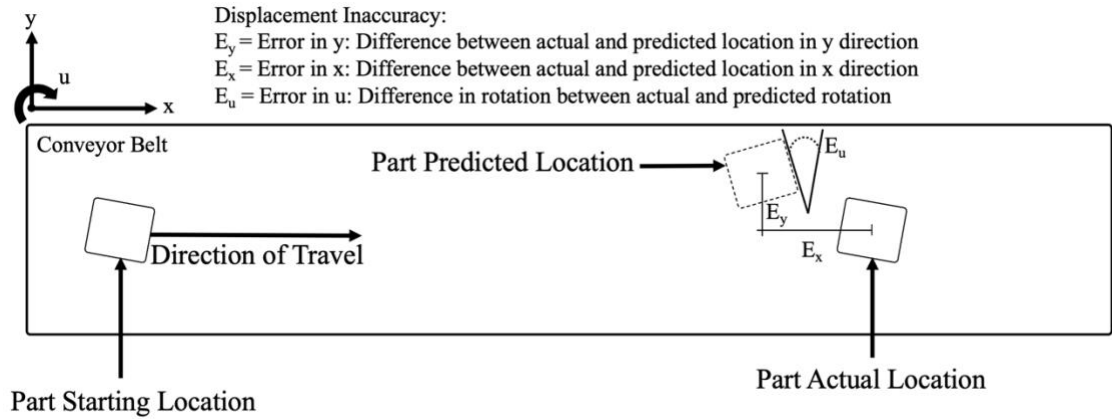


Figure 5: Error Visualization Diagram. This diagram shows a top-down view of the conveyor belt, with the part movement and predicted/actual locations. The error, or difference between these two locations, is depicted as well.

4.1.1 Direct Kinematic Method

The Direct Kinematic Method calculated interception time based on the robot's vertical descent to the pickup plane, assuming trapezoidal motion and constant part velocity. This method, being the simplest of the three, served as a baseline model and did not take into account the robot's full diagonal travel path or its multi-axis motion profile. Instead, the time to intercept was computed solely from the known scanning height and the robot's speed and acceleration limits, using a vertical displacement equation.

In terms of x direction positional accuracy, the Direct Kinematic Method produced an average error of 40.11 mm. The minimum error recorded across trials was 31.90 mm, while the maximum reached 48.00 mm, indicating considerable variability in prediction accuracy. The high error values in x suggest that relying solely on vertical descent time for predictive modeling is insufficient for accurate interception, particularly because the horizontal motion component of the robot's trajectory is ignored. The y direction positional

accuracy, however, was better. The average error in y was 30.14 mm, with recorded errors ranging from 20.80 mm to 41.00 mm.

Rotationally, the system achieved an average angular error of 6.77° using this method. The lowest recorded error was 5.40° , while the maximum reached 8.10° . Although this method did not include any explicit strategy for minimizing rotational error beyond standard vision-based alignment, the rotational results were reasonably consistent, albeit slightly higher than those recorded for the Hypotenuse Method.

The average error across x and y only, excluding rotational error, was 35.13 mm. Despite its simplicity and limitations, the method was able to successfully complete all ten trials without failure and maintained a consistent performance profile, with an average deviation of 2.27 mm.

4.1.2 Hypotenuse-Based Trajectory Method

The Hypotenuse Method estimated interception time by considering the full diagonal travel path between the scanning height and the predicted pickup location. This method modeled the motion along the hypotenuse of a right triangle formed between the vertical scanning offset and the planar distance from the camera to the part. By using the robot's velocity and acceleration parameters in conjunction with the predicted position, the required time to intercept was calculated using a trapezoidal motion profile.

This method significantly improved positional accuracy compared to the Direct Kinematic Method. The average error in the x direction was 14.57 mm, with values ranging from 6.40 mm to 22.90 mm. In the y direction, the average error was 31.46 mm, with a minimum of 26.60 mm and a maximum of 37.60 mm. While the y error remained relatively

high, the x error was substantially reduced, suggesting that the Hypotenuse Method's improved prediction model more accurately accounted for the robot's actual motion characteristics.

The average angular error was 5.91° , with a minimum of 5.10° and a maximum of 6.60° , making it the best-performing method rotationally. The system maintained consistent alignment performance without requiring separate rotational modeling or compensation. The average positional error in the x and y directions was 23.01 mm, making this method more accurate than the previous. All ten trials were completed successfully, and the results showed a relatively tight spread, with an average deviation of 2.07 mm.

4.1.3 Ambush Method

The Ambush Method used a static interception time set slightly shorter than the average travel time predicted by the Hypotenuse Method. Instead of calculating a precise moment of interception, this method relied on triggering the pickup action earlier than expected, allowing the robot to “wait” for the part to arrive. This approach avoided motion prediction altogether and instead prioritized timing safety, which is sometimes favored in high-speed systems with variable part trajectories.

The positional accuracy in the x direction was lower than the other methods, with an average error of 43.28 mm and a range between 29.30 mm and 58.80 mm. The y direction error was lower than x, with an average of 21.12 mm and a range from 14.30 mm to 27.90 mm. This imbalance indicates that the method may have performed better at estimating lateral position due to timing alignment but suffered along the robot's primary movement axis.

The average angular error was 6.56° , with a minimum of 5.90° and a maximum of 7.30° , comparable to the other two methods. Although the system did not rely on real-time motion modeling, the part's consistent placement and conveyor speed helped maintain rotational alignment within a tolerable range.

The average positional error across x and y was 32.20 mm, placing this method between the Hypotenuse and Direct Kinematic methods in terms of accuracy. All ten trials were successful, and the average deviation was 3.24 mm, slightly higher than the other methods and reflecting a broader spread in performance.

4.1.4 Comparative Summary of Pickup Methods

To allow a consolidated view of each method's performance, Table 6 and Fig. 6. summarize the average error values recorded across the three tested predictive strategies. In Table 6 and Fig. 6., the Average data represents the average of the x and y positional errors only. Rotational error (u) is excluded from this calculation to ensure consistent unit handling and reflect pure displacement accuracy. Detection rate indicates how many of the ten trials resulted in successful part identification by the vision system. While detection was successful in all trials for pickup method testing, later experiments occasionally experienced detection failures.

Table 6: Component Error and Detection

Method	Average X Error (mm)	Average Y Error (mm)	Average U Error ($^\circ$)	Average (mm)	Detection Rate
Kinematic	40.11	30.14	6.77	35.13	10/10
Hypotenuse	14.57	31.46	5.91	23.01	10/10
Ambush	43.28	21.12	6.56	32.20	10/10

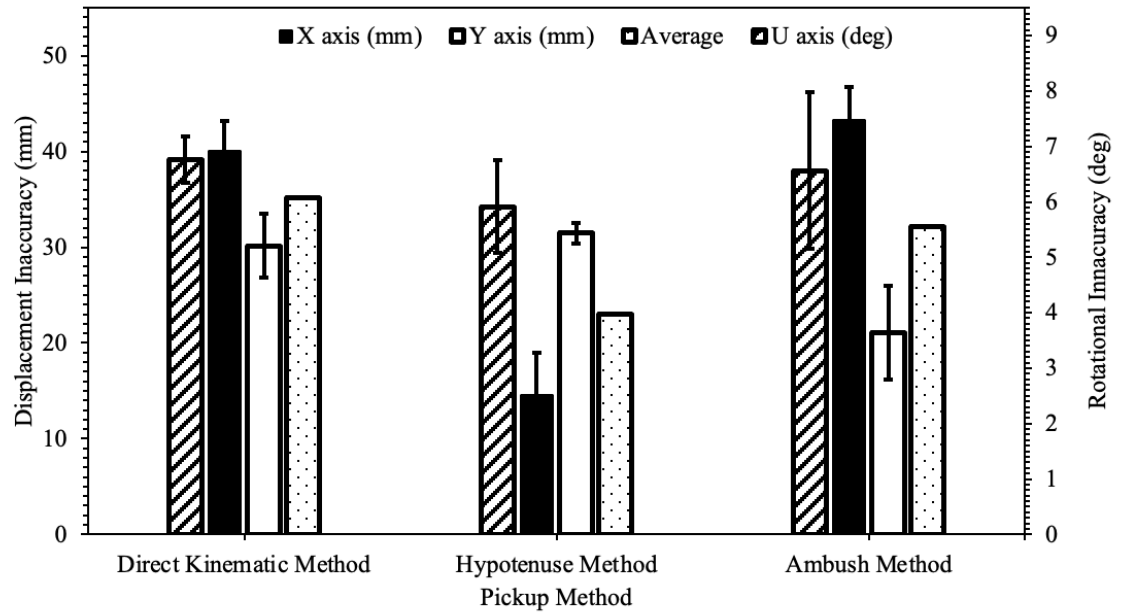


Figure 6: Displacement Inaccuracy vs. Pickup Method

The data demonstrates clear trade-offs between the methods. The Hypotenuse Method offered the best accuracy in the x direction and rotational alignment but performed the worst in y direction positioning. The Ambush Method achieved the best average y accuracy and maintained a moderate x error, though it suffered in rotational precision. The Direct Kinematic Method, although the least sophisticated and most limited in terms of predictive modeling, offered reasonably balanced performance in y and u, but exhibited the largest x direction error. All three methods could complete all ten trials without failure, demonstrating a fundamental level of robustness and stability under consistent experimental conditions. However, the variation in performance metrics reveals that the

choice of predictive method significantly influences both accuracy and consistency, depending on the axis of interest and the specific goals of the application.

Beyond just the average error values, the standard deviation of each error component adds an important dimension to the system's characterization—namely, how reliably each method performs from trial to trial. A method with a low average error but high variability may not be desirable in situations requiring consistent output. Conversely, a method with moderate average error but tight consistency might be better suited for operations where predictability and repeatability are critical.

The Direct Kinematic Method exhibited the lowest standard deviation in x error, measured at 3.06 mm. This is notable given that its average x error was the highest of the three methods, indicating that while it consistently missed the target position in the x-direction, it did so in a predictable manner. Additionally, its y error standard deviation was 3.32 mm, which falls within a reasonable range, and its u error standard deviation of 0.42° was the lowest among the methods. This indicates that the method was particularly strong in repeatable rotational positioning, which is impressive considering that it lacked any sophisticated rotational optimization.

The Hypotenuse Method, while the most accurate overall in terms of average x and u errors, showed slightly higher variability in its x direction results with a standard deviation of 4.35 mm. Still, this value reflects consistent x position predictions across all trials. More impressively, this method demonstrated the lowest y error standard deviation at 1.03 mm, revealing an ability to maintain extremely tight lateral positioning accuracy even though its average y-error was higher than desired. In rotational accuracy, the method also performed well, with a u error standard deviation of 0.84° , confirming relatively stable

orientation alignment throughout the testing sequence. These characteristics make the Hypotenuse Method not only highly accurate but also consistent in two of the three tested dimensions.

The Ambush Method demonstrated mixed results. It had a moderate x error standard deviation of 3.41 mm, which is better than the Hypotenuse Method but slightly worse than the Direct Kinematic Method. However, its y error standard deviation was the highest at 4.90 mm, suggesting that its excellent average y accuracy (only 0.63 mm) came with a trade-off in trial-to-trial repeatability. This method also showed the highest standard deviation in u, at 1.41° , reflecting greater inconsistency in achieving precise gripper orientation during the pickup.

Together, these standard deviation metrics offer a more comprehensive view of how each method performs—not only in terms of hitting the target on average, but also how reliably that performance is repeated across multiple trials. The Direct Kinematic Method stands out for its consistent but misaligned x and u predictions. The Hypotenuse Method provides strong all-around consistency, especially in lateral and rotational dimensions, with some systematic offset in y. The Ambush Method, while simple and strong in average y performance, introduces more unpredictability, particularly in rotational alignment.

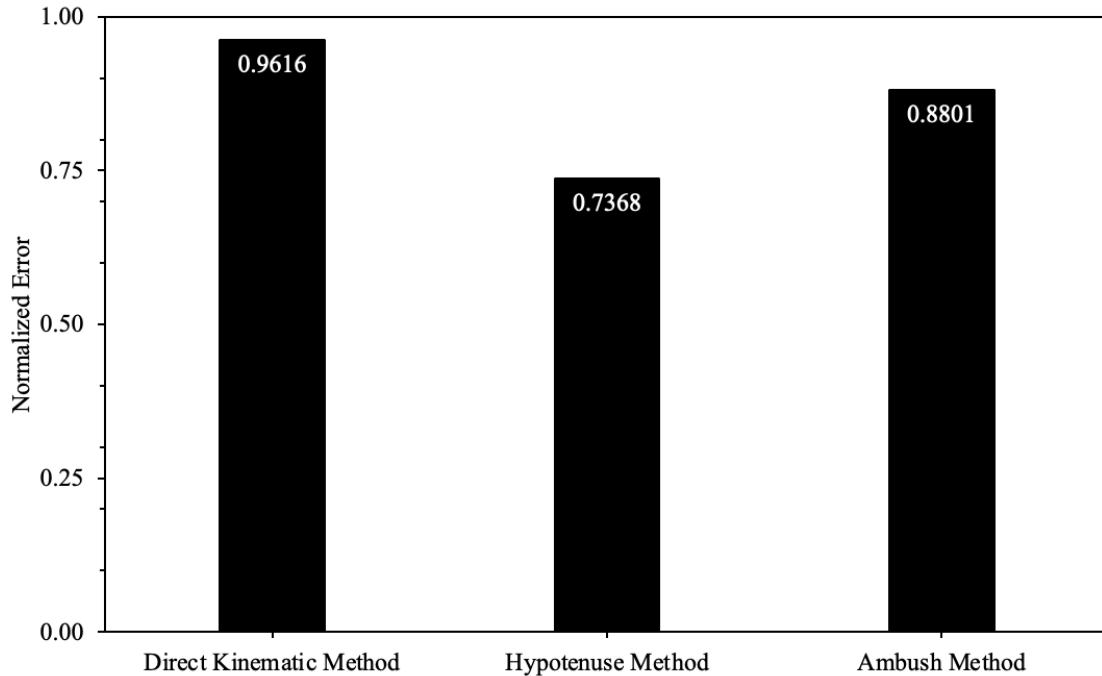


Figure 7: Normalized Error for the Pickup Methods

In addition to evaluating each individual axis, a normalized error analysis was performed to enable a unitless comparison across all three motion components, x, y, and rotational error (u). Each error value was normalized by dividing it by the maximum observed error for that axis across all three methods. The normalized x, y, and u values were then averaged to yield a single composite score between 0 and 1 for each method, where lower values represent better overall performance.

As shown in Fig. 7., the Hypotenuse Method achieved the lowest normalized error at 0.7368, followed by the Ambush Method at 0.8801, and the Direct Kinematic Method at 0.9616. These results reinforce the earlier findings that the Hypotenuse Method offers the most balanced and consistently accurate performance when considering all three axes of positioning.

4.2 Comparison of System Accuracy to Industry Standards

To evaluate the viability of a vision-guided, single-camera robotic system in a dynamic pick-and-place setting, the accuracy data collected through experimentation must be analyzed in the context of established industry standards. This section compares the system's performance across its three primary pickup configurations—focusing on positional accuracy in the x and y directions, as well as rotational accuracy—against commonly cited benchmarks for commercial robotic arms used in similar pick-and-place tasks.

