Automated Vision-Based Measurement and Inspection Techniques for Robotic Drilling of Aerospace Composites

by

Ali Maghami

A thesis submitted to the Faculty of Graduate Studies of
The University of Manitoba
in partial fulfillment of the requirements of the degree of

DOCTOR OF PHILOSOPHY

Department of Mechanical Engineering
University of Manitoba
Winnipeg

Abstract

Drilling of composite structures is one of the most commonly used operations in the manufacturing of aircraft. Currently, most aerospace drilling operations are performed either manually or by extensive programming and fine-tuning of typical industrial robots. Developing fully autonomous robotic drilling cells powered by sensory systems can significantly improve the efficiency, reliability, and accuracy of aerospace composite manufacturing. To this end, the present study aims at developing vision-based techniques to enable two main functionalities: (1) accurate measurement of robot pose and workpiece placement for robot error correction, and (2) automated inspection of drilled holes using deep learning techniques.

Off-the-shelf industrial robots have relatively low absolute accuracy (error~1 mm) and cannot readily satisfy the strict tolerance requirements of aerospace drilling. To provide an optimum resolution, the vision system is calibrated over the robot's workspace using a laser tracker. Circular targets on the spindle are tracked through this system. Different target localization methods for robot motion are compared with laser tracker's accuracy in distance and pose measurements. Online robot error correction using relative motion towards part fixtures is implemented. A similar approach is used to correct errors between cooperative robots in an offline manner. The developed methodologies in both

online and offline error correction methods are tested to demonstrate their accuracy and effectiveness.

Vision-based inspection systems efficiently inspect drilled holes in aerospace composites. However, detecting damage in images is challenging due to dark and textured surfaces in Carbon Fiber Reinforced Polymers (CFRPs). This study presents a fully autonomous system to detect and segment damages and cracks around drilled CFRP holes. The system includes: 1) automated multi-light imaging to illuminate holes in varying light, 2) image processing for hole profile, damage area, and crack segmentation, and 3) a deep Fully Convolutional Network (FCN) with U-Net architecture for pixel-wise image segmentation. The final solution has been tested and its ability to reliably detect the hole profile, damage area, and cracks has been demonstrated.

Vision-based measurement and inspection frameworks proposed in this thesis provide effective and robust tools for the autonomous robotic drilling of aerospace composites.

Contents

Contents ..... **iv**
List of Tables ..... **vii**
List of Figures ..... **viii**
Acknowledgment ..... **xii**
Chapter 1 Introduction **1**
1.1 Background 1
1.2 Problem definition 2
1.3 Thesis objectives 4
1.4 Contributions 6
1.5 Thesis structure 8
Chapter 2 Vision-Based Pose Measurement **11**
2.1 Background 11
2.2 Target localization methods 14
2.3 Circular target localization methods 18
2.3.1 Point ellipse 18
2.3.2 Line ellipse 20
2.3.3 Point-line ellipse 21
2.3.4 Centroiding 22
2.4 Accuracy and precision analysis 22
2.4.1 Physical targets 23
2.4.2 Synthetic dataset 24
2.4.3 Input data 26
2.4.4 Simulation results 28
2.5 Experimental evaluation in robotic applications 30
2.5.1 Experimental setup 30
2.5.2 Camera calibration 31
2.5.3 3D accuracy 33
2.5.4 6D accuracy 36
2.6 Summary 41

Chapter 3 Online Relative Pose Correction 44
3.1 Background 44
3.2 Pose measurement 51
3.2.1 Software interface 52
3.3 Robot-vision calibration 56
3.4 Relative pose correction 58
3.5 $\mathbf{AX} = \mathbf{XB}$ calibration 60
3.5.1 Experimental evaluation 62
3.6 Iterative online relative pose correction 64
3.7 Closed loop relative pose correction 68
3.8 Summary 72
Chapter 4 Offline Relative Pose Correction 74
4.1 Background 74
4.2 Methodology 80
4.2.1 Relative vs absolute accuracy 83
4.2.2 Master-slave relative error 84
4.3 Master-slave absolute accuracy 87
4.4 Experimental data measurement 89
4.4.1 Identification of base frame and flange frame 90
4.4.2 Measurement procedure 91
4.5 Neural network architecture 93
4.5.1 Optimal MLP architecture 95
4.6 Results and discussion 100
4.7 Summary 105
Chapter 5 Vision-Based Inspection Automation 107
5.1 Background 107
5.1.1 Overview of the proposed framework and contributions 111
5.2 Multi-light setup and image fusion 113
5.2.1 Multi-light imaging setup 113
5.2.2 Image fusion for enhancing contrast of the damage area 116
5.2.3 Image fusion for extracting the hole profile 119
5.3 Image processing and segmentation 121

5.3.1 Segmentation of hole area....121
5.3.2 Segmentation of damage area....122
5.3.3 Segmentation of cracks....123
5.4 Deep learning model for autonomous inspection....126
5.4.1 U-Net architecture....126
5.4.2 Training of U-Net for hole inspection....131
5.5 Calculation of the delamination factor....134
5.6 Test results....137
5.7 Summary....139
Chapter 6 Conclusion....141
6.1 Summary of completed work....141
6.2 Future research directions....143
Bibliography....146