In high-performance industrial systems employing external sensor networks and custom calibration routines, average positional accuracy in planar x-y space is often maintained within ± 1 mm to ± 5 mm under ideal conditions. For rotational accuracy, these systems typically demonstrate deviations of less than $\pm 1^\circ$ in U-axis orientation when operating in structured environments with high-resolution 3D cameras or external feedback systems [1], [2].

By contrast, the simplified system presented in this study used only a single end-effector-mounted camera without external sensing, calibration grids, or 3D feedback. Under these constraints, the most accurate configuration tested—using the Hypotenuse Method—produced an average positional error of 14.57 mm in the X-direction and 31.46 mm in the Y-direction. The rotational error for this configuration was 5.91° .

These results represent a clear deviation from industry best-in-class standards. Specifically, the x direction error exceeds the upper threshold of industry-grade planar accuracy by nearly 10 mm. The y direction error is even more substantial, overshooting the ± 5 mm threshold by more than 26 mm. In rotational alignment, the system's 5.91° average

deviation is considerably larger than the sub-degree tolerance generally expected in precision-driven tasks. This indicates that while the system can effectively track and intercept moving parts, it lacks the fine-grained accuracy necessary for ultra-precise applications such as electronics assembly or high-speed sorting where part misalignment of even a few millimeters can cause a failure in downstream operations.

Still, when placed in the context of mid-tier industrial systems or low-cost automation setups, these figures are not without merit. For example, some commercially available collaborative robots intended for flexible assembly lines advertise positional repeatability in the range of ± 0.1 mm to ± 1 mm, but these numbers are often quoted under static conditions without real-time vision tracking or dynamic part movement. When dynamic factors are introduced—such as part velocity, single-camera tracking delays, and robotic motion lag—systems operating with positional errors between 10 mm and 30 mm may still be considered viable for operations where exact placement is not critical, such as bin picking, loose sorting, or basic part transfer [2], [8].

When comparing all three methods, the Direct Kinematic Method had an average x error of 40.11 mm, a y error of 30.14 mm, and a u error of 6.77° . The Ambush Method produced a somewhat more balanced profile: an x error of 43.28 mm, a y error of 21.12 mm, and a u error of 6.56° . Based on the combined average positional error (x and y), the Direct Kinematic Method had a total error of 35.13 mm, the Ambush Method measured 32.20 mm, and the Hypotenuse Method achieved the lowest at 23.01 mm.

To further support this comparison, a normalized error metric was also applied to account for differences in units across x, y, and u axes. The Hypotenuse Method again

performed best, with the lowest normalized error of 0.7368, reinforcing its advantage as the most consistently accurate configuration overall.

These results reveal that while none of the methods met the stringent thresholds of high-end industrial accuracy, they do exhibit operational viability under less demanding conditions.

Rotationally, the system underperformed relative to industry standards in every configuration. With α errors consistently ranging between 5.91° and 6.77° , the gap remains substantial when compared to the $<1^\circ$ angular tolerance expected in high-precision orientation tasks. The best result— 5.91° from the Hypotenuse Method—still falls well outside the range required for many modern robotic applications that demand strict alignment of components.

While raw positional accuracy was the central metric under review, it is important to acknowledge that high standard deviation in any method would imply greater unpredictability in the system's real-world performance. For instance, although the Hypotenuse Method had the best average error values overall, its relatively modest standard deviations—4.35 mm in x and 0.84° in α —suggest consistent and reliable performance within those boundaries. By contrast, the Direct Kinematic and Ambush Methods exhibited higher standard deviations in x , making them less predictable even in scenarios where their mean accuracy might seem acceptable.

In summary, the results demonstrate that under ideal conditions, the system is capable of moderately consistent performance in loosely constrained scenarios. In practical terms, this could translate to successful deployment in small-scale operations where placement precision is less critical—such as sorting similarly sized objects, removing parts

from conveyor belts for quality inspection, or assisting human operators in semi-structured environments [8], [3], [4]. However, the clear margin by which these results fall short of industrial standards reaffirms that this type of minimalist single-camera system may be best suited for low-cost, low-precision automation environments. Table 7 summarizes the comparison between the best results achieved in this study and typical industry standards, [1], [2], [8].

Table 7: Standards Comparison

Metric	Best Result	Industry Standard	Comparison
X Error	14.57 mm	5 mm	~2.7x higher
Y Error	21.12 mm	5 mm	Within range
Rotational Error	5.91°	1°	~5x higher

4.3 Practical Limitations Identified

This section addresses the third primary research objective: to identify practical limitations inherent to a vision-based robotic system utilizing a single, end-mounted camera. Each experimental variable—scanning height, conveyor belt speed, velocity estimation iteration count, and robot motion constraints—was tested independently while holding other system parameters at baseline to isolate and quantify the impact of that variable on pickup accuracy. The metrics evaluated in each case were absolute positional error in the x and y directions (in millimeters) and rotational error in ψ (in degrees). These errors reflect the deviation between the system’s predicted pickup location and the actual location of the part at the time of interception. The practical limitations discussed below are based solely on the outcomes of the recorded trials.

4.3.1 Scanning Height

The scanning height—the vertical distance between the mounted camera and the conveyor belt—was varied across six configurations: 650 mm, 680 mm, 710 mm, 740 mm, 770 mm (baseline), and 800 mm. Changes in scanning height impacted both image resolution and the size of the observable field, and consequently influenced the reliability of part detection, velocity estimation, and final pickup accuracy.

At the lowest tested height of 650 mm, the system completed 8 out of 10 trials successfully. The x direction error averaged 43.15 mm, with a minimum of 4.04 mm and a maximum of 157.49 mm—the highest recorded in any scanning height configuration. Error in y was more stable, averaging 4.13 mm with a range from 0 mm to 10.87 mm, while rotational error averaged 8.37° , ranging from 6.71° to 9.65° . The average positional error in x and y was 23.64 mm. These results indicate inconsistent positional tracking at close range.

At 680 mm, the success rate improved to 10 out of 10, with an average x error of 38.95 mm and bounds between 18.24 mm and 65.39 mm. y error also increased slightly to 7.12 mm (range: 0 mm to 23.82 mm), while u error averaged 7.58° , showing improved rotational stability over the previous height. The combined x and y error was 23.04 mm. Although x error remained relatively high, the narrower rotational range indicated more consistent angle alignment.

The 710 mm height yielded stronger results, again with 10/10 successful trials. Error in x fell to an average of 16.45 mm, with a minimum of 0 mm and a maximum of 29.37 mm. Y error showed a marked improvement, dropping to 0.08 mm on average and

remaining within 0.00 mm to 0.30 mm. The error in u averaged 7.42° , ranging from 7.14° to 7.65° . The average positional error in x and y was 8.26 mm. These results reflected one of the most consistent and accurate sets, particularly in y direction positioning.

At 740 mm, performance continued to improve across all metrics. The x error averaged 21.42 mm, ranging between 0.00 mm and 42.72 mm. y error remained relatively low at 2.85 mm, with a range from 0.00 mm to 6.86 mm. The u error averaged 7.12° , with a minimum of 4.83° and a maximum of 9.74° . The combined x and y error was 12.13 mm. This configuration provided one of the most balanced performances, demonstrating low deviation across all parameters.

The 770 mm scanning height—the baseline configuration—delivered a strong and reliable set of results. All 10 trials were successful, and the x error averaged 14.57 mm, ranging from 2.60 mm to 26.10 mm. The error in y , however, increased significantly to 31.46 mm, ranging from 27.00 mm to 33.40 mm, the highest average y error across all configurations. Despite the higher y deviation, rotational accuracy improved with a u error average of 5.91° , ranging from 2.90° to 8.10° , placing it among the most consistent rotational performers. The combined average error in x and y was 23.01 mm.

At the maximum tested height of 800 mm, only 1 out of 10 trials succeeded. In that singular successful trial, the recorded x error was 62.13 mm, y error was 8.74 mm, and u error was 7.11° . The 90% failure rate at this height clearly illustrates the system's practical detection limits when image resolution falls too low to reliably identify the part's profile. This marked drop in success strongly suggests the highest viable scanning height lies below 800 mm. The average positional error in x and y for the successful trial was 35.44 mm.

The standard deviation values provide additional insight into the variability of each configuration. The highest x direction variability occurred at 650 mm, with a standard deviation of 4.04 mm, despite its poor average accuracy. In contrast, 770 mm showed the most stable x error at 2.18 mm, highlighting consistency despite increased y error. For y error, the 770 mm configuration showed the greatest variability with a standard deviation of 1.03 mm, while the 710 mm configuration displayed near-perfect stability with a y standard deviation of just 0.06 mm. U error was most consistent at 650 mm (0.42°) and least consistent at 740 mm (3.42°). These patterns reflect how scanning height adjustments affected not only average accuracy but also the repeatability of the system's performance across trials. Figure 8 summarizes the observed displacement inaccuracies across all scanning height configurations tested.

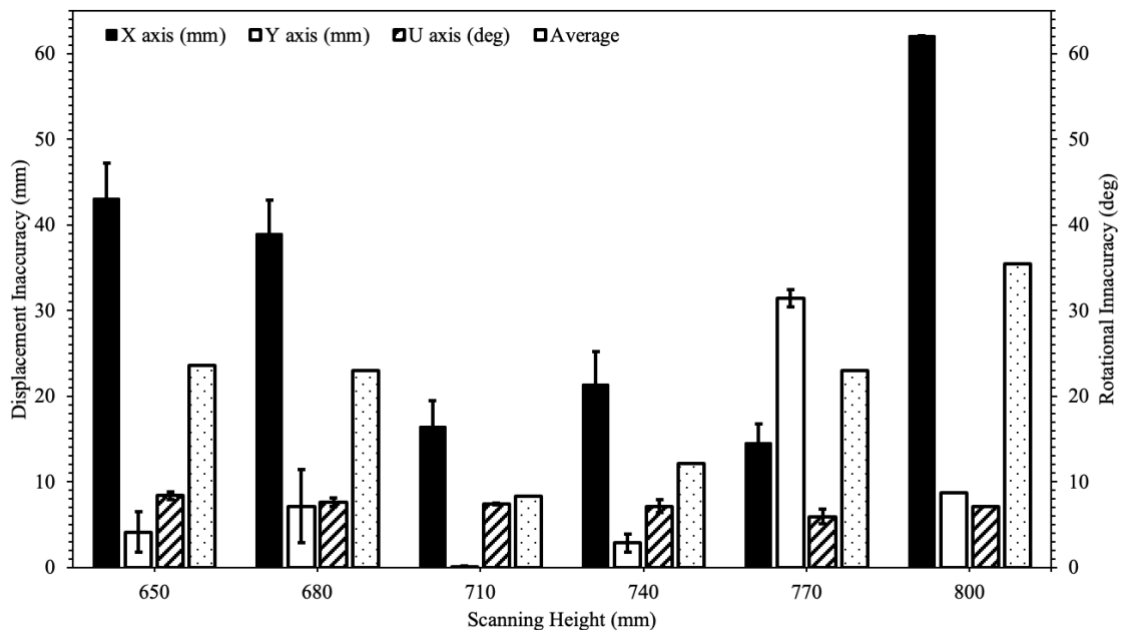


Figure 8: Displacement Inaccuracy vs. Scanning Height

To better evaluate the relative accuracy of each scanning height across both spatial and rotational axes, a normalized error analysis was performed. This method compensates for differences in unit scale between displacement (mm) and rotation (degrees) by normalizing each error axis relative to the maximum observed value and averaging the results. The outcome is a unitless score between 0 and 1, where lower values indicate better overall performance.

As shown in Fig. 9., the lowest normalized error occurred at 710 mm (0.3845), followed closely by 740 mm (0.4288), reinforcing earlier conclusions that these heights offered the most balanced and consistent performance. The highest normalized error was observed at 800 mm (0.7091), confirming the poor effectiveness at extreme scanning distances. Interestingly, the baseline configuration at 770 mm yielded a relatively high normalized error of 0.6471 despite its strong rotational accuracy, due to its large y-direction deviation.

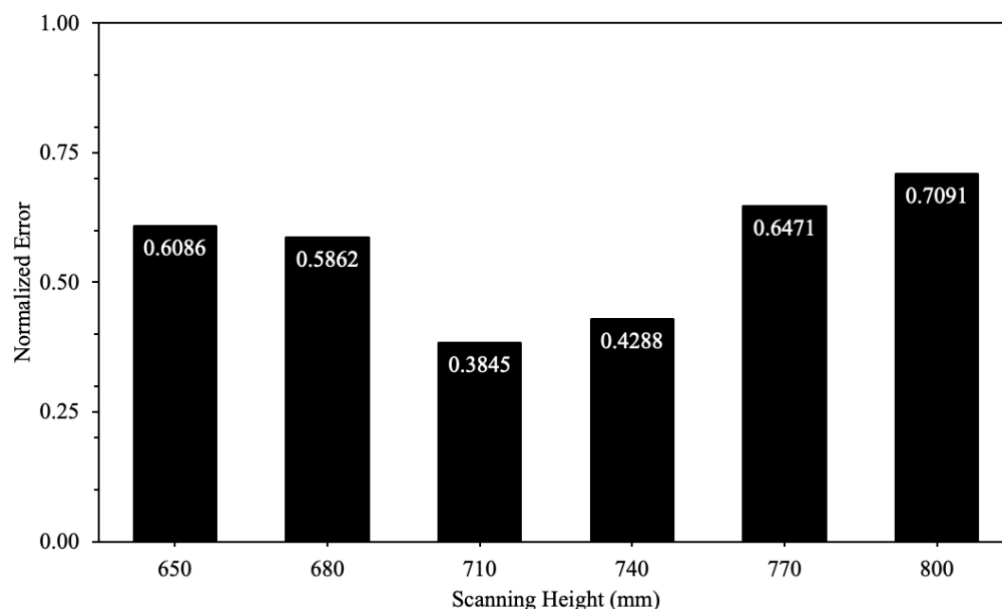


Figure 9: Normalized Error vs. Scanning Height

4.3.2 Conveyor Belt Speed

The effect of part velocity on system performance was assessed by testing three conveyor belt speeds using the system's dial control settings: 0.75, 1.0 (baseline), and 1.5. These settings were calibrated to represent linear part speeds of approximately 32 mm/s, 50 mm/s, and 95 mm/s, respectively.

At the lowest speed (0.75 setting, ~32 mm/s), the average error in x was 37.75 mm, with a minimum of 7.14 mm and a maximum of 96.96 mm. Y error remained relatively low at 2.37 mm, with a range from 0 mm to 10.59 mm, while u error averaged 5.43°, ranging from 3.00° to 7.21°. The combined average positional error in x and y was 20.06 mm. Despite the slow movement allowing more time for prediction, positional estimation became somewhat inconsistent, particularly in the x direction.

At the 1.0 speed setting (~50 mm/s)—used as the baseline for most experiments—the average x error dropped to 14.57 mm, while y error increased to 31.46 mm. The rotational error averaged 5.91°, with bounds between 2.90° and 8.10°. The combined x and y error was 23.01 mm. This configuration provided a relatively strong balance between detection and robot reaction time and was the most stable across repeated trials, with all 10 being successful.

At the highest tested speed of 1.5 (~95 mm/s), the success rate dropped to 4 out of 10 trials. In those four successful cases, the average x error was 25.49 mm (range: 1.48 mm to 59.45 mm), while y error averaged 14.60 mm and u error was 7.82°, with angular errors ranging from 7.33° to 8.55°. The average positional error was 20.04 mm. The failure in six of ten trials was due to the part exiting the camera frame too quickly, preventing full

tracking and velocity estimation. Even in the successful trials, reduced data collection time introduced moderate variability in all axes.