List of Tables

Table 2.1: A comparison of the computation time and error RMS obtained from the robot and different target localization methods, as measured by the laser tracker. 41
Table 3.1: Controller gain values..... 69
Table 4.1 Maximum, mean, and standard deviation of master-slave relative errors..... 101
Table 4.2 Maximum, mean, and standard deviation of master errors 103
Table 5.1 Technical specifications of the inspection camera and lens. 115
Table 5.2 Comparison between delamination factors from image processing (IP) and semantic segmentation (U-Net). 138

List of Figures

Figure 1.1: Application of vision measurement system in (a) part localization and (b) drilling navigation 5
Figure 1.2: Implementation of vision-based inspection system 6
Figure 2.1: Stereo optical tracker and circular targets 13
Figure 2.2: (a) Canny edge points, (b) corrected edge points with sub-pixel accuracy, (c) intersecting gradient vectors in a circle, and (d) lines perpendicular to gradient vectors forming a line ellipse. 18
Figure 2.3: Schematics and dimensions of (a) active and (b) passive targets along with (c) final test targets. 24
Figure 2.4: (a) Active, passive, and synthetic ellipses and (b) their pixel intensity profiles. 25
Figure 2.5: Transition edge between the target and the background in gradient magnitude of images from (a) active, passive, and synthetic targets along with (b) their normalized cross-sectional profiles and selected regions under red highlight. 28
Figure 2.6: (a) Mean and (b) standard deviation of center position errors of different methods in different levels of noise. 29
Figure 2.7: (a) Mean and (b) standard deviation of center position errors of different methods in different levels of blur. 30
Figure 2.8: (a) IR stereo vision system, Leica laser tracker and (b) the Universal Robots UR5e robot with the measurement end-effector. 31
Figure 2.9: Measurement volume in robot workspace, calibration grid points, and calibration end-effector. 33
Figure 2.10: A comparison between (a) mean and (b) standard deviation of 3D distance errors from different target localization methods with different number of passive targets. 35
Figure 2.11: (a) T-Mac and passive targets attached to the robot flange in (b) different target configurations. 37
Figure 2.12: Calibration process between retro-reflective targets and T-Mac probe. 38
Figure 2.13: RMS errors of robot 6D pose estimation of different target detection methods with different number of targets. 39
Figure 3.1: Measured pose of drilling spindle in the optical tracker (world) frame. 52
Figure 3.2: Developed software interface for vision measurement system and robot communication. 54

Figure 3.3: (a) Drilling spindle with targets and (b) assigned frame to the rigid body using the developed software and visualized in RoboDK. 55
Figure 3.4: Schematic of connections between different elements. 56
Figure 3.5: Kinematic chain between vision system and robot. 57
Figure 3.6: Relative pose error measurement between the drilling spindle and fixture. 59
Figure 3.7: Schematics of $\mathbf{AX = XB}$ calibration between the spindle and flange frames. .. 61
Figure 3.8: Simulation dataset of calibration poses inside the robot workspace. 63
Figure 3.9: Mean error, em, obtained from experiment with different methods and different number of measurements. 64
Figure 3.10: Flowchart of error correction algorithm. 65
Figure 3.11: Relative positional and orientational measurement noise level in robot base. 67
Figure 3.12: Relative positional and orientational error correction in an iterative approach. 68
Figure 3.13: Relative pose error correction control loop. 69
Figure 3.14: Relative positional and orientational error correction using PID controller. .. 71
Figure 3.15: Relative positional and orientational error correction in 6D error correction using PID controller. 72
Figure 4.1: Examples of dual-robot cooperative system in (a) indirect and (b) direct geometric coupling modes 76
Figure 4.2: Important frames in a dual-robot system forming a kinematic chain 82
Figure 4.3: Schematics of nominal and measured robot postures and their corresponding flange frames 86
Figure 4.4: Slave flange compensation relative to the master flange 87
Figure 4.5: Schematics of master flange pose error calculation 89
Figure 4.6: Experimental master-slave setup including two KUKA KR300 R2500 ULTRA robots, Leica LT800 laser tracker, measurement tooling, and SMR targets 90
Figure 4.7: Identification process for robot base frame and flange frame in the world frame 91
Figure 4.8: Schematics of data measurement volume and the pair of measurement poses . 93
Figure 4.9: Schematics of ANN aechitectures for prediction of (a) master-slave relative error $\boldsymbol{\varepsilon rel}$. and (b) master absolute error $\boldsymbol{\varepsilon m}$ 95
Figure 4.10: Data preparation and splits for training, test and 5-fold cross-validation 96
Figure 4.11: 5-fold cross-validation results of (a) master-slave relative pose error, $\boldsymbol{\varepsilon rel}$, and (b) master error, $\boldsymbol{\varepsilon m}$, before geometric calibration subject to different number of layers and neurons 98
Figure 4.12: 5-fold cross-validation results of (a) master-slave relative pose error, $\boldsymbol{\varepsilon rel}$., and (b) master error, $\boldsymbol{\varepsilon m}$, after geometric calibration subject to different number of layers and neurons 99