Standard deviation analysis reflects these patterns in consistency across different speeds. At the lowest speed of 32 mm/s, the standard deviation for x error was 12.96 mm, for y error was 1.91 mm, and for u error was 0.67°. The baseline speed of 50 mm/s had the lowest variability in x (4.35 mm) and u (0.84°), but the y error variability was slightly higher at 1.03 mm. At the highest speed of 95 mm/s, standard deviation increased in every category: 12.34 mm in x, 3.23 mm in y, and 3.23° in u. This reflects a marked increase in trial-to-trial error dispersion at higher conveyor velocities. Figure 10 shows the values for conveyor speed against displacement inaccuracies.

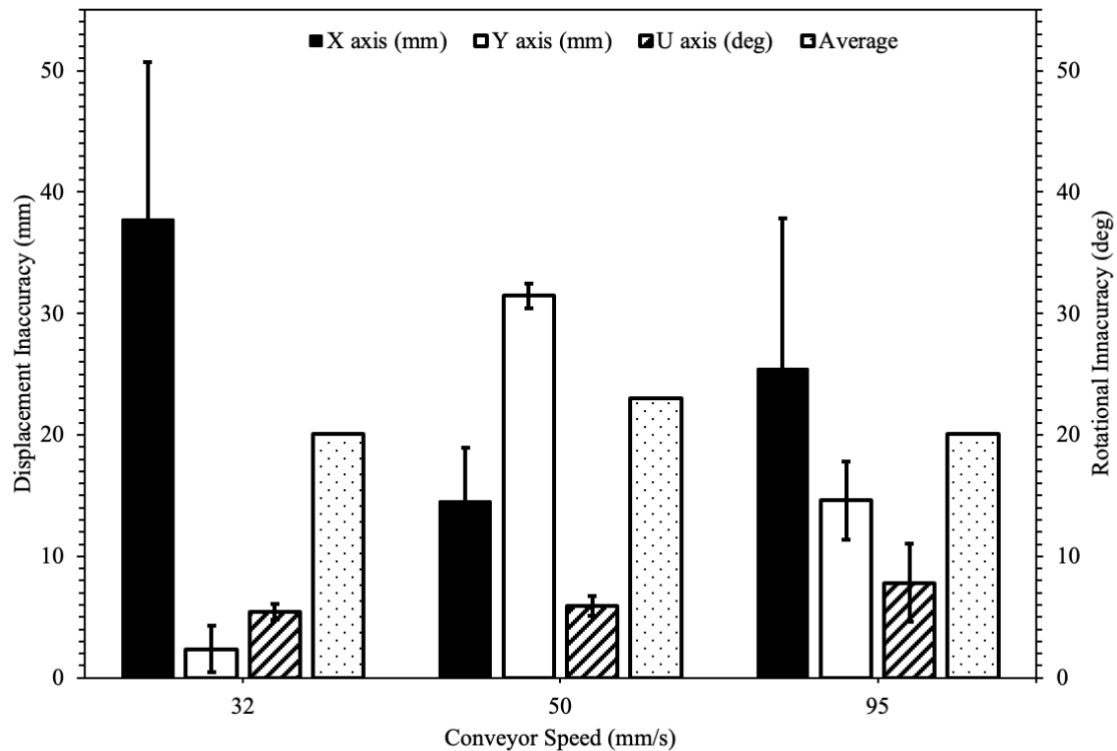


Figure 10: Displacement Inaccuracy vs. Conveyor Speed

A normalized error analysis was performed for conveyor belt speed to account for cross-dimensional variability and to better visualize combined system performance. This method normalizes each axis by its maximum observed error and then averages the values, yielding a unitless score between 0 and 1.

As seen in Fig. 11., normalized error increased as conveyor speed increased, indicating a decline in overall system coordination and reliability. At 32 mm/s, the normalized error was 0.5899, the lowest among all speeds. The baseline speed of 50 mm/s produced the highest normalized error at 0.7142, driven largely by the high y-direction inaccuracy. Interestingly, while the average positional error at 95 mm/s was not drastically higher than the others, its normalized error of 0.7131 reveals a performance closer to the baseline configuration than initially assumed. This discrepancy underscores how normalized metrics can uncover subtleties in multi-dimensional behavior not immediately apparent in unnormalized averages.

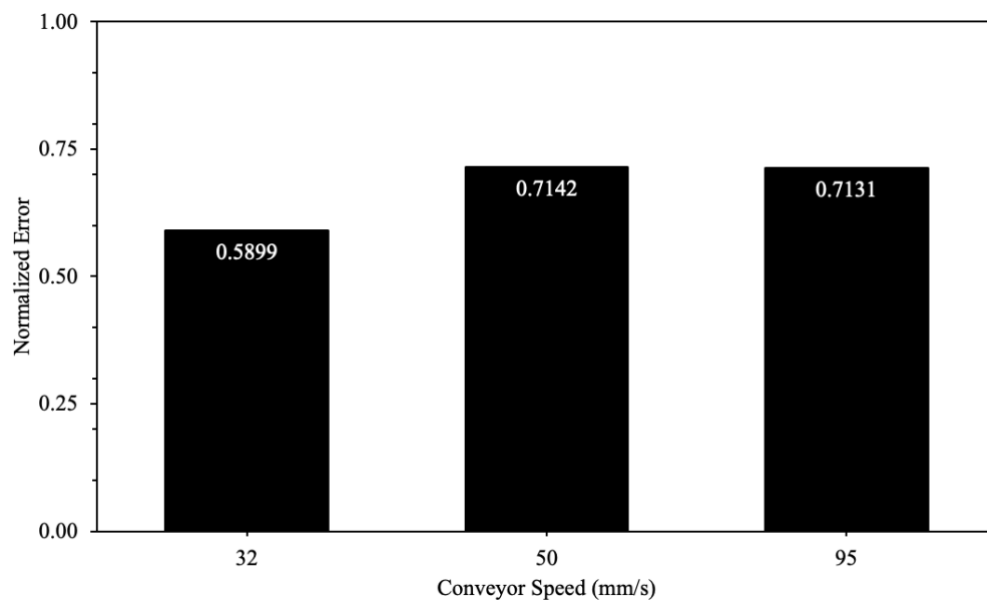


Figure 11: Normalized Error vs. Conveyor Speed

4.3.3 Velocity Estimation Iterations

To assess the influence of data sampling frequency on predictive accuracy, the number of iterations used to estimate part velocity was varied from 8 to 14. Each iteration captured a new part position, which was used to compute an updated velocity average.

At 8 iterations, the system produced an average x error of 15.29 mm, with values ranging from 3.85 mm to 34.67 mm. The average y error increased significantly to 33.47 mm (range: 26.84 mm to 43.90 mm), while u error was 7.05°, ranging from 6.65° to 7.71°. These results demonstrated improved x performance but decreased y stability. At 9 iterations, the system achieved an x error of 29.71 mm, y error of 30.90 mm, and u error of 6.62°. The respective error ranges were 0 mm to 54.32 mm in x, 21.60 mm to 39.08 mm in y, and 3.87° to 9.02° in u. The 10-iteration setup improved average x error to 14.57 mm, with a range from 2.60 mm to 26.10 mm. Y error increased slightly to 31.46 mm, while u error improved to 5.91°, ranging from 2.90° to 8.10°. This configuration provided one of the most balanced trade-offs across all axes.

At 11 iterations, results remained stable with an average x error of 18.61 mm (range: 0.02 mm to 46.56 mm), y error of 30.20 mm (range: 15.54 mm to 48.92 mm), and u error of 6.64°, ranging from 3.84° to 10.85°. By 12 iterations, average x error decreased slightly to 13.02 mm, y error remained stable at 30.88 mm, and u error was 6.45°, with performance beginning to plateau. At the maximum tested count of 14 iterations, the system produced an x error of 15.52 mm (range: 0 mm to 42.46 mm), y error of 6.22 mm (range: 0 mm to 42.46 mm), and u error of 6.56°, reflecting a drop in accuracy.

The standard deviation values offer insight into the consistency of each iteration setting. At 8 iterations, the x error deviation dropped to 2.58 mm, but y error deviation rose

to 3.12 mm, indicating greater inconsistency. At 9 iterations, standard deviation rose to 3.90 mm in x and 3.20 mm in y, marking a return to broader trial-to-trial variability.

At 10 iterations, x error standard deviation was 2.07 mm, indicating the most consistent x predictions across trials. Deviations in y and u were also minimized at 1.03 mm and 0.84°, respectively, suggesting this was the most stable configuration overall. As iterations increased beyond 10, standard deviations began to climb again. At 11 iterations, the x error deviation was 4.62 mm, y error deviation was 4.68 mm, and u error deviation peaked at 1.05°. Similar trends held at 12 and 14 iterations. These results reinforce the notion that while higher iteration counts can smooth velocity estimates, they also introduce processing delays and may reduce responsiveness to rapid motion changes.

A trendline analysis was applied to the average positional error across iterations to better visualize the relationship between sampling frequency and accuracy. The resulting exponential decay curve, shown in Fig. 12., followed the model $y = 33.58e^{-0.11x}$, with a coefficient of determination $R^2 = 0.5152$. While this R^2 value indicates only moderate correlation, it does support the observed trend that increasing the number of velocity iterations generally improves positional accuracy, albeit with diminishing returns after a certain point.

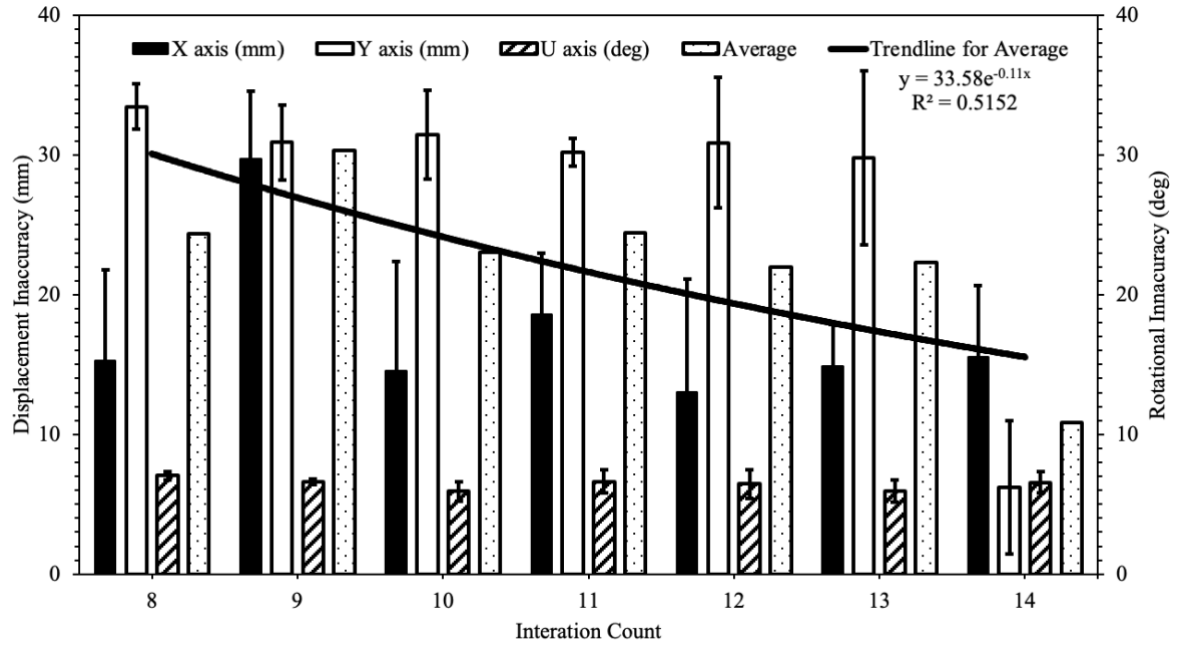


Figure 12: Displacement Inaccuracy vs. Iteration Count. Trendline for the Average series is depicted as well, with the equation and R^2 value included.

To further evaluate the impact of iteration count on overall accuracy, normalized error values were calculated using axis-wise max normalization. This method accounts for differences in unit scale across x, y, and u dimensions, enabling a more unified metric for comparison. The results show a distinct pattern: the normalized error peaked at iteration count 9 with a value of 1.0000, indicating the poorest overall performance in terms of combined displacement and rotational inaccuracy. Conversely, the lowest normalized error occurred at 14 iterations, with a value of 0.5821, highlighting a significant gain in total system accuracy. Notably, although the raw average error values did not always follow a strict monotonic trend, the normalized error reveals that iteration counts above 10 generally yielded better overall accuracy across all axes. These results emphasize the importance of considering normalized error alongside raw values to uncover underlying performance

improvements that may otherwise be obscured by unit disparities or axis-specific variability. Figure 13 illustrates this relationship, plotting normalized error against iteration count.

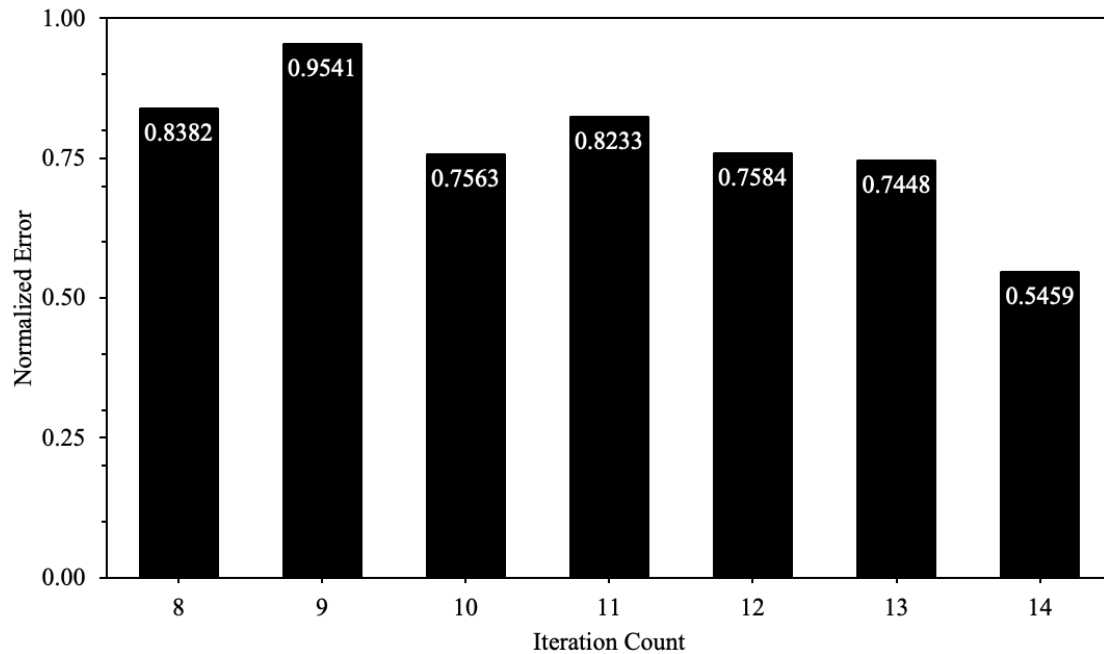


Figure 13: Normalized Error vs. Iteration Count

4.3.4 Robot Motion Constraints

To assess the effect of physical robot responsiveness, tests were conducted by adjusting both maximum speed and acceleration simultaneously. Three configurations were tested: 500 mm/s and 500 mm/s², 1000 mm/s and 1000 mm/s² (baseline), and 1500 mm/s and 1500 mm/s².