Figure 4.13: 5-fold cross-validation results of (a) master-slave relative pose error, $\boldsymbol{\varepsilon rel}$., and (b) master error, $\boldsymbol{\varepsilon m}$, before geometric calibration subject to different batch sizes and epoch numbers 99
Figure 4.14: 5-fold cross-validation results of (a) master-slave relative pose error, $\boldsymbol{\varepsilon rel}$., and (b) master error, $\boldsymbol{\varepsilon m}$, after geometric calibration subject to different batch sizes and epoch numbers 100
Figure 4.15: Simulation results for master-slave relative position error on test data for (a) uncalibrated and geometrically calibrated cases and their distributions in (b) and (c), respectively 102
Figure 4.16: Simulation results for master-slave relative orientation error on test data for (a) uncalibrated and geometrically calibrated cases and their distributions in (b) and (c), respectively 102
Figure 4.17: Simulation results for master position error on test data for (a) uncalibrated and geometrically calibrated cases and their distributions in (b) and (c), respectively 104
Figure 4.18: Simulation results for master orientation error on test data for (a) uncalibrated and geometrically calibrated cases and their distributions in (b) and (c), respectively 104
Figure 5.1: A schematic of the proposed method based on U-Net. 113
Figure 5.2: Overview of the multi-light imaging system..... 114
Figure 5.3: (a) Four captured inspection images from the hole (diameter: 6 mm), (b) fused image ImgFD (proposed) in comparison with (c) image captured with dark-field lighting (conventional). 116
Figure 5.4: Light reflections on the damage surface. 118
Figure 5.5: Comparison between gray level profile of the center column pixels in the proposed fused image (Figure 5.3 (b)) and conventional dark-field lighting (Figure 5.3 (c)). 119
Figure 5.6: (a) Light rays hitting the inner wall of the hole, (b) fused images for extracting the hole profile, ImgFH, (c) gray level profile of the center column pixels in ImgFH. ... 121
Figure 5.7: (a) Fused image ImgFH, (b) inverted and binarized image, (c) binary image after closing and removing noise, (d) fused image with minimum enclosing circle. 122
Figure 5.8: (a) Fused image ImgFD, (b) binarized image, (c) binary image after adding the hole area and closing filter, (d) fused image with outlined damage area. 123
Figure 5.9: (a) Fused image ImgFD, (b) image after subtracting the damage contour and applying Sobel filters, (c) binarized image after median blur filter, (d) fused image with segmented cracks (red). 124
Figure 5.10: Architecture of the developed U-Net neural network for autonomous hole segmentation. 127
Figure 5.11: Training and validation (a) losses and (b) F1 scores (Dice) for each epoch. 133

Figure 5.12: Results for the segmentation of a sample validation hole at different epochs. 133
Figure 5.13: Schematic of drilling induced damages: (a) uniform damage area, (b) damage area with fine cracks. 135
Figure 5.14: (a)Fused images of test holes, (b) segmented damages obtained from image processing, and (c) fused image with prediction outlines. All dimensions are in pixels... 139

Acknowledgment

I would like to express my sincere gratitude to a number of individuals and organizations who have played a pivotal role in the successful completion of this thesis.

Foremost among them is my research advisor, Dr. Matt Khoshdarregi. His unwavering guidance, mentorship, and support have been instrumental in shaping my research journey. I am profoundly grateful for his invaluable insights, which have enriched my academic understanding and helped me overcome various challenges.

I extend my heartfelt appreciation to my advisory committee members, Dr. Ahmed Ashraf and Dr. Raghavan Jayaraman. Their insightful feedback and critiques have significantly enhanced this thesis.

Gratitude is also due to the University of Manitoba Graduate Fellowship (UMGF) and Mitacs Accelerate program for their financial support.

I am indebted to Dr. Ehsan Jalayeri for his technical expertise and steadfast assistance. Also, the camaraderie and intellectual stimulation among my peers in the Intelligent Digital Manufacturing Lab have been invaluable.

Special thanks to Dr. Reyhaneh Salehi-Tabar for her constant support and encouragement in both my academic and personal life.

My family deserves special recognition for their patience, unconditional love, and consistent support. The sacrifices made by my parents have been the cornerstone of my success. Finally, I extend my deepest appreciation to my colleague, best friend, and beloved wife, Meshkat Tabar. Her consistent encouragement and belief in my abilities have been crucial to my academic achievements. Without her, this endeavor would not have been possible.

To Meshkat, my guiding light,

and to Neil, our sunrise.

Chapter 1

Introduction

1.1 Background

Composite materials have been a primary choice in the structural design of new-generation commercial aircraft due to their high stiffness-to-weight ratio [1]. Boeing 787 and Airbus A350 XWB are good examples of this approach with over 50% of the total weight being made of composites [2]. Carbon fiber reinforced polymers (CFRPs) are one the most widely used types of composites in the aerospace industry.

Aircraft body parts, e.g., fuselage and wing, are made of large panels mounted on supporting frames. In most cases, structural elements are assembled using mechanical fasteners such as rivets and bolts. In these structural assemblies, reliable joints under high tensile, compressive, shear and fatigue loadings are

required [3, 4]. The assembly process requires a large number of drilled holes for riveting and bolting operations. For instance, a typical aircraft wing can have up to 5000 holes, and a transport aircraft has approximately 1 to 2 million holes in total [5]. Due to the independent deformations of assembled elements, these holes pose a high potential for damage initiation and progression as they create large stress concentration zones [6]. Therefore, the drilling operation must satisfy critical geometric tolerances and quality inspection to ensure the effectiveness and safety of the final assembly.

Traditionally, drilling and inspection of drilled holes have relied on manual labor. Manual operations increase the production cost particularly in drilling and inspection of large and complex panels [5]. Industrial robots have been increasingly adopted to enhance the efficiency and reliability of production lines by automating time-consuming tasks such as drilling and inspection [7]. However, typical robots need to be programmed and calibrated for each specific task, followed by extensive program touch-ups, in order to meet the required tolerances in aerospace manufacturing.