At the lowest setting (500 mm/s, 500 mm/s²), 9 out of 10 trials succeeded. The average x error was 17.55 mm, with values ranging from 0 mm to 30.74 mm. The y error was low at 0.13 mm, ranging between 0 mm and 0.64 mm. Rotational error averaged 6.02°,

with a range of 5.72° to 6.44° . The total average error for this configuration was 8.84 mm. The single failure trial was due to the robot's inability to reach the pickup point in time, as the reduced speed and acceleration limited its range and responsiveness.

At the baseline setting (1000 mm/s, 1000 mm/s²), the robot successfully completed all ten trials. The average x error was 14.57 mm, ranging from 2.60 mm to 26.10 mm. Y error increased to 31.46 mm, with values between 27 mm and 33.4 mm. U error averaged 5.91° , ranging from 2.90° to 8.10° . The total average error for this setting rose to 23.01 mm. This configuration represented a strong compromise between mechanical speed and positional stability, maintaining high success and acceptable accuracy across all axes.

At the highest setting (1500 mm/s, 1500 mm/s²), the average x error rose to 44.61 mm (range: 29.87 mm to 52.6 mm). However, y error improved dramatically to 6.15 mm, with a minimum of 0 mm and a maximum of 10.23 mm. Rotational accuracy also improved slightly, with an average u error of 5.59° and a range between 3.43° and 8.85° . Despite improved y and u performance, the x error increase resulted in a total average error of 25.38 mm. This configuration demonstrated that while faster motion limits reduced y and u error, they introduced increased error in the x direction.

Standard deviation results further support the observed trends in variability across the three motion settings. At the lowest speed and acceleration (500 mm/s²), x error standard deviation was 5.92 mm—the highest among the three—indicating unstable horizontal positioning across trials. However, this setting maintained the lowest variability in both y and u axes, with standard deviations of 0.143 mm and 0.137° , respectively. The baseline configuration (1000 mm/s²) demonstrated moderate consistency in the x error, with a standard deviation of 4.7 mm, lower than the slowest setting, though not as

consistent as the highest setting. The y error variability increased significantly to 1.11 mm, while u error standard deviation was 0.90°, reflecting broader trial-to-trial shifts, particularly in y. At the highest motion setting (1500 mm/s²), the x error standard deviation decreased to 3.76 mm, while the y error standard deviation dropped sharply to 1.93 mm. The u error had moderate rotational consistency with a low standard deviation of 1.04°, indicating improved consistency in rotational alignment despite faster robot movements. These values are plotted in Fig. 14.

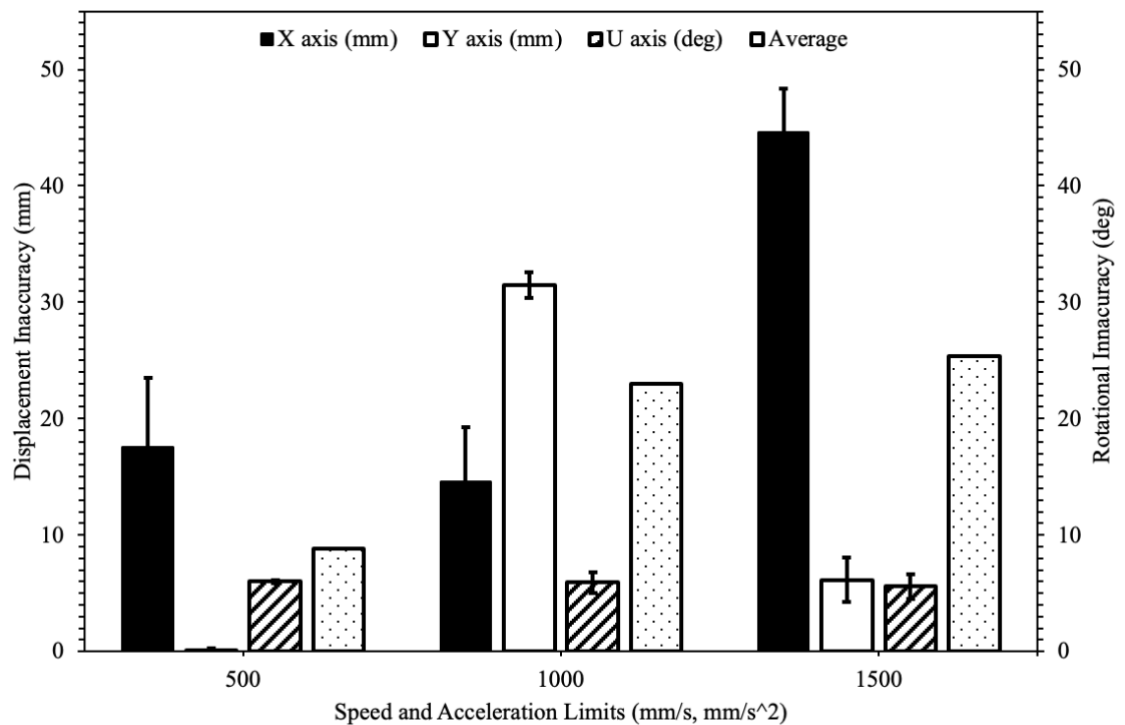


Figure 14: Displacement Inaccuracy vs. Robot Motion Limits

In addition to raw error values, normalized error analysis provides further insight into overall configuration performance, graphed in Fig. 15. By applying axis-wise max normalization, the composite error in x, y, and u was scaled to allow direct comparison

across configurations. The 500 mm/s setting achieved the lowest normalized error at 0.4658, suggesting strong combined accuracy despite minor increases in trial-to-trial x variation. The baseline configuration of 1000 mm/s exhibited the highest normalized error at 0.7698, largely driven by the significant increase in y error. Interestingly, the 1500 mm/s setting resulted in a slightly improved normalized error of 0.7080 compared to the baseline, driven by much lower y error despite the sharp rise in x.

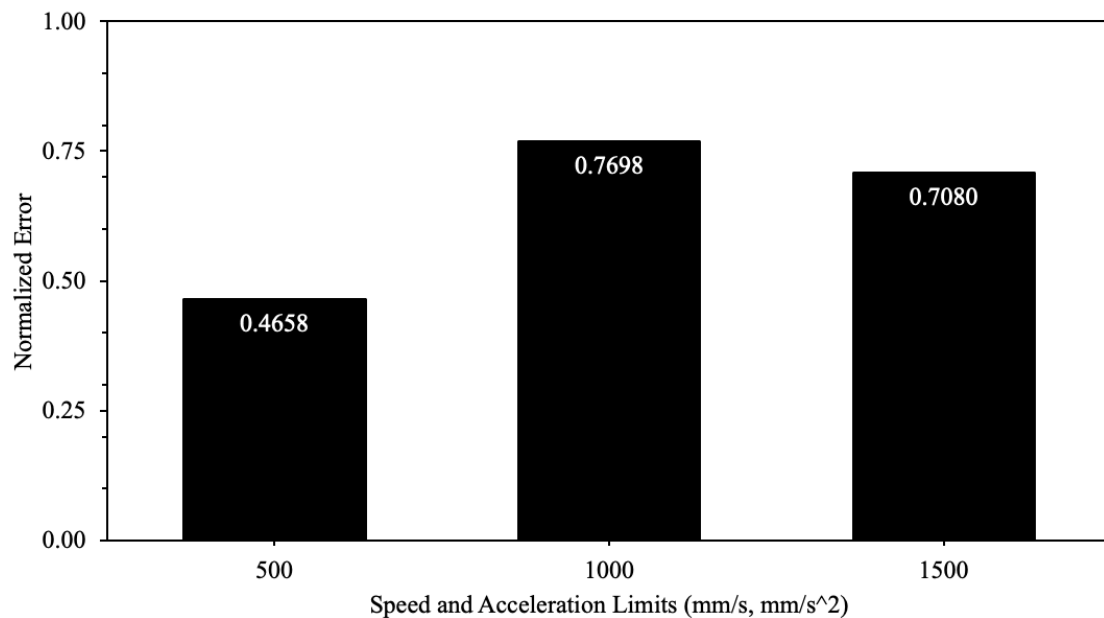


Figure 15: Normalized Error vs. Robot Motion Limits

4.3.5 Summary of Failed Attempts

While the majority of trials across all test scenarios were successful, several configurations experienced partial or total failure in achieving accurate pickups. These failures occurred in cases where either the robot was unable to reach the predicted position in time, or the part exited the field of view before velocity estimation and interception could

be completed. This section summarizes the test conditions under which failures occurred, including the frequency and potential mechanical limitations observed.

The most severe failures occurred at the maximum scanning height of 800 mm, where 9 out of 10 trials failed. At this height, the camera's reduced resolution and diminished ability to consistently identify the part profile led to frequent tracking errors. The single successful trial yielded a relatively high x error of 62.13 mm. Detection failure at this height was likely due to the part's small visual footprint in the image frame, which increased the chance of mismatch with the stored template. At the 650 mm scanning height, although 8 out of 10 trials were successful, the two failures stemmed from abrupt part reorientation at close range, which exceeded the field of view before the camera could compensate. These failures suggest a visibility issue, where the limited viewing angle restricted the system's ability to gather sufficient data for velocity estimation, resulting in either inaccurate predictions or complete tracking loss.

Failures also occurred while testing the effects of conveyor belt speed. At the 1.5 dial setting (95 mm/s), the system completed only 4 out of 10 trials successfully. The primary reason for failure was the rapid movement of the part, which often caused it to exit the frame before sufficient data was collected. In many of these cases, the camera failed to capture enough high-quality images before the part left the observable area, preventing a complete velocity profile from being formed. This speed exceeded the practical capture rate of the system's camera, introducing both motion blur and data gaps in the tracking phase.

Additionally, at the 500 mm/s and 500 mm/s² motion limit setting, 1 out of 10 trials failed due to the robot's inability to reach the predicted interception point in time. The

reduced acceleration caused a significant delay in movement initiation, which—although not affecting y or u accuracy—prevented the gripper from arriving at the correct x position fast enough. In this trial, the part moved out of reach by the time the robot initiated the descent phase.

In total, failures were most closely associated with configurations that introduced excessive uncertainty into either the part's estimated trajectory or the robot's ability to respond. Specifically, high conveyor speeds and high scanning heights posed the greatest challenges to the system's vision tracking reliability, while low motion constraints primarily limited the robot's mechanical responsiveness. Across all failure cases, the common thread was insufficient coordination between visual detection and mechanical actuation, preventing timely and accurate interception.

4.4 Summary of Key Findings

This section consolidates the major outcomes presented throughout Chapter IV, highlighting recurring patterns, relative performance across configurations, and statistically notable accuracy trends. The purpose is to provide a clear overview of the experimental results obtained without analysis or interpretation.

Among the three predictive strategies evaluated, the Hypotenuse Method consistently yielded the highest accuracy across most measured metrics. It achieved the lowest average x error of 14.57 mm, a moderate y error of 31.46 mm, and a rotational error of 5.91°, all while maintaining a 100% trial success rate. The Direct Kinematic Method averaged 40.11 mm in the x error, with stronger performance in y and rotational alignment. The Ambush Method produced intermediate x error levels of 43.28 mm, the best y error at

21.12 mm, and the second highest u error at 6.56° , indicating mixed but successful results across axes.

Optimal scanning height was found between 710 mm and 770 mm, with these configurations delivering the best balance of positional and rotational accuracy. At 770 mm, the system achieved one of the lowest x errors (14.57 mm) and strong rotational performance (5.91°), though it also recorded the highest average y error. Scanning heights of 650 mm and 800 mm demonstrated the most instability. At 650 mm, x error averaged 43.15 mm, and the u error was 8.37° , while at 800 mm, only 1 of 10 trials succeeded. These results reflect a strong dependency on maintaining a camera height within a functional detection window.

The baseline conveyor speed of 50 mm/s (setting 1.0) produced the most stable results, with an average x error of 14.57 mm, y error of 31.46 mm, and u error of 5.91° . At the slower speed of 32 mm/s (setting 0.75), x error increased significantly to 37.75 mm, with a similar trend in rotational error (5.43°). At the highest tested speed of 95 mm/s (setting 1.5), the success rate dropped to 4 out of 10 trials, and x error averaged 25.49 mm, y error 14.60 mm, and u error 7.82° , reflecting a degradation in accuracy with speed increases.

Velocity estimation iteration counts between 9 and 12 yielded the most consistent accuracy. At 10 iterations, the system produced an x error of 14.57 mm, y error of 31.46 mm, and rotational alignment of 5.91° . At the maximum tested iteration count of 14, the system achieved an x error of 15.52 mm, y error of 6.22 mm, and u error of 6.56° , reflecting improved y performance at the cost of slight increases elsewhere. Additionally, a trendline analysis of iteration count versus average error revealed an exponential decay trend $y =$

$33.58e^{-0.11x}$ with an R^2 value of 0.5152, indicating moderate correlation between increased sampling frequency and improved accuracy.

Motion constraints significantly influenced performance. The baseline configuration (1000 mm/s velocity and 1000 mm/s² acceleration) provided balanced results, with an x error of 14.57 mm, y error of 31.46 mm, and u error of 5.91°. Increasing speed and acceleration to 1500 mm/s and mm/s² reduced y error to 6.15 mm and u error to 5.59°, but increased x error to 44.61 mm. Conversely, reducing motion parameters to 500 mm/s and 500 mm/s² improved y error to 0.13 mm, while increasing x error to 17.55 mm and u error to 6.02°.

Failures were most frequently observed at high scanning heights and fast conveyor speeds, where vision detection and part tracking became unreliable. The system failed in 9 of 10 trials at 800 mm scanning height and 6 of 10 trials at 95 mm/s conveyor speed. A single failure occurred during testing with low motion constraints (500 mm/s and mm/s²), where the robot was physically unable to reach the part before it moved out of reach. No complete failures occurred during iteration count tests, although lower iteration counts exhibited more extreme variability in x error, including deviations exceeding 149 mm.

In addition to average errors, standard deviation values offer insight into the system's precision and consistency across repeated trials. The Direct Kinematic Method produced the second lowest standard deviation in x error (3.06 mm), while the Hypotenuse Method showed greater consistency in u error (0.84°) and moderate variation in x and y. The Ambush Method had the highest standard deviation in y error (4.90 mm), as well as the highest standard deviation in rotational alignment (1.41°). For other variables, the lowest standard deviation in u error occurred at 8 iterations (0.18°), and the highest x error

variability occurred at 650 mm scanning height (81.67 mm). These trends reveal where the system achieved not just accuracy, but repeatability.

In addition to traditional raw error analysis, normalized error values were calculated using axis-wise maximum normalization to unify positional and rotational inaccuracies into a common performance metric. These normalized results revealed deeper performance distinctions, highlighting that the 14-iteration configuration produced the lowest overall normalized error (0.5821), while faster conveyor speeds and extreme scanning heights generally led to worse normalized performance. Normalized error analysis proved particularly valuable in cases where raw x, y, and u values shifted independently, providing a single metric that better captures true overall system performance across different conditions.