1.2 Problem definition

Positioning accuracy of robots is normally reported in two forms: absolute accuracy and repeatability (precision). Industrial robots are able to repeatedly position the end-effector with errors smaller than 100 microns. However, their absolute accuracy is typically lower with errors in the order of millimeter(s) [8]. These

errors can be due to assembly errors, temperature variations, kinematic errors, and elastic deflections in the robot structure [9, 10]. The acceptable tolerance in the automotive industry for operations such as spot welding is of the order of 1 millimeter. In the aerospace industry, on the other hand, the tolerances are tighter by a factor of at least 10, i.e. 0.1 mm [11]. Sources of error such as link deflection and thermal expansion are not detectable at the robot's joint encoders [12]. Therefore, position and orientation errors can quickly accumulate at the Tool Centre Point (TCP).

Aerospace composite materials are usually designed in large dimensions with complex curved surfaces for aerodynamic efficiency [13]. Manual inspection of drilled holes can be challenging in terms of repeatability of position and orientation of inspection probe when the structure is large and complex. Industrial robots can offer reliable automated solutions with high-speed scans and high repeatability to minimize the downtime of inspection processes. However, due to the dark and textured background of composites (mainly CFRPs), quality evaluation of drilled holes is a challenging task for vision systems [14]. Separating the hole profile from the background and measuring parameters such as damage and crack area become even more challenging when lighting conditions are not consistent over a large composite part.

In order to reduce the challenges in robotic drilling and inspection, sensory systems can be integrated into robotic cells to provide feedback from the environment. Cameras and imaging sensors have been widely used for industrial

quality control and inspection [15]. They can also be used to measure robot motions and improve the accuracy of robotic operations.

1.3 Thesis objectives

The motivation behind this study originates from the prevalent reliance on manual labor and the time-consuming nature of robot programming within conventional aerospace drilling practices. The overarching objective is to transition towards fully autonomous robotic drilling systems, fortified with sensory capabilities, gaining advancements in efficiency, robustness, and accuracy. In pursuit of this objective, our goal is to leverage vision-based techniques to augment the precision and autonomy of industrial robots used in drilling and inspecting aerospace composites.

To realize this vision, we developed an optical measurement system and an inspection setup, subjecting them to thorough performance evaluations. The study delves into the development and application of systematic models and governing frameworks for optical systems, with the ultimate aim of achieving accurate robot control and facilitating effective inspection protocols.

In addressing the accuracy limitations of off-the-shelf industrial robots in aerospace drilling, our study proposes a vision-based measurement solution. The primary objective of this system is to accurately measure the spatial coordinates of the robot's drilling spindle. Calibration of the vision system takes place within the robot's workspace, facilitated by a laser tracker. Circular targets affixed to the

drilling spindle are tracked through the calibrated vision system, with a comprehensive evaluation of various target localization methods, emphasizing their performance in robot motion tracking.

The optical measurement system serves as a pivotal component in achieving high accuracy in robotic drilling of composites. The developed optical system not only provides an optimal solution for measurings and correcting the robot's pose in drilling tasks but also introduces the potential for a digital assembly environment for part localization, thereby reducing fixturing time for each part. Figure 1.1 illustrates the implementation schematics of an optical measurement system for part localization and drilling navigation.

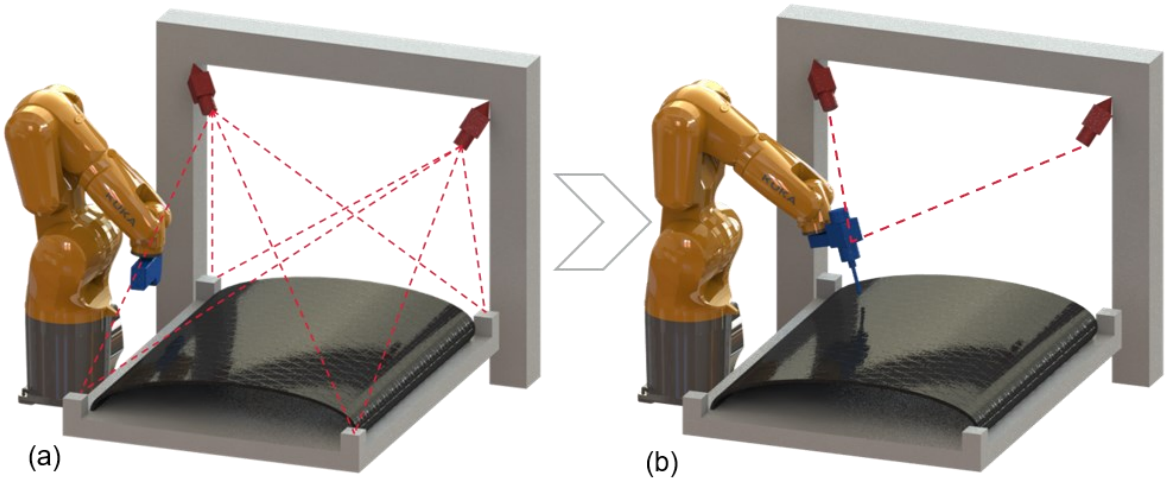

Figure 1.1: Application of vision measurement system in (a) part localization and (b) drilling navigation

A promising avenue for the efficient evaluation of drilled holes in aerospace composites lies in vision-based inspection systems. However, the complex nature of this task, exacerbated by the characteristics of Carbon composites, necessitates

innovative approaches. Our proposed solution introduces an autonomous system designed for the detection and segmentation of damage and cracks around drilled holes in CFRP composites. The proposed vision-based inspection solution, as depicted in Figure 1.2, addresses the challenges associated with automatically inspecting drilled holes. By integrating an imaging sensor with a multi-light system, it reliably extracts features from drilled holes. Additionally, the system incorporates automated algorithms capable of accommodating the arbitrary orientation of fibers in the material texture, a crucial requirement. Employing robust and fast algorithms can significantly enhance the inspection speed for each hole, consequently reducing the overall downtime needed for inspecting thousands of holes in a part.

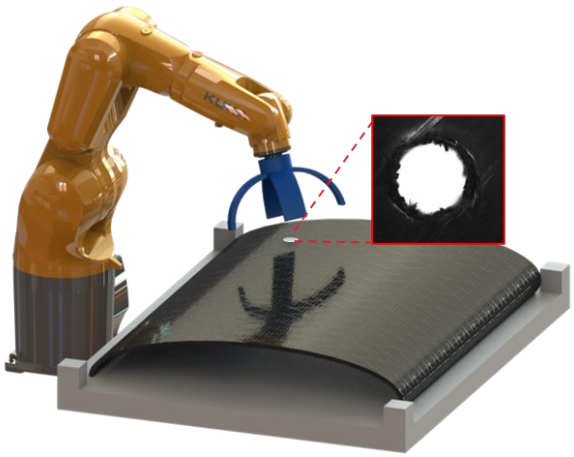

Figure 1.2: Implementation of vision-based inspection system

1.4 Contributions

This study develops vision-based measurement and inspection frameworks along with underlying models to enhance the autonomy and accuracy of robotic drilling

systems. The proposed systems and frameworks remove or relax the current limitations and weaknesses in robotic implementation of drilling and inspection tasks in the aerospace industry. Along with the development of the vision systems, new models and systematic methodologies are proposed to enhance the performance and scalability of the proposed systems.

This study presents several novel and unique contributions to the field of robotic manufacturing including:

- High-accuracy drilling:
    - Development and calibration of a tracker system for achieving high accuracy measurements over the workspace of industrial robots.
    - Comparison of methodologies for achieving sub-pixel accuracy in extraction of targets in optical tracking of robotic applications.
    - Methods for localizing parts and drilling spindle in the robot's workspace and correcting the robot errors with respect to the part.
    - Novel calibration method for collaborative systems based on the decoupling of relative errors.
    - An error compensation approach based on Artificial Neural Networks (ANNs) for relative and absolute error compensation in dual-robot systems.
- Autonomous inspection:
    - Design and development of a multi-light imaging end-effector for automatic vision-based inspection of drilling-induced damages.

- Development of a multi-light technique based on a pixel-level image fusion algorithm for the extraction of hole profile without the need for back lighting.
- Development of an image processing methodology that is capable of assessing drilling-induced damage in CFRPs and extracting the hole profile, damage area and cracks.
- Implementation of an efficient fully convolutional neural network with U-Net architecture to increase the robustness of image processing against directional properties of CFRPs and crack angles.

The findings and contributions of this thesis have been published/submitted across four journal articles and one conference paper.

1.5 Thesis structure

The main body of the present thesis is organized in four chapters, each comprising one of the published/submitted papers. Each chapter includes a comprehensive literature review related to the studied topic.

Chapter 2 provides the background, methodologies, and process of developing the optical tracker system for measuring robot pose. This chapter studies different target localization methods and their effect on the accuracy and precision of robot pose measurements. Chapter 2 has been submitted as the following paper [16]:

- **Maghami, A.,** Khoshdarregi, M.: Target Localization Approaches for High-Precision Optical Tracking in Robotic Applications. J. Intell. Robot. Syst. (Submitted in May 2023).

Chapter 3 uses the developed optical tracker system to localize parts in the robot workspace and correct robot's pose with respect to the part fixturing. Chapter 3 has been submitted as the following paper [17]:

- **Maghami, A.**, Khoshdarregi, M.: Online Correction of Relative Errors Between Robot and Workpiece Using an Optical Tracking Solution. Int. J. Robot. Autom. (Submitted in Aug 2023)

Chapter 4 focuses on error correction of dual robot cells for manipulation of large aircraft structures. The developed models have been published in [18]:

- **Maghami, A.**, Imbert, A., Côté, G., Monsarrat, B., Birglen, L., Khoshdarregi, M.: Calibration of Multi-Robot Cooperative Systems Using Deep Neural Networks. J. Intell. Robot. Syst. 107, 55 (2023). https://doi.org/10.1007/s10846-023-01867-6.

Chapter 5 presents the developed robot end-effector and methodologies for vision-based inspection of drilled holes. The work presented in this chapter has been published as [19, 20]:

- **Maghami, A.**, Salehi, M., Khoshdarregi, M.: Automated vision-based inspection of drilled CFRP composites using multi-light imaging and deep learning. CIRP J. Manuf. Sci. Technol. 35, 441–453 (2021). https://doi.org/10.1016/j.cirpj.2021.07.015.

- Hernandez, A., **Maghami, A.**, Khoshdarregi, M.: A Machine Vision Framework for Autonomous Inspection of Drilled Holes in CFRP Panels. 2020 6th Int. Conf. Control. Autom. Robot. ICCAR 2020. 669–675 (2020). https://doi.org/10.1109/ICCAR49639.2020.9108000.