Chapter V: Conclusion

5.0 Introduction

This chapter consolidates the primary outcomes of the study, reflecting on experimental observations, performance trends, and key takeaways from testing the vision-guided pick-and-place robotic system. It is organized to provide both a concise summary of results and a broader interpretation of what those results suggest in practical and technical contexts. Section 5.1 offers a recap of major findings across all variables, including pickup method, scanning height, conveyor speed, iteration count, and robot motion constraints. It summarizes which configurations performed best and identifies patterns in positional and rotational accuracy, as well as trial success rates. Section 5.2 presents a deeper discussion and interpretation of the results. Each subsection focuses on a specific variable and explains why the observed trends may have occurred. These interpretations incorporate insights related to vision system limitations, robot dynamics, and environmental factors, connecting the data to broader engineering principles and practical implications. Section 5.3 considers how these results relate to system reliability and real-world deployment. It discusses system-level behavior beyond raw accuracy—such as part detection consistency, success rates, and gripper performance—and reflects on what makes this type of vision-guided system viable or limited in different application scenarios. Section 5.4 addresses experimental limitations and outlines potential areas for future work. This includes improvements to hardware, sensor resolution, image processing algorithms, and control logic. It also reflects on the sources of error in the experimental setup and proposes adjustments that could enhance system precision, repeatability, or speed in future iterations of this work.

Finally, Section 5.5 offers concluding remarks on the broader significance and feasibility of this research. It reaffirms the core objectives of the project and reflects on its potential role in expanding access to industrial automation for smaller-scale manufacturers.

5.1 Summary of Key Results

The experimental findings revealed clear trends in system performance as each major parameter was varied. This section summarizes the most important outcomes across pickup strategy, scanning height, conveyor speed, iteration count, and robot motion constraints, with emphasis on comparative accuracy and success rate performance. While detailed values were provided in Chapter IV, the following overview highlights relative strengths and weaknesses of different configurations based on average x and y positional error, rotational alignment, and normalized accuracy.

The Hypotenuse Method yielded the best combined positional accuracy across all three tested methods. It achieved an average x error of 14.57 mm and a y error of 31.46 mm, resulting in a combined average positional error of 23.02 mm. In comparison, the Direct Kinematic Method recorded an x error of 40.11 mm and y error of 30.14 mm, for a combined average of 35.13 mm, while the Ambush Method showed the best y accuracy at 21.12 mm but the highest x error at 43.28 mm, resulting in a combined average of 32.20 mm. These comparisons highlight the Hypotenuse Method's superiority in minimizing error across both axes, achieving a 34% lower average positional error than the next best method.

Optimal performance was achieved within the 710 mm to 770 mm range, with the 770 mm height (baseline) offering the lowest x error (14.57 mm) and strong rotational

performance (5.91°), albeit at the cost of the highest y error (31.46 mm). The 710 mm configuration, on the other hand, produced nearly perfect y accuracy (0.08 mm) and a moderately higher x error (16.45 mm), yielding a combined positional error of 8.27 mm—the best overall. Performance deteriorated outside this range, with the 650 mm height yielding an average x error of 43.15 mm and 800 mm resulting in only 1 out of 10 successful trials. This confirms that scanning height must be tuned within a specific detection window to balance field of view, resolution, and predictive consistency.

The baseline speed of 50 mm/s combined a full success rate with solid accuracy, producing an x error of 14.57 mm and a y error of 31.46 mm. Slower motion at 32 mm/s worsened performance, increasing x error to 37.75 mm and reducing precision despite the longer prediction window. At the fastest setting (95 mm/s), the success rate dropped to 4 out of 10, with an average x error of 25.49 mm and y error of 14.60 mm. Though the positional error seemed lower at high speed, the inconsistency and trial failures suggest diminished reliability. The baseline speed thus remains the most stable configuration for continuous performance.

Accuracy improved steadily with increasing iteration count, confirming that larger sample sizes allowed better velocity estimation. The best performance occurred at 14 iterations, with the lowest x error (15.52 mm) and a substantially reduced y error (6.22 mm), resulting in a combined average error of 10.87 mm—the best across all configurations. At 10 iterations, x error matched that of 14 iterations (14.57 mm) but y error was significantly worse (31.46 mm), producing a combined error of 23.02 mm. This validates that although 10 iterations were stable, 14 iterations cut total positional error by

over 50%. A fitted exponential decay trendline, $y = 33.58e^{-0.11x}$ with $R^2 = 0.5152$, further illustrated diminishing gains beyond 12 iterations.

Changes in speed and acceleration directly impacted positional error. The baseline configuration (1000 mm/s, 1000 mm/s²) had a balanced x error of 14.57 mm and y error of 31.46 mm, for a combined error of 23.02 mm. Increasing the limits to 1500 mm/s and mm/s² improved y accuracy dramatically (6.15 mm) but more than tripled x error to 44.61 mm, yielding the worst combined average of 25.38 mm. Reducing limits to 500 mm/s and mm/s² led to the lowest y error (0.128 mm) but an x error of 17.55 mm, producing a positional average of 8.84 mm—second only to the best scanning height result. Normalized error analysis supported the finding that faster robot movements reduced control in the x-direction while improving other dimensions, highlighting the need for application-specific trade-offs.

5.2 Interpretation and Analysis of Results

While Section 5.1 provided a condensed summary of key outcomes, the following subsections aim to explain why the observed trends occurred by examining the physical, mechanical, and logical behaviors underpinning each variable. Each subsection focuses on one of the core tested parameters; pickup method, scanning height, conveyor belt speed, iteration count, and robot motion limits, and offers reasoning for the system's performance. In some cases, the trends are intuitive and easily explainable, while in others, the reasoning is more speculative or remains undetermined. Where appropriate, this section introduces original hypotheses based on practical system behavior and engineering reasoning. Key

performance metrics referenced here have already been summarized in Section 5.1, presented in full in Chapter IV, and detailed comprehensively in the appendix tables.

5.2.1 Pickup Method

Among the three predictive strategies evaluated—Direct Kinematic, Hypotenuse, and Ambush—the Hypotenuse Method consistently outperformed the others in terms of overall accuracy. Its effectiveness is largely attributable to how it models the physical behavior of the robotic system. Unlike the Direct Kinematic Method, which only considers the vertical (z-axis) travel distance between the robot’s starting position and the part’s expected location, the Hypotenuse Method accounts for the true diagonal path the robot’s end effector must follow to reach the pickup point. By factoring in both vertical and horizontal displacement, it provides a more realistic estimate of the required motion and time needed for interception, aligning more closely with the robot’s actual trajectory and dynamic constraints.

The Direct Kinematic Method, by contrast, assumes the robot moves straight down vertically toward the part’s pickup height. This oversimplification fails to account for the significant horizontal motion the robot must perform, especially when the part is located farther from the centerline of the robot’s home position. As a result, it often underestimates the time required for pickup and causes the robot to arrive late or misaligned. While this method is computationally simple and can produce consistent rotational alignment due to its static orientation assumptions, its disregard for horizontal positioning leads to increased error in the x-direction. This was reflected in the average x error, which was more than double that of the Hypotenuse Method.

The Ambush Method takes a fundamentally different approach. Rather than aiming to intercept the part at its predicted future position, it attempts to “ambush” the part by traveling to a fixed position ahead of its current trajectory and waiting there to intercept it. Although this eliminates the need for continuous time or distance estimation during motion, it introduces a critical vulnerability: error propagation. Since the robot moves significantly farther from the part’s initial location than the other methods, even small inaccuracies in initial velocity or position estimation grow larger as the robot waits for the part to arrive. The longer the gap between prediction and pickup, the more these discrepancies accumulate—especially in the presence of noisy image processing data or slight miscalculations in the part’s speed. This likely explains why the Ambush Method exhibited the highest variability in performance across trials, particularly in rotational alignment and y-direction placement.

Moreover, while both the Direct Kinematic and Hypotenuse Methods are interception strategies, the Ambush Method relies on anticipation. For a given part speed, the Ambush pickup point is typically located farther from the part’s current position than the interception methods. This increased distance allows more time for error to accumulate between the predicted and actual part position. Additionally, the Ambush Method requires precise timing for the gripper’s actuation. Since the robot is already stationed at the pickup point, the exact moment when the gripper closes becomes critical. Even minor discrepancies in timing—caused by sensor lag, velocity estimation error, or mechanical response delays—can result in a failed pickup. As such, the Ambush Method not only amplifies spatial error over time but also becomes increasingly sensitive to temporal

inaccuracies, making the overall pickup success rate more dependent on part size, gripper geometry, and system response precision.

Normalized error metrics reinforce these trends. While the Hypotenuse Method not only yielded the lowest raw x error and strong rotational consistency, it also had the lowest normalized combined error, confirming that it offered the most well-rounded and scalable performance across all dimensions. These results suggest that in vision-guided robotic applications where both timing precision and trajectory accuracy are critical, strategies that model real-world motion paths, such as the Hypotenuse Method, are significantly more reliable than those based on simplified or anticipatory heuristics.

5.2.2 Scanning Height

Testing revealed a clear performance window between 710 mm and 770 mm for the camera's scanning height, with both configurations producing strong results in positional and rotational accuracy. Within this range, the system demonstrated consistently low x-direction error, solid part recognition success rates, and stable motion prediction—indicating that this zone represents an effective balance between field of view and image resolution. The best overall results occurred at 770 mm, while 710 mm showed exceptional stability in y-direction accuracy. Together, these heights define a practical sweet spot for system deployment. Understanding why this window exists requires examining the system's limitations at the extremes. At the upper bound of 800 mm, the camera offered a large field of view, but resolution suffered significantly. The resulting image quality was often insufficient for the vision system to reliably identify the part's profile, particularly in cases where contrast or lighting varied. Only one of ten trials was successful at this height. In the majority of failed cases, the part either went entirely undetected or the system lost

track of it mid-sequence—sometimes detecting the first position but failing to maintain consistent detection over subsequent iterations. This breakdown in continuity meant that the robot could not compute a valid pickup location. At the opposite end, the lowest tested height of 650 mm introduced a different set of challenges. While the resolution was highest at this distance, the field of view became extremely narrow, leaving little spatial buffer for the part to be tracked over time. Parts traversed the camera’s view rapidly, often appearing in only a few frames before exiting the field entirely. In addition, small tracking errors were amplified by the reduced viewing area. Blurring caused by part motion and the robot’s own adjustments further degraded the image quality, making it difficult to capture and process clean part profiles. Although the success rate at this height was better than at 800 mm, performance was still inconsistent and prone to failure in tightly timed sequences.

The 710 mm to 770 mm range represents a balance point between these two extremes—offering a field of view wide enough to capture the part’s full trajectory without significantly sacrificing resolution. Within this optimal window, the average x error was 18.16 mm and the average y error was 11.13 mm, with all trials at both heights resulting in successful pickups. In contrast, outside this window, the system saw a sharp drop in accuracy and reliability. At 650 mm and 800 mm, the average x error rose to 43.15 mm and 62.13 mm respectively, while y errors increased to 4.13 mm and 8.74 mm. Additionally, the 800 mm configuration had only one successful trial out of ten. This comparison underscores the system’s sensitivity to camera positioning: even small shifts outside the ideal range led to performance declines of over 100% in x-direction error alone. These results illustrate that neither resolution nor field of view can be optimized in isolation. High resolution with a restricted view leads to truncated tracking, while a wide

field with low resolution compromises image clarity. The 710–770 mm range emerged as the only region where both variables were simultaneously acceptable—enabling reliable tracking and accurate velocity estimation.

It should be noted that the specific results achieved here are partially influenced by the camera’s fixed internal settings and physical mounting position. Future work will explore how adjustable camera parameters—such as focus depth, resolution modes, and exposure timing—may shift or even broaden the optimal scanning height range. These findings reinforce the importance of camera selection and setup as key factors in the overall accuracy and reliability of vision-guided robotic systems.

5.2.3 Conveyor Belt Speed

The effect of conveyor belt speed on system accuracy followed a mildly increasing trend in both average and normalized error as part velocity rose, though the overall differences across the tested speeds were small. At the lowest tested speed of 32 mm/s, the system achieved the lowest average error (20.06 mm) and the lowest normalized error (0.5899). At the baseline speed of 50 mm/s, the average error slightly increased to 23.01 mm, and the normalized error rose to 0.7142. At the fastest tested speed of 95 mm/s, the average error decreased slightly to 20.04 mm, though the normalized error remained high at 0.7131. This highlights a key distinction between the two metrics: while raw average error did not vary dramatically across speeds, normalized error shows a more consistent upward trend, suggesting that the system’s proportional error increases slightly as part speed increases, even if absolute error remains relatively constant.

This trend makes intuitive sense. A faster-moving part becomes more difficult to intercept accurately due to reduced reaction time and increased distance between detection and pickup. As observed in the ambush method, any deviation in velocity estimation or detection timing becomes amplified the farther the part travels before pickup. In the context of conveyor belt speed, a faster part inherently requires the robot to intercept further downstream, magnifying the effects of any small errors in velocity prediction or gripper trajectory. This explains the rise in normalized error, even when average error appears to plateau.

However, the relative stability of average error values across the tested range underscores a valuable characteristic of the system: within its operational detection window, part velocity does not significantly degrade accuracy. The highest and lowest average errors across the three speeds differed by less than 3 mm, and all three configurations resulted in successful trial completion rates of 100% (10/10). This robustness suggests that the system can be deployed across a moderate range of part speeds without necessitating major algorithmic or hardware adjustments.

It's important to note, though, that this stability exists only within the functional speed limits of the system. During exploratory tests at conveyor speeds above 95 mm/s, the system began to fail in recognizing the part consistently due to excessive motion blur and insufficient frame coverage. Even using the baseline scanning height of 770 mm, which provided a balanced field of view and resolution, the part would exit the frame too quickly for reliable multi-frame tracking. This limitation echoes the issues observed at extreme scanning heights and underscores that, above a certain threshold, part velocity does impact

system performance in a more severe and binary fashion—making it impossible to collect reliable pickup data at all.

Conversely, speeds below 32 mm/s were not tested due to hardware constraints of the conveyor belt controller, which could not maintain slower, consistent movement. Future work may involve testing sub-32 mm/s speeds using more precise motor controllers to determine if ultra-slow motion offers marginal accuracy gains or introduces new issues such as part oscillation or image processing stutter. For now, the takeaway is that within the range of 32–95 mm/s, the system demonstrates strong velocity tolerance. Accuracy does decline gradually as speed increases, particularly when measured proportionally, but this effect remains relatively minor until the system’s upper speed limit is surpassed.

5.2.4 Iteration Count

The number of iterations used to estimate part velocity was found to be a critical factor influencing system accuracy. Though earlier results appeared to suggest that 8 iterations yielded relatively low average error, further analysis reveals that this value is likely an outlier. The outlier removal strategy—based on boxplot statistical filtering—may have removed extreme deviations from the 8-iteration data, artificially lowering the average error by excluding multiple poor predictions. This is likely due to the limited sampling available at low iteration counts, where with only a few observed positions, the system might randomly “get lucky” in estimating velocity for some trials. As a result, while 8 iterations appears relatively accurate in raw output, it does not follow the broader trend observed in the remaining data and may be misleading as an indicator of consistent performance.

Beginning at 9 iterations and continuing through 14, a clear and consistent trend emerges: increasing the number of velocity estimation iterations generally improves overall system accuracy [17]. This trend is particularly evident when examining normalized error, which decreases steadily from 0.9541 at 9 iterations to just 0.5459 at 14 iterations. A similar improvement is observable in average error as well, with x and y displacement inaccuracies falling with higher iteration counts. For example, the average error drops from 30.31 mm at 9 iterations to just 10.87 mm at 14 iterations. These results reinforce the intuitive understanding that additional iterations provide more velocity data points, allowing for a more stable and accurate velocity estimate.