Finally, Chapter 6 summarizes the findings and contributions of the thesis and presents potential directions for continuation of this research.

Chapter 2

Vision-Based Pose Measurement

2.1 Background

Vision-based pose tracking systems are high-accuracy measurement devices that use images to localize positioning targets and provide accurate 6-Degree-of-freedom (6-DoF) coordinates [21, 22]. Due to their relatively low cost compared to industrial robots, a few of these systems can be integrated into a robotic cell from different viewing angles to offer an optimal solution for accurate pose measurement, error correction, and robot guidance [23, 24]. These measurement systems can also be used to synchronize a physical robot cell and its digital twin (simulation model) for monitoring and control purposes [25].

To achieve a high accuracy in measurement, measurement systems use positioning targets to mark the points of interest. Accurate detection of these

positioning targets in the images is one of the main stages in the measurement process. Flat circular targets are commonly used in these systems due to their detection accuracy and robustness [26]. Figure 2.1 shows a stereo optical tracker and circular targets used for measuring robot's pose. These targets are typically used in two forms of active and passive targets. Active targets are normally made of infrared (IR) or near-IR light-emitting diodes (LEDs) [27]. Due to the radiometric characteristic of IR light and with the help of near-IR-bandpass imaging filters, the effect of environmental light can be reduced which results in high contrast images of the targets. Although active targets have the advantage of high visibility in measurements, they are difficult to implement due to wiring requirements and their large dimensions [28]. Passive targets, on the other hand, are made of retro-reflective materials deposited on a thin plastic film and can be manufactured as circular stickers [29, 30]. These targets reflect the light in almost the same direction of incident. On-axis illumination of these targets by installing IR LEDs close to the camera results in high contrast images of the targets. The main advantage of using passive targets is their light weight, which makes it possible to use many of them on a robot without limiting the robot's motion.

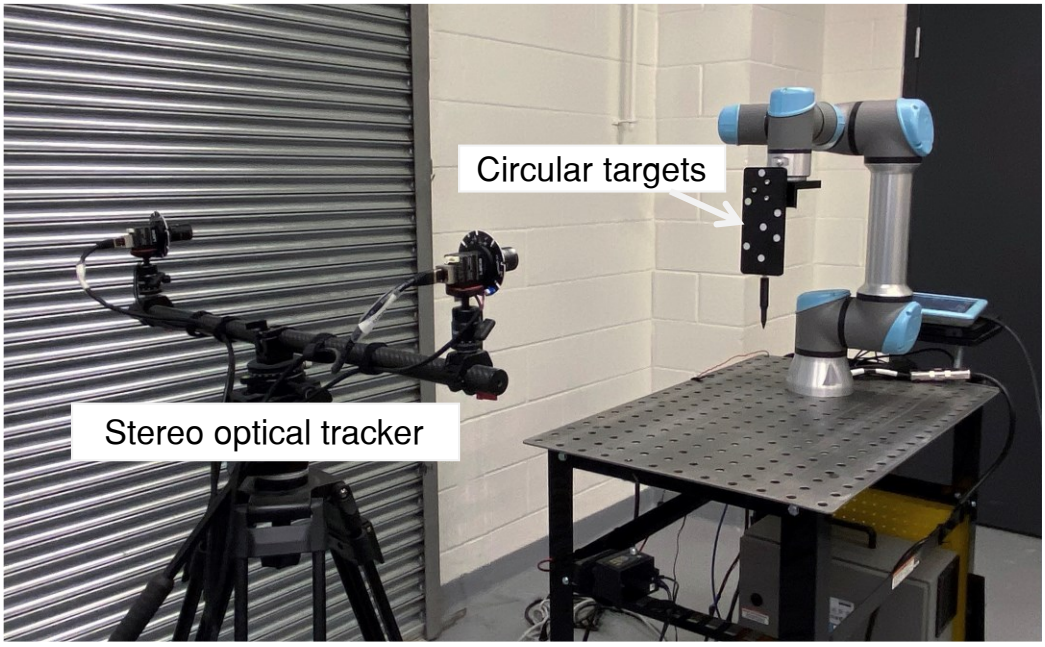

Figure 2.1: Stereo optical tracker and circular targets

In general, vision-based tracking systems use triangulation, based on the images of targets from different angles, to find the three-dimensional (3D) coordinates of the targets. The projective transformation of a planar circular target to the image plane can be approximated as an ellipse. Ellipse center coordinates with sub-pixel accuracy are used for triangulating the center coordinates of the target in 3D space.

Accurate target center measurement is a key element of pose measurement in robotic applications. The succeeding steps in 3D coordinate triangulation and six-dimensional (6D) pose estimation rely on the results of target center location from images. While many studies in the past have investigated the image processing aspects of target localization, the performance of these methods have not yet been compared in detail in the context of robot pose measurement. Here, we investigate

aspects such as augmenting and sub-pixel pre-processing of data points to improve the accuracy of target center location in passive and active circular targets. We compare different methods in the context of industrial robotic applications, where robustness against noise and blur is important for high-precision measurements. Finally, we test the effect of different methods in terms of accuracy and precision in measuring robot's motion in 3D distance and 6D pose estimation.