This trend is further supported by the exponential regression model applied to the data, which fit an equation of the form:

$$\text{Error} = 33.58e^{-0.11x}$$

with a coefficient of determination $R^2 = 0.5152$, reflecting a reasonably strong fit. Extrapolating from this equation provides insight into expected performance beyond the tested range. At 25 iterations, the model predicts an average error of approximately 2.15 mm. By 35 iterations, this falls to 0.71 mm, and by 45 iterations it approaches 0.24 mm. While it is unlikely that actual system performance would meet these exact predictions, especially given real-world noise and limits in precision, these projections do suggest that continued improvement can be expected, albeit at a diminishing rate. Indeed, diminishing returns are a central consideration when increasing iteration count. Each additional sample contributes progressively less to the improvement of the velocity estimate. Statistically, the marginal effect of a new observation on a running average decreases as the sample size increases. This effect places a practical upper bound on the benefits gained from additional

iterations, even in an ideal system. Moreover, real-world implementation introduces further constraints. As mentioned in Chapter IV, the system ignores the first three observed velocity values in each test to allow the robot to stabilize and align with the part's motion. Thus, a 14-iteration test includes only 11 active data points in practice. Increasing iteration count further would lengthen the system's scanning duration, pushing the predicted pickup point farther down the conveyor. Eventually, this delay could result in the part exiting the robot's reach or even leaving the vision system's frame entirely before pickup occurs. In setups with shorter belts or limited reach, this limitation becomes a critical concern. Therefore, while higher iteration counts improve accuracy, they also introduce risks of failure due to physical constraints.

In summary, increasing iteration count improves prediction accuracy due to more robust velocity estimation, a trend that aligns with both theoretical reasoning and experimental results. However, practical considerations—including processing time, robot reach, and diminishing returns—place an upper bound on the number of iterations that can be used effectively. This trade-off must be carefully considered in real-world applications, especially in time-constrained or spatially limited environments.

5.2.5 Robot Motion Limits

Among all tested variables, robot motion limits presented some of the least intuitive results. One might expect that improving the robot's mechanical responsiveness would enhance pickup accuracy by enabling it to reach the predicted part location more quickly. However, the data showed the opposite trend: as the speed and acceleration settings increased from 500 to 1500 mm/s and mm/s², the system's accuracy declined, especially when measured by normalized error. The normalized error increased from 0.4658 at 500

mm/s to 0.7080 at 1500 mm/s, and average error rose from 8.84 mm to 25.38 mm across the same range.

While these findings are repeatable, their underlying cause remains unclear. One likely explanation is that increasing robot speed and acceleration introduces higher-frequency mechanical oscillations that affect the mounted camera. Because the camera is attached via a relatively lightweight plastic mount, these oscillations may create small but significant vibrations—especially during rapid accelerations or abrupt stops. Since the camera is angled downward toward the part, even slight tilting or vibration can alter the perceived profile of the part or distort its edges during image capture. This can cause subtle shifts in the estimated part centroid or orientation, ultimately leading to inaccurate pickup predictions despite precise robot movement. This idea is supported by exploratory testing performed at the robot's maximum motion limits of 2000 mm/s and mm/s². Although full trials were not completed at that level, informal observations revealed consistent detection issues, particularly when trying to track multiple part positions in sequence. The system could reliably detect the first part position while the robot was stationary, but it often failed to capture subsequent observations after initiating motion. This behavior suggests that mechanical disturbance introduced by high-speed robot operation could be impacting the camera's ability to capture consistent frames.

Interestingly, although the robot itself has high internal accuracy and repeatability, the pickup accuracy did not improve with increased motion speed. This indicates that the source of the error is not in the robot's motion execution but rather in the upstream process—specifically in how part location is perceived and interpreted. Faster robot movement may introduce timing irregularities or misalignments between when the image

is captured and when the data is processed. Alternatively, it may shorten the exposure window or increase motion blur if the robot is still moving or vibrating when a frame is captured. These effects are especially impactful in vision-based systems relying on rapid, accurate sampling. Another potential contributing factor is the synchronization between robot motion commands and image acquisition. At higher speeds, even slight lags in system timing or control latency may have a disproportionate impact on performance, particularly when parts are moving quickly through the field of view. These timing mismatches could explain why the camera successfully detects the part's first position but fails to reliably detect subsequent ones. Although not directly reflected in the final trials, testing at lower-than-500 mm/s speeds was attempted. However, the robot's motion became so slow that the predicted pickup point occasionally fell outside the robot's reachable workspace or off the edge of the conveyor entirely. As a result, 500 mm/s was selected as the lower limit for meaningful testing.

In conclusion, while increasing robot motion limits might appear beneficial on paper, the results show that there is an optimal balance between mechanical responsiveness and system stability. Beyond a certain point, speed-related disturbances appear to outweigh any gains in reactivity. The observed trend is not as easily attributable to a single source as with other variables, but plausible explanations point to vibration, motion blur, timing mismatches, and the physical limitations of the camera mount. Future improvements might involve stiffening the camera fixture, refining synchronization between sensing and motion, or implementing damping techniques to reduce vibration at high speeds.

5.3 Feasibility, System-Specific Considerations, and Research Objective Reflection

A core goal of this research was to evaluate the feasibility of a vision-guided, single-camera robotic pick-and-place system for use in practical automation environments. Specifically, the system was designed with small- to mid-sized manufacturers in mind—those who may seek accessible, adaptable solutions to begin integrating automation into existing processes without the high cost or complexity of industrial-grade systems. The experiments described in Chapter IV and the performance trends discussed in Section V.2 provide a basis to determine whether this concept can be successfully applied in real-world contexts.

To begin, Research Objective 1 sought to determine the maximal achievable accuracy of the system. This was done through testing across a range of variable configurations, yielding best-case results such as an average displacement error of 10.87 mm and normalized error as low as 0.5459. While these values are not sufficient for high-precision manufacturing contexts, the aim of this project was not to meet the standards of advanced multi-sensor platforms, but to assess whether a lower-cost, single-sensor approach could achieve reliability within acceptable functional limits for less demanding tasks. As shown in the industry comparison section of Chapter IV, accuracy expectations differ significantly across industrial applications. In highly regulated or tolerance-critical sectors such as aerospace, semiconductors, and fine electronics assembly, even small deviations can render a system unusable. In contrast, many general-purpose manufacturing or logistics tasks—such as part staging, basic pick-and-place, or rough sorting—can tolerate some margin of error depending on the use case. While this system’s average accuracy of ~94% does not meet universal standards, it does represent a meaningful level

of reliability that could be appropriate for certain operations, particularly if supported by thoughtful integration with part and gripper design. This leads to an important system-specific consideration not emphasized in earlier sections: the relationship between average accuracy and overall pickup success. While accuracy was measured as the distance between the robot's gripper center and the actual center of the part, the true practical impact of that deviation depends on the geometry of the part and the dimensions of the end effector. For example, in this experiment, the gripper width left roughly 8 mm of tolerance on each side of the part. This meant that inaccuracy greater than 8 mm in either direction would likely result in a failed pickup. However, if a manufacturer used a gripper that extended 20 mm beyond the part width on each side, that same average deviation could result in a much higher pickup success rate. This illustrates a key concept: physical configuration (e.g., gripper type, part dimensions) plays a vital role in offsetting system limitations. The robot's errors are always measured in absolute positional terms, but how much those errors matter depends on the margins built into the hardware.

For companies working with larger parts, more forgiving tolerances, or recyclable production streams (where the occasional missed part can be recirculated or removed without halting operations), a 94% pickup rate may be more than sufficient. Conversely, in environments where 100% reliability is mandatory and margin for error is nonexistent, the current version of this system would not be suitable. This speaks directly to Research Objective 2: determining whether the system can offer a viable level of accuracy relative to real-world functional needs. Based on these results, it is clear that while the system may not suit every industry, it does align well with use cases that prioritize cost-effectiveness and adaptability over extreme precision.

Research Objective 3 called for an assessment of practical system limitations. Among the most relevant findings are that system accuracy varies most significantly at the extreme values of the tested variable configurations. For example, overly high conveyor speeds or excessively low scanning heights both resulted in frequent detection failures. Additionally, higher robot speed and acceleration limits degraded accuracy in unexpected ways—possibly due to vibrations or instability in the lightweight camera mount. This trend was not predicted in advance, and while further testing would be needed to isolate the cause, it reinforces the importance of evaluating system behavior across its entire usable range rather than assuming linear relationships.

These findings also suggest that improvements in robustness may not necessarily require additional sensors or expensive upgrades. For example, refining software-based prediction algorithms, optimizing velocity filtering, or improving mechanical stability of the camera mount could enhance performance without changing the system’s fundamental design. The ability to make such improvements while retaining the low-cost, single-camera approach strengthens the system’s value as a modular automation concept.

Ultimately, the system’s feasibility should be judged by its reliability in real-world pickup tasks—not just by abstract measures of accuracy. With modest tuning of end effector geometry and part layout, this setup can achieve pickup success rates that approach practical utility in a wide range of environments. Its simplicity, affordability, and ease of deployment make it a strong candidate for manufacturers who are looking to take early steps into automation without requiring the full infrastructure or precision demanded by larger-scale systems.

In conclusion, the system successfully meets the goals of all three research objectives. Objective 1, determining maximal achievable accuracy, was addressed through extensive testing. Objective 2, evaluating alignment with practical industry tolerances, revealed that while the system may not suit high-precision applications, it performs well enough to support general-purpose manufacturing workflows when configured correctly. Objective 3, understanding system limitations, helped define the operational window where the system is most effective and highlighted factors such as part speed, scanning height, and iteration count that influence reliability. Taken together, these results affirm the feasibility of this vision-guided robotic system as a viable proof of concept for smaller manufacturers seeking scalable, low-cost entry into industrial automation.

5.4 Limitations and Future Work

While the system demonstrated promising performance and reliability, several limitations inherent to the experimental setup were identified throughout the development and testing process. These limitations represent areas where future work can directly improve the system's accuracy, consistency, and real-world applicability.

One key source of error was the manual alignment method used to evaluate the accuracy of pickup predictions. As outlined in the methodology, the robot's gripper was manually jogged to align its center with the center of the part. Although visual alignment provided a reasonably consistent benchmark, it inevitably introduced small measurement inaccuracies, likely on the order of a few millimeters. While this is minor relative to the 20–30 mm average errors observed across most tests, it could still slightly skew accuracy metrics. Future iterations of this work may incorporate a calibration tool—a 3D-printed

device with matching geometry to the test part that attaches to the end effector and allows more consistent centering. Alternatively, a downward-facing laser pointer, precisely aligned to the gripper's center, could be used to align with a marked center point on the part, reducing user-dependent measurement variability.

Another experimental limitation was the conveyor belt control. While the belt maintained relatively stable speeds once set, the adjustment dial was analog, making precise and repeatable speed selection difficult. This analog nature also prevented speeds below approximately 32 mm/s, where torque limitations likely caused inconsistent motion. Future work could integrate a digitally controlled conveyor belt system capable of accurate low-speed operation and consistent stepwise control. This would improve consistency across trials and allow a broader range of part velocities to be explored, particularly at the lower end where performance remains understudied.

Lighting conditions during testing also posed challenges. Although efforts were made to reduce variability—such as closing shades and minimizing personnel movement—ambient light from windows and ceiling fixtures introduced uncontrolled variance. Shadows cast by people or changing daylight conditions affected part visibility and image contrast. While the system was designed to be adaptable to uncontrolled lighting environments, controlled lighting during the testing and development phase would have helped isolate performance-limiting variables and refine the vision system's robustness.

Camera limitations also played a role. The Epson camera used in this experiment was a 5-megapixel monochromatic sensor, which, while sufficient for basic shape recognition, introduced constraints in resolution and exposure control. A higher-resolution sensor or one capable of faster shutter speeds could improve image clarity and part profile

definition, especially during faster part movement. Additionally, the focus was adjusted manually based on visual feedback, which introduces another layer of user-dependent variability. A more advanced camera system with digital or assisted focus adjustment—potentially using a calibration target—would ensure optimal image sharpness and consistency. Further, while the Epson RC+ software allows manual adjustment of internal camera settings (e.g., brightness, contrast, sharpness), future research could systematically explore how these parameters affect detection reliability under varying conditions.

Beyond these limitations, there are several promising directions for future development. One area involves refining the velocity estimation algorithm. Currently, the system captures multiple part positions over time and averages the resulting velocities to estimate motion. While effective, the process could be improved with active outlier filtering. For example, the system could ignore velocity values that deviate sharply from the running average—such as sudden spikes caused by blur, lighting changes, or early robot movement—provided they exceed a reasonable threshold. Such a filter could help mitigate inaccuracies introduced by initial motion irregularities, improving stability without increasing computational complexity. Importantly, this kind of filtering must be carefully designed to avoid rejecting legitimate values from rapid motion or sudden changes.

Expanding the tested parameter space is also a key avenue for future work. Several variables—such as conveyor speed, robot speed, and robot acceleration—were tested in only three configurations due to time constraints. Increasing the resolution of these tests by using smaller intervals and more data points would allow a clearer understanding of nonlinear trends and error inflection points. Additionally, decoupling the speed and

acceleration variables and testing them independently would help clarify their distinct effects on system performance.

Iterations could also be pushed beyond 14. As shown in Chapter IV, there is a clear trend of improved accuracy with increased iteration count. Based on the exponential decay trendline derived from the data, error continues to decrease as more iterations are used, though with diminishing returns. For example, extrapolated values suggest that 25 iterations may result in an average error of 2.15 mm, 35 iterations in 0.71 mm, and 45 iterations in just 0.24 mm. These predictions are theoretical and assume diminishing error will continue along the established curve, though practical constraints such as time and range must also be considered. As noted, the system disregards the first three velocity readings to avoid early jerk effects, meaning a 14-iteration configuration effectively uses only 11 active readings. Testing higher iteration counts in future work would clarify how far this trend continues before plateauing.

Additional tests should also examine alternate part delivery mechanisms, such as angled slides or vibratory feeders, to evaluate how different motion profiles impact detection and velocity prediction. Similarly, rotating the conveyor belt's orientation within the robot's field of view—introducing greater x- and y-axis mixing—would offer insight into system robustness across diverse spatial configurations. These tests could help validate the system's flexibility in adapting to complex layouts or unusual part trajectories.

Beyond experimental testing, future work should aim to deploy the system across a range of industrial conditions. This includes testing with parts of different sizes, shapes, materials, and surface finishes, as well as exploring a broader array of end effectors. Such work would help determine whether the ~94% accuracy observed in this study holds

consistently or varies depending on configuration and would allow researchers to quantify the degree to which part and gripper dimensions offset positioning error in practice.

Additionally, incorporating lightweight artificial intelligence (AI) components may help further optimize detection and estimation. While it is important to avoid overcomplicating the system—especially given its goal of remaining accessible and affordable—a simple AI model trained on part movement patterns or camera outputs could improve both velocity estimation and part recognition. This could take the form of neural network-based image analysis or predictive trajectory modeling based on prior data. Importantly, any such addition should preserve the system’s ease of use and low overhead, which are key differentiators from more expensive multi-sensor solutions.

In summary, the experimental limitations observed in this work provide clear paths for immediate system improvement. Many of the errors stem from practical, solvable issues in measurement precision, camera tuning, conveyor control, and lighting consistency. Meanwhile, longer-term future work should focus on increasing testing coverage, refining algorithms, exploring hardware variation, and potentially integrating modest AI assistance. Each of these developments would enhance the system’s accuracy, repeatability, and robustness, helping it better meet the demands of real-world industrial automation while remaining cost-effective and adaptable.

5.5 Concluding Remarks

This work set out with three core research objectives: to determine the maximal achievable accuracy of a single-camera robotic pick-and-place system, to evaluate the key variables that influence this accuracy, and to assess the system’s practical viability in real-

world industrial applications. Through extensive experimental testing and analysis across multiple operational parameters—including pickup method, scanning height, part velocity, iteration count, and robot motion constraints—each objective was successfully addressed. The data collected offers a clear picture of how design decisions influence system performance, revealing key trends and trade-offs across configurations.

While the system does not meet the precision thresholds required by high-end industrial robotics, it consistently demonstrated reliable detection and pickup performance across a range of realistic conditions. These results support the system’s viability as a robust and adaptable solution for simpler automation tasks. Its straightforward design, low hardware complexity, and flexible testbed make it a compelling candidate for research, prototyping, and certain real-world deployment scenarios—particularly where ultra-precise motion is not essential. Importantly, the actual pickup success rate in applied contexts will depend on system-specific factors such as the size of the gripper, the geometry of the part, and the required tolerances of the task. When those elements are thoughtfully configured, this system can maintain a high rate of successful operation. The ability to compensate for accuracy limitations through part and gripper design enables a level of adaptability not often emphasized in more rigid, high-precision systems.

The broader significance of this research lies not just in its proof-of-concept results, but in what it represents for the accessibility of automation technology. Industrial automation, while increasingly vital across sectors, often remains out of reach for small-to-midsize manufacturers due to high capital costs, integration complexity, and the expertise needed to deploy cutting-edge systems. These barriers disproportionately benefit larger corporations with the infrastructure to support such investments. What this system

demonstrates is that a simpler, more modular, and more cost-effective approach can still yield meaningful automation—opening the door to efficiency improvements for companies that have traditionally been excluded from these benefits. Moreover, the work points to a promising path forward. Future enhancements in code sophistication, camera quality, image processing algorithms, and intelligent velocity filtering may allow systems such as this to shrink the performance gap between affordable solutions and top-tier industrial setups. The integration of lightweight AI models to assist in prediction and refinement further strengthens this potential. As these improvements are realized, the trade-off between system cost and operational accuracy may narrow even further, making affordable automation not only viable, but competitive. Beyond the technical contributions, this research underscores the value of designing engineering solutions that prioritize inclusivity, scalability, and practicality. The system’s adaptability to different environments, its modular testing framework, and its focus on quantifiable performance provide a model for how applied robotics can serve real-world needs—even when perfection isn’t the goal. Creating tools that help level the playing field for smaller manufacturers reflects not only a meaningful engineering challenge, but a socially significant one as well. Ultimately, this project represents more than a set of experiments or error statistics, it embodies a practical vision for bridging the gap between what automation can do and who can afford to implement it. In exploring the capabilities of a minimalist, single-camera solution, this research has delivered a compelling proof of concept with genuine promise for future adoption, refinement, and impact. Its success lies not just in the numbers it achieved, but in the possibility it creates for innovation, accessibility, and more equitable participation in the future of automated manufacturing.

REFERENCES

- [1] L. Pan, L. Yang, Y. Yin, and Y. Tan, “An improved APF-RRT* algorithm for robot path planning in complex environments,” **Machines**, vol. 12, no. 1, p. 19, 2023. [Online]. Available: <https://doi.org/10.3390/machines12010019>. [Accessed: Apr. 8, 2025].
- [2] T. Ma et al., “CLSQL: Improved Q-learning algorithm based on continuous local search policy for mobile robot path planning,” **Sensors**, vol. 22, no. 15, p. 5910, 2022.
- [3] X. Zhang et al., “Kinematic modeling and dynamic prediction of moving parts for robotic pick-and-place tasks,” **IEEE Access**, vol. 9, pp. 58347–58355, 2021.
- [4] H. Qiu and Y. Zhu, “3D trajectory prediction of moving targets for robotic arms using hypotenuse-based methods,” **IEEE Trans. Ind. Informat.**, vol. 16, no. 8, pp. 5522–5530, 2020.
- [5] K. Sato, T. Yamada, and H. Nakamoto, “Enhancing visual detection by contrast optimization in monochromatic vision systems,” **IEEE Access**, vol. 9, pp. 150702–150710, 2021.
- [6] P. Wang and R. Liu, “Contrast adjustment impact on computer vision accuracy for robotic applications,” **IEEE Sensors J.**, vol. 21, no. 12, pp. 13554–13562, 2021.
- [7] R. Singh and P. Mistry, “Iterative velocity estimation for vision-guided robotic pick and place,” **IEEE Access**, vol. 8, pp. 106784–106793, 2020.
- [8] S. Qin et al., “Implementation of a real-time object pick-and-place system based on a changing strategy for rapidly-exploring random tree,” **Sensors**, vol. 23, no. 10, p. 4814, 2023. [Online]. Available: <https://doi.org/10.3390/s23104814>. [Accessed: Apr. 8, 2025].
- [9] R. J. Li, B. Li, and Z. Jin, “Application of ROS-based robotic systems in automated manufacturing: A review,” **IEEE Access**, vol. 9, pp. 115220–115232, 2021.
- [10] W. Li, Z. Zhao, and H. Xu, “Vision-based object detection and sorting using industrial robot,” in **Proc. IEEE Int. Conf. Autom. Sci. Eng. (CASE)**, 2021.
- [11] C. C. Chang and H. T. Chen, “Hybrid RRT-APF path planning for mobile robots,” **IEEE Access**, vol. 8, pp. 155520–155532, 2020.
- [12] J. Huang, B. Zhang, and T. He, “Multi-camera and RL-based robotic pick and place in dynamic environments,” **IEEE Access**, vol. 10, pp. 24080–24089, 2022.

- [13] Y. Duan et al., “Monocular vision-based part recognition and pick-up strategy for robotic systems,” **IEEE Trans. Autom. Sci. Eng.**, vol. 19, no. 4, pp. 3492–3504, 2022.
- [14] Z. Wang and P. Chen, “Dynamic pick-and-place using predictive models in 3D environments,” **IEEE Access**, vol. 8, pp. 94563–94572, 2020.
- [15] A. Kim and J. Choi, “Adaptive lead-and-wait strategy for conveyor pick-up by industrial robots,” **IEEE Robot. Autom. Lett.**, vol. 6, no. 2, pp. 1438–1445, 2021.
- [16] J. Chen et al., “Effect of conveyor speed on dynamic object manipulation with vision-based robots,” **IEEE/ASME Trans. Mechatronics**, vol. 27, no. 1, pp. 100–110, 2022.
- [17] L. Gao et al., “Influence of scanning height on 2D vision-based robotic manipulation accuracy,” **IEEE Access**, vol. 10, pp. 45621–45631, 2022.
- [18] C. Wang et al., “Robot speed and acceleration constraints for improved dynamic pick-and-place accuracy,” **IEEE Trans. Ind. Electron.**, vol. 68, no. 6, pp. 5123–5133, 2021.
- [19] Epson Robots, “VT6L 6-axis robot,” Epson America, 2024. [Online]. Available: <https://epson.com/For-Work/Robots/6-Axis/Epson-VT6L-All-in-One-6-Axis-Robot/p/RVT6L-A901SS>. [Accessed: Apr. 8, 2025].
- [20] OnRobot, “2FG7 parallel gripper datasheet,” OnRobot, 2024. [Online]. Available: <https://onrobot.com/us/products/2fg7>. [Accessed: Apr. 8, 2025].
- [21] Epson Industrial Cameras, “Epson CV-X5000 series 5.0MP GigE camera datasheet,” Epson, 2024. [Online]. Available: https://www.epson.eu/en_EU/products/robot-options/robot-peripherals/epson-5m-gige-mono-camera/p/19422. [Accessed: Apr. 8, 2025].
- [22] Epson RC+, “Epson RC+ 7.0 user manual,” Epson, 2023. [Online]. Available: <https://epson.com/robots-industrial-automation-development-software>. [Accessed: Apr. 8, 2025].

Appendix A

Appendix A presents the full data for all predictive pickup tests conducted across each method and parameter variation described in Chapter IV. Each table in this appendix follows an identical structure: ten individual trials are shown along with recorded errors in the X, Y, and U directions, and an “Outlier” column flags values identified as outliers based on interquartile range (IQR) analysis. Below each trial block, statistical summaries are provided, including averages, standard deviations, quartiles (Q1, Q2, Q3), IQR values, and calculated upper/lower bounds for outlier detection. A blank value signifies a failed attempt. Unless otherwise noted, all tables follow this structure. Each table caption identifies the specific test case or parameter value under which the data was collected.

Table A.1 – Full data for the Direct Kinematic Method test case.

Trials	X	Y	U	Outlier
1	42.8	26	6.9	No
2	32.6	20.8	7.4	No
3	34.9	35.6	7	No
4	33.3	35.1	6	No
5	48	26.8	7.3	No
6	31.9	20.8	7.5	No
7	47.2	41	5.6	No
8	47	36.1	5.4	No
9	43.2	33.5	6.5	No
10	40.2	25.7	8.1	No
Averages	40.11	30.14	6.77	
Standard Dev	6.11644505	6.64563014	0.83432608	
Median	41.5	30.15	6.95	
Q1	33.7	25.775	6.125	
Q2	41.5	30.15	6.95	
Q3	46.05	35.475	7.375	
IQR	12.35	9.7	1.25	
Lower Bound	15.175	11.225	4.25	
Upper Bound	64.575	50.025	9.25	

Table A.2 – Full data for the Hypotenuse Method test case.

Trials	X	Y	U	Outlier
1	8.4	43.8	6.1	Yes
2	11.8	33.3	8.1	No
3	14.6	33.4	4.4	No
4	14.5	15.1	6	Yes
5	8.4	41.1	4.5	Yes
6	20.5	32.2	2.9	No
7	3	32	5.6	No
8	2.6	27	7	No
9	23.4	32	7.4	No
10	26.1	30.3	6	No
Averages	13.33	32.02	5.8	
Standard Dev	7.69195034	7.34326903	1.46833239	
Median	13.15	32.1	6	
Q1	8.4	30.725	4.775	
Q2	13.15	32.1	6	
Q3	19.025	33.375	6.775	
IQR	10.625	2.65	2	
Lower Bound	-7.5375	26.75	1.775	
Upper Bound	34.9625	37.35	9.775	

Table A.3 – Full data for the Ambush Method test case.

Trials	X	Y	U	Outlier
1	42.47	25.69	10	No
2	49.48	10.87	2.33	No
3	33.88	27.35	7.62	No
4	55.34	27.64	9.91	No
5	0.33	22.99	5.21	Yes
6	110.62	1.4	7.83	Yes
7	3	4.96	3.37	Yes
8	38.09	5.57	3	No
9	38.61	35.03	7.27	No
10	45.07	15.66	5.8	No
Averages	41.689	17.716	6.234	
Standard Dev	28.8121372	11.0000984	2.61489273	
Median	40.54	19.325	6.535	
Q1	34.9325	6.895	3.83	
Q2	40.54	19.325	6.535	
Q3	48.3775	26.935	7.7775	
IQR	13.445	20.04	3.9475	
Lower Bound	14.765	-23.165	-2.09125	
Upper Bound	68.545	56.995	13.69875	

Table A.4 – Full data for the 32 mm/s conveyor belt speed case.

Trials	X	Y	U	Outlier
1	26.29	0	7.21	No
2	294.95	7.62	7.81	Yes
3	301.44	12.33	3.95	Yes
4	96.96	0	6.79	No
5	27.11	0	6.82	No
6	7.14	0	4.63	No
7	20.25	0	3	No
8	40.38	10.59	4.93	No
9	54.96	1.42	5.43	No
10	28.88	6.93	4.61	No
Averages	89.836	3.889	5.518	
Standard Dev	106.738047	4.7014837	1.48979059	
Median	34.63	0.71	5.18	
Q1	26.495	0	4.615	
Q2	34.63	0.71	5.18	
Q3	86.46	7.4475	6.8125	
IQR	59.965	7.4475	2.1975	
Lower Bound	-63.4525	-11.17125	1.31875	
Upper Bound	176.4075	18.61875	10.10875	

Table A.5 – Full data for the 50 mm/s conveyor belt speed case.

Trials	X	Y	U	Outlier
1	8.4	43.8	6.1	Yes
2	11.8	33.3	8.1	No
3	14.6	33.4	4.4	No
4	14.5	15.1	6	Yes
5	8.4	41.1	4.5	Yes
6	20.5	32.2	2.9	No
7	3	32	5.6	No
8	2.6	27	7	No
9	23.4	32	7.4	No
10	26.1	30.3	6	No
Averages	13.33	32.02	5.8	
Standard Dev	7.69195034	7.34326903	1.46833239	
Median	13.15	32.1	6	
Q1	8.4	30.725	4.775	
Q2	13.15	32.1	6	
Q3	19.025	33.375	6.775	
IQR	10.625	2.65	2	
Lower Bound	-7.5375	26.75	1.775	
Upper Bound	34.9625	37.35	9.775	

Table A.6 – Full data for the 95 mm/s conveyor belt speed case.

Trials	X	Y	U	Outlier
1	15.54	23.7	7.57	No
2				No
3				No
4	0	13.46	5.02	Yes
5	1.48	10.57	8.55	No
6				No
7	59.45	9.52	7.33	No
8				No
9				No
10				No
Averages	19.1175	14.3125	7.1175	
Standard Dev	24.0627787	5.60861558	1.29436036	
Median	8.51	12.015	7.45	
Q1	1.11	10.3075	6.7525	
Q2	8.51	12.015	7.45	
Q3	26.5175	16.02	7.815	
IQR	25.4075	5.7125	1.0625	
Lower Bound	-37.00125	1.73875	5.15875	
Upper Bound	64.62875	24.58875	9.40875	

Table A.7 – Full data for the 500 mm/s and mm/s² robot speed and acceleration limits case.

Trials	X	Y	U	Outlier
1	218.35	0	7.45	Yes
2				No
3	24.17	0.64	6	No
4	30.74	0	5.84	No
5	0	0	6.09	No
6	12.42	0	5.72	No
7	0	2.17	5.19	Yes
8	61.57	0	7.75	Yes
9	20.41	0	6.44	No
10	22.05	6.4	5.79	Yes
Averages	43.3011111	1.02333333	6.25222222	
Standard Dev	64.2727477	2.01807389	0.78869481	
Median	22.05	0	6	
Q1	12.42	0	5.79	
Q2	22.05	0	6	
Q3	30.74	0.64	6.44	
IQR	18.32	0.64	0.65	
Lower Bound	-15.06	-0.96	4.815	
Upper Bound	58.22	1.6	7.415	

Table A.8 – Full data for the 1000 mm/s and mm/s² robot speed and acceleration limits case.

Trials	X	Y	U	Outlier
1	8.4	43.8	6.1	Yes
2	11.8	33.3	8.1	No
3	14.6	33.4	4.4	No
4	14.5	15.1	6	Yes
5	8.4	41.1	4.5	Yes
6	20.5	32.2	2.9	No
7	3	32	5.6	No
8	2.6	27	7	No
9	23.4	32	7.4	No
10	26.1	30.3	6	No
Averages	13.33	32.02	5.8	
Standard Dev	7.69195034	7.34326903	1.46833239	
Median	13.15	32.1	6	
Q1	8.4	30.725	4.775	
Q2	13.15	32.1	6	
Q3	19.025	33.375	6.775	
IQR	10.625	2.65	2	
Lower Bound	-7.5375	26.75	1.775	
Upper Bound	34.9625	37.35	9.775	

Table A.9 – Full data for the 1500 mm/s and mm/s² robot speed and acceleration limits case.