2.2 Target localization methods

In general, optical tracking systems use triangulation, based on the images of targets from different angles, to find the 3D coordinates of the targets. The projective transformation of a planar circular target to the image plane can be approximated as an ellipse. Ellipse center coordinates with sub-pixel accuracy are used for triangulating the center coordinates of the target in 3D space. The commonly used algorithms for center calculation can be divided into two categories; centroiding methods and ellipse fitting methods [27, 31]. Centroiding methods are relatively simple and efficient algorithms. Binary centroid and gray-weighted centroid are two of the commonly used methods in photogrammetry applications. These methods are usually applied on images obtained from a selected threshold. The gray-weighted centroid method uses the grayscale value of each pixel as the weight and can result in higher accuracy in center location as opposed to the binary centroid which gives equal weight to all selected pixels.

In ellipse fitting methods, the full ellipse geometry that fits best to the ellipse points is found rather than finding only the center of the ellipse. The ellipse points in images are usually pixels extracted using edge detection algorithms such as Canny [32]. Ellipse fitting algorithms are more computationally complex than centroiding methods. In measurement systems for robotic applications, we need to consider not only accuracy but also time efficiency. Based on these requirements, here, we narrow down our options in different classes of ellipse fitting. We mainly focus on fully deterministic methods, where the data model and the operators are both deterministic, in contrast with statistical models and voting operators such as RANSAC [33, 34]. Deterministic methods can be classified into two main categories: clustering (Hough transform), and least-squares methods.

Hough-based methods [35, 36] can provide robustness against outliers. However, their high computational and memory load makes them unsuitable for online measurement applications. Least-squares methods are mainly based on the definition of a distance (cost) function and finding the minimum value of this function, i.e., best fit. The distance function represents the error between the fitted ellipse and input data points. Based on the definition of the distance parameter used in these methods, they can be classified into geometric and algebraic approaches. In the geometric approach, the distance is a geometrical parameter, such as Euclidean distance [37, 38]. In this case, the cost function with respect to the ellipse parameters becomes non-linear and requires iterative methods for minimization, and hence, it is computationally expensive. In algebraic fitting of

ellipses, on the other hand, an algebraic distance is used [39]. This approach usually results in a linear cost function with respect to ellipse parameters. These formulations can result in an efficient closed form solution having selected a quadratic constraint function for the minimization process [39]. One of the main disadvantages of these methods is the lack of invariance in the cost function, which can result in forms of fitted curve other than the desired ellipse, such as hyperbola or parabola [37]. To solve this problem, Fitzgibbon et. al. [39] incorporated the ellipticity constraint $4ac - b^2 > 0$ into the minimization constraint in a quadratic form and found an ellipse-specific closed form solution. Since IR-bandpass images of targets are bimodal images with minimal background noise and complexity, and due to the computational time limitations in robotic applications, in this study we focus on non-iterative algebraic least-squares methods and investigate different variations of them in the context of industrial optical measurement systems.

Many research studies have been conducted in the last decades on different formulations of least-squares approach for ellipse fitting. Here, we try to investigate the performance of different variations and combinations of these methods in IR-bandpass images from a robot pose measurement system. From the viewpoint of input data, we divide the algebraic least-squares approach into two types of point and line ellipse fitting. In fitting an ellipse using points, i.e., a point ellipse, the edge pixels are used as sampled input data and are usually found by exploiting the local maxima and using edge detectors such as Canny. The accuracy

of ellipse fitting in these methods depends on the accuracy of sampled pixels at the edge. The accuracy of edge pixels directly depends on the resolution of ellipse image, which is not high in the case of imaging a small circular target in a relatively large field of view, see Figure 2.2 (a). Therefore, preprocessing the input data and finding the edge points with sub-pixel accuracy can improve the accuracy of ellipse estimations. Heikkila [40] proposed a sub-pixel ellipse boundary detection algorithm that convolves each pixel on the edge and corrects its location using the local gradient information in the neighboring pixels [40], see Figure 2.2 (b).

The addition of edge correction steps adds to the complexity of the solution and the computation time. However, directly exploiting the gradient information in an operator can increase the accuracy of estimations and at the same time reduce the solution time for each ellipse. Forstner et al. [41] proposed one of the first operators of this kind for the image of a circle and its surrounding edge pixels (Figure 2.2 (c)). Ouellet et al. [42] proposed an operator that uses perpendicular lines to gradient vectors for fitting a line (dual) ellipse, see Figure 2.2 s(d). Ouellet's implementation of gradient field with perpendicular lines reduced the angular error caused by noise compared to Forstner's method which directly uses gradient vectors. Later, Patraucean et al. [43] augmented Ouellet's solution by the simultaneous usage of gradient and point data, here called point-line method. They showed when the data is sampled along incomplete (occluded) ellipses,

especially when the ellipse is small, simultaneous solution of positional and gradient data improves the accuracy of fitting results.

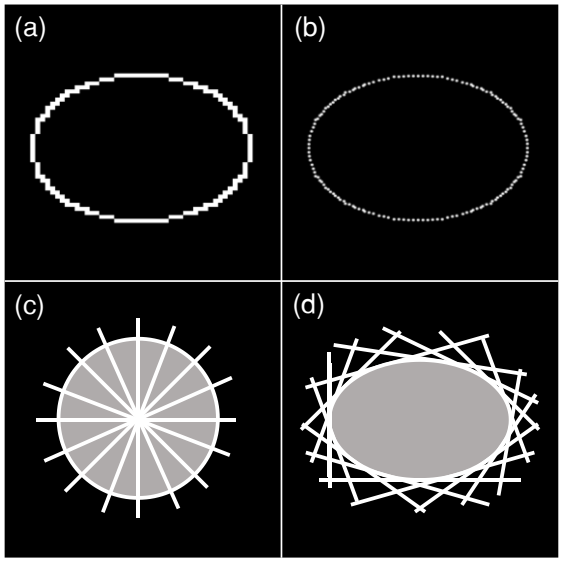

Figure 2.2: (a) Canny edge points, (b) corrected edge points with sub-pixel accuracy, (c) intersecting gradient vectors in a circle, and (d) lines perpendicular to gradient vectors forming a line ellipse.