Trials	X	Y	U	Outlier
1	50.41	7.18	8.85	No
2	125.65	0	4.83	Yes
3	52.6	10.21	3.43	No
4	48.15	2.82	6.04	No
5	84.75	0	7.69	Yes
6	29.87	4.44	3.8	No
7	41.32	10.23	3.92	No
8	46.05	0	5.35	No
9	43.88	8.17	7.73	No
10	24.25	9.74	8.4	Yes
Averages	54.693	5.279	6.004	
Standard Dev	28.137647	4.13586134	1.93120791	
Median	47.1	5.81	5.695	
Q1	41.96	0.705	4.1475	
Q2	47.1	5.81	5.695	
Q3	52.0525	9.3475	7.72	
IQR	10.0925	8.6425	3.5725	
Lower Bound	26.82125	-12.25875	-1.21125	
Upper Bound	67.19125	22.31125	13.07875	

Table A.10 – Full data for the 650 mm scanning height case.

Trials	X	Y	U	Outlier
1	15.83	0	9.47	No
2	31.66	0	8.26	No
3	250.21	17.456	26.76	Yes
4	157.49	10.87	6.97	No
5				No
6	5.83	0.01	6.71	No
7	4.04	9.35	9.65	No
8	22.26	8.67	9.39	No
9				No
10	64.97	0	8.13	No
Averages	69.03625	5.7945	10.6675	
Standard Dev	83.1414286	6.29335457	6.17088071	
Median	26.96	4.34	8.825	
Q1	13.33	0	7.84	
Q2	26.96	4.34	8.825	
Q3	88.1	9.73	9.515	
IQR	74.77	9.73	1.675	
Lower Bound	-98.825	-14.595	5.3275	
Upper Bound	200.255	24.325	12.0275	

Table A.11 – Full data for the 680 mm scanning height case.

Trials	X	Y	U	Outlier
1	49.81	0	7.86	No
2	28.14	16.74	7.45	No
3	102	12.37	9.27	No
4	23.99	23.82	9.49	No
5	57.41	0.01	6.8	No
6	65.39	10.62	8.47	No
7	78.12	6.6	2.89	Yes
8	36.31	0	7.11	No
9	32.34	0.002	7.28	No
10	18.24	5.78	6.14	No
Averages	49.175	7.5942	7.276	
Standard Dev	25.3812794	7.83371602	1.77359635	
Median	43.06	6.19	7.365	
Q1	29.19	0.004	6.8775	
Q2	43.06	6.19	7.365	
Q3	63.395	11.9325	8.3175	
IQR	34.205	11.9285	1.44	
Lower Bound	-22.1175	-17.88875	4.7175	
Upper Bound	114.7025	29.82525	10.4775	

Table A.12 – Full data for the 710 mm scanning height case.

Trials	X	Y	U	Outlier
1	24.228	0.005	5.891	Yes
2	10.324	0	7.424	No
3	26.103	0	7.647	No
4	41.159	0	8.6033	Yes
5	29.365	0	7.462	No
6	0	0.3	7.136	Yes
7	22.636	0.001	10.217	Yes
8	17.3825	0.001	5.185	Yes
9	0.001	6.004	7.147	Yes
10	68.926	0	7.402	Yes
Averages	24.01245	0.6311	7.41143	
Standard Dev	19.3149419	1.7931863	1.29478019	
Median	23.432	0.0005	7.413	
Q1	12.088625	0	7.13875	
Q2	23.432	0.0005	7.413	
Q3	28.5495	0.004	7.60075	
IQR	16.460875	0.004	0.462	
Lower Bound	-12.602688	-0.006	6.44575	
Upper Bound	53.2408125	0.01	8.29375	

Table A.13 – Full data for the 740 mm scanning height case.

Trials	X	Y	U	Outlier
1	0	16.01	6.72	Yes
2	11.81	4.49	6.03	No
3	36.82	3.08	5.5	No
4	40.74	6.86	7.11	No
5	0	2.31	6.94	No
6	18.874	0.003	9.74	No
7	24.93	0	8.86	No
8	42.72	1.26	7.86	No
9	16.873	3.062	7.241	No
10	0	4.574	4.825	No
Averages	19.2767	4.1649	7.0826	
Standard Dev	15.8948224	4.43602648	1.40400422	
Median	17.8735	3.071	7.025	
Q1	2.9525	1.5225	6.2025	
Q2	17.8735	3.071	7.025	
Q3	33.8475	4.553	7.70525	
IQR	30.895	3.0305	1.50275	
Lower Bound	-43.39	-3.02325	3.948375	
Upper Bound	80.19	9.09875	9.959375	

Table A.14 – Full data for the 770 mm scanning height case.

Trials	X	Y	U	Outlier
1	8.4	43.8	6.1	Yes
2	11.8	33.3	8.1	No
3	14.6	33.4	4.4	No
4	14.5	15.1	6	Yes
5	8.4	41.1	4.5	Yes
6	20.5	32.2	2.9	No
7	3	32	5.6	No
8	2.6	27	7	No
9	23.4	32	7.4	No
10	26.1	30.3	6	No
Averages	13.33	32.02	6	
Standard Dev	7.69195034	7.34326903	1.46833239	
Median	13.15	32.1	6	
Q1	8.4	30.725	4.775	
Q2	13.15	32.1	6	
Q3	19.025	33.375	6.775	
IQR	10.625	2.65	2	
Lower Bound	-7.5375	26.75	1.775	
Upper Bound	34.9625	37.35	9.775	

Table A.15 – Full data for the 800 mm scanning height case.

Trials	X	Y	U	Outlier
1	62.13	8.74	7.11	No
2				No
3				No
4				No
5				No
6				No
7				No
8				No
9				No
10				No
Averages	62.13	8.74	7.11	
Standard Dev	0	0	0	
Median	62.13	8.74	7.11	
Q1	62.13	8.74	7.11	
Q2	62.13	8.74	7.11	
Q3	62.13	8.74	7.11	
IQR	0	0	0	
Lower Bound	62.13	8.74	7.11	
Upper Bound	62.13	8.74	7.11	

Table A.16 – Full data for the 7 iterations case. Excluded from trend analysis.

Trials	X	Y	U	Outlier
1	27.66	10.14	6.79	No
2	42.12	6.43	5.85	No
3	15.04	0	7.43	Yes
4	21.78	8.61	7.04	No
5	0	7.84	9.82	Yes
6	76.86	10.87	7.25	Yes
7	24.5	7.86	5.86	No
8	15.69	0.1	10.04	Yes
9	37.5	7.35	6.12	No
10	0	2.76	7.33	No
Averages	26.115	6.196	7.353	
Standard Dev	21.4008024	3.70735269	1.4052192	
Median	23.14	7.595	7.145	
Q1	15.2025	3.6775	6.2875	
Q2	23.14	7.595	7.145	
Q3	35.04	8.4225	7.405	
IQR	19.8375	4.745	1.1175	
Lower Bound	-14.55375	-3.44	4.61125	
Upper Bound	64.79625	15.54	9.08125	

Table A.17 – Full data for the 8 iterations case.

Trials	X	Y	U	Outlier
1	20.9	30.798	4.72	Yes
2	17.347	36.092	6.652	No
3	22.27	43.898	7.16	No
4	3.854	36.5406	6.856	No
5	8.998	30.558	7.39	No
6	26.285	28.059	1.567	Yes
7	12.246	26.837	7.707	No
8	9.61	49.069	6.109	Yes
9	34.671	30.592	6.72	No
10	7.624	29.766	6.876	No
Averages	16.3805	34.22096	6.1757	
Standard Dev	9.16358164	6.88210347	1.72128528	
Median	14.7965	30.695	6.788	
Q1	9.151	29.964	6.24475	
Q2	14.7965	30.695	6.788	
Q3	21.9275	36.42845	7.089	
IQR	12.7765	6.46445	0.84425	
Lower Bound	-10.01375	20.267325	4.978375	
Upper Bound	41.09225	46.125125	8.355375	

Table A.18 – Full data for the 9 iterations case.

Trials	X	Y	U	Outlier
1	41.327	21.6	6.35	No
2	21.512	35.281	7	No
3	20.44	21.805	6.609	No
4	54.3175	25.492	3.872	No
5	33.85	36.249	9.022	No
6	43.5	39.075	7.405	No
7	0	35.266	5.367	No
8	36.3187	35.853	7.078	No
9	16.155	27.483	6.888	No
10	24.79	23.235	1.708	Yes
Averages	29.22102	30.1339	6.1299	
Standard Dev	14.9426904	6.4903333	1.94247561	
Median	29.32	31.3745	6.7485	
Q1	20.708	23.79925	5.61275	
Q2	29.32	31.3745	6.7485	
Q3	40.074925	35.71	7.0585	
IQR	19.366925	11.91075	1.44575	
Lower Bound	-8.3423875	5.933125	3.444125	
Upper Bound	69.1253125	53.576125	9.227125	

Table A.19 – Full data for the 10 iterations case.

Trials	X	Y	U	Outlier
1	8.4	43.8	6.1	Yes
2	11.8	33.3	8.1	No
3	14.6	33.4	4.4	No
4	14.5	15.1	6	Yes
5	8.4	41.1	4.5	Yes
6	20.5	32.2	2.9	No
7	3	32	5.6	No
8	2.6	27	7	No
9	23.4	32	7.4	No
10	26.1	30.3	6	No
Averages	13.33	32.02	5.8	
Standard Dev	7.69195034	7.34326903	1.46833239	
Median	13.15	32.1	6	
Q1	8.4	30.725	4.775	
Q2	13.15	32.1	6	
Q3	19.025	33.375	6.775	
IQR	10.625	2.65	2	
Lower Bound	-7.5375	26.75	1.775	
Upper Bound	34.9625	37.35	9.775	

Table A.20 – Full data for the 11 iterations case.

Trials	X	Y	U	Outlier
1	14.453	27.975	8.383	No
2	19	25.84	4.699	No
3	0.022	36.1	5.109	No
4	46.559	23.927	10.849	No
5	43.315	48.919	4.109	No
6	17.735	38.51	3.843	No
7	28.105	24.345	7.007	No
8	1.481	23.024	7.838	No
9	5.196	15.54	6.557	No
10	10.614	37.808	7.98	No
Averages	18.648	30.1988	6.6374	
Standard Dev	15.4357066	9.36029415	2.10871184	
Median	16.094	26.9075	6.782	
Q1	6.5505	24.0315	4.8015	
Q2	16.094	26.9075	6.782	
Q3	25.82875	37.381	7.9445	
IQR	19.27825	13.3495	3.143	
Lower Bound	-22.366875	4.00725	0.087	
Upper Bound	54.746125	57.40525	12.659	

Table A.21 – Full data for the 12 iterations case.

Trials	X	Y	U	Outlier
1	21.262	21.146	4.936	No
2	10.926	25.514	4.772	No
3	43.62	21.664	6.83	Yes
4	11.435	30.309	7.639	No
5	14.852	39.145	7.67	No
6	2.233	28.124	7.41	No
7	14.016	39.978	6.7784	No
8	21.319	27.77	3.684	No
9	62.755	31.336	7.478	Yes
10	8.093	35.082	8.693	No
Averages	21.0511	30.0068	6.58904	
Standard Dev	17.4746345	6.22412108	1.50761783	
Median	14.434	29.2165	7.12	
Q1	11.05325	26.078	5.3966	
Q2	14.434	29.2165	7.12	
Q3	21.30475	34.1455	7.59875	
IQR	10.2515	8.0675	2.20215	
Lower Bound	-4.324	13.97675	2.093375	
Upper Bound	36.682	46.24675	10.901975	

Table A.22 – Full data for the 13 iterations case.

Trials	X	Y	U	Outlier
1	5.529	38.572	7.611	No
2	6.932	20.386	7.081	No
3	0	35.232	4.7515	No
4	20.427	15.16	8.281	No
5	35.378	28.403	4.89	No
6	9.083	22.644	4.12	No
7	21.041	46.071	7.853	No
8	14.068	27.505	4.694	No
9	10.297	41.132	6.025	No
10	26.075	22.887	4.146	No
Averages	14.883	29.7992	5.94525	
Standard Dev	10.205478	9.53301527	1.54399824	
Median	12.1825	27.954	5.4575	
Q1	7.46975	22.70475	4.708375	
Q2	12.1825	27.954	5.4575	
Q3	20.8875	37.737	7.4785	
IQR	13.41775	15.03225	2.770125	
Lower Bound	-	0.156375	0.5531875	
Upper Bound	41.014125	60.285375	11.6336875	

Table A.23 – Full data for the 14 iterations case.

Trials	X	Y	U	Outlier
1	10.34	11.13	7.69	No
2	15.82	10.94	5.42	No
3	7.78	0	8.95	No
4	10.45	0	7.82	No
5	25.65	14.61	4.41	No
6	17.58	9.27	7.36	No
7	9.63	6.4	7.16	No
8	52.91	0	9.63	Yes
9	42.46	3.59	5.72	No
10	0	0	4.47	No
Averages	19.262	5.594	6.863	
Standard Dev	15.743913	5.32790052	1.70526274	
Median	13.135	4.995	7.26	
Q1	9.8075	0	5.495	
Q2	13.135	4.995	7.26	
Q3	23.6325	10.5225	7.7875	
IQR	13.825	10.5225	2.2925	
Lower Bound	-10.93	-15.78375	2.05625	
Upper Bound	44.37	26.30625	11.22625	

Appendix B

Appendix B outlines the complete testing procedure used to evaluate the performance of the vision-guided robotic system. The procedure was applied consistently across all combinations of predictive pickup methods and system parameter variations. The term “arming” refers to the process of initializing the robot and camera to begin tracking for an approaching part. Although the robot did not perform physical pickups during testing, it generated predicted x, y, and u coordinates at the moment of virtual pickup, which were later compared to the actual required pickup coordinates. The step-by-step procedure below provides a detailed description of how these values were collected across all test cases.

Testing Procedure

1. The robot was powered on and initialized using the Epson RC+ software environment.
2. Experimental conditions were configured based on the variable under investigation. For example, when testing the influence of conveyor belt speed, the desired speed was set in the system parameters. Similarly, if scanning height or other factors were being tested, those values were adjusted accordingly.
3. The conveyor belt was turned on and brought to the desired operating speed.
4. The system was armed, placing the robot in a ready state in its starting position and allowing the camera to begin monitoring for incoming parts.
5. A test part was placed manually at the start of the conveyor in a randomized orientation.

6. As the part moved along the conveyor, the robot system tracked its position and orientation. The robot did not attempt an actual pickup during this phase; it only executed predictive tracking routines.
7. At the precise moment the robot would have executed the pickup, the conveyor belt was stopped, and the robot system displayed its predicted pickup coordinates in the X, Y, and U (rotation) directions.
8. The predicted pickup coordinates were recorded.
9. Using the Epson RC+ jog function, the robot end effector was manually moved to the physical location of the part to determine the actual X, Y, and U coordinates required for a successful pickup. These coordinates were displayed live through the jog interface.
10. The actual coordinates were recorded and later compared to the predicted values to assess positional and rotational accuracy.
11. Steps 5–10 were repeated for a total of ten trials per test case.
12. This procedure was repeated for each combination of predictive pickup method and test parameter condition evaluated in the study.