2.3 Circular target localization methods

A conic section $\mathcal{C}$ is the set of all points satisfying the bivariate quadratic polynomial

$$Ax^2 + Bxy + Cy^2 + Dx + Ey + F = 0, \tag{2.1}$$

where A, B, C, D, E, F are the conic section parameters, which from here on are referred to as vector $\boldsymbol{\theta}_{\mathcal{C}} = [A, B, C, D, E, F]^T$, and x, y are the Cartesian point coordinates. The following subsections briefly review the theories of the target localization methods investigated in this work.

2.3.1 Point ellipse

In the homogenous coordinates, where the point $\mathbf{p} = (x, y, 1)^{\mathrm{T}}$, equation (2.1) can be written in the matrix form as

$$\mathbf{p}^{\mathbf{T}}\mathbf{C}_{\mathcal{C}}\mathbf{p} = 0, \tag{2.2}$$

where $\mathbf{C}_{\mathcal{C}}$ is the matrix of coefficients for the conic section $\mathcal{C}$ and can be found as

$$\mathbf{C}_{\mathcal{C}} = \begin{bmatrix} A & B/2 & D/2 \\ B/2 & C & E/2 \\ D/2 & E/2 & F \end{bmatrix}. \tag{2.3}$$

Let $\mathcal{C}$ be an ellipse represented by equation (2.2), with $\mathbf{p}$ as the homogenous coordinates of a data point with (x, y) coordinates and $\mathbf{C}_{\mathcal{C}}$ as the matrix of coefficients. Now, having n points to fit with ellipse $\mathcal{C}$, the problem is to minimize the distance

$$E = \sum_{i=1}^{n} \left(F(\mathbf{p}_i, \boldsymbol{\theta}_{\mathcal{C}})\right)^2 = \|\mathbf{D}\,\boldsymbol{\theta}_{\mathcal{C}}\|^2, \tag{2.4}$$

where $F(\mathbf{p}_i, \boldsymbol{\theta}_{\mathcal{C}})$, the cost function, is the algebraic distance of the point $\mathbf{p}_i$ to the fitted ellipse and

$$\mathbf{D} = \begin{bmatrix} x_1^2, & x_1 y_1, & y_1^2, & x_1, & y_1, & 1 \\ & & \vdots & & & \\ x_n^2, & x_n y_n, & y_n^2, & x_n, & y_n, & 1 \end{bmatrix} \tag{2.5}$$

is the so-called design matrix with $n \times 6$ dimensions [39]. To avoid trivial solutions and recognize that any multiply of $\boldsymbol{\theta}_{\mathcal{C}}$ denotes the same solution, the minimization problem is subject to the constraint function $h(\boldsymbol{\theta}_{\mathcal{C}}) = 0$ which links the conic section parameters together for a desired output geometry, in this case ellipses.

The direct ellipse-specific constraint can ensure ellipticity by embedding $4ac - b^2 > 0$ and can be used in the quadratic form of

$$h(\boldsymbol{\theta}_{\mathcal{C}}) = \boldsymbol{\theta}_{\mathcal{C}}^{\mathrm{T}}\mathbf{N}\boldsymbol{\theta}_{\mathcal{C}} - 1, \tag{2.6}$$

where $\mathbf{N}$ is the 6×6 constraint matrix defined as

$$\mathbf{N} = \begin{bmatrix} 0 & 0 & 2 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 & 0 & 0 \\ 2 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}. \tag{2.7}$$

A closed-form solution exists for this problem and can be solved using the method of Lagrange multipliers [39]. The same constraint can also be used for fitting a line (dual) ellipse [42].

In point ellipse methods, where the input data are pixel points such as Fitzgibbon's [39], we can improve the fitting results by correcting the edge points using local curvature information. Although this process increases the computation and complexity of the estimator, we investigate the effect of this approach on the input data for point ellipse method by Fitzgibbon [39] and refer to it as 'corrected point ellipse'. The details about the edge correction algorithm can be found in [40] by Heikkila.

2.3.2 Line ellipse

The dual of a conic section can also be represented using the line $\mathbf{l} = (a, b, c)^{\mathrm{T}}$ that is tangent to the dual conic satisfying the equation [42]

$$\mathbf{l}^{\mathbf{T}}\mathbf{C}_{\mathcal{C}}^{*}\mathbf{l} = 0, \tag{2.8}$$

where $\mathbf{C}_{\mathcal{C}}^{*}$ is the inverse of $\mathbf{C}_{\mathcal{C}}$, called dual matrix of coefficients. Here, the resulting ellipse represented with lines is called line ellipse in contrast with the ellipse

represented by points, i.e., point ellipse. An example of a line ellipse can be seen in Figure 1 (d).

Given a point $\mathbf{p}_i = (x_i, y_i)^{\mathrm{T}}$ and image gradient vector $\mathbf{g}_i = \left(\frac{\partial I}{\partial x}(x_i, y_i), \frac{\partial I}{\partial y}(x_i, y_i)\right)^{\mathrm{T}}$ on the ellipse $\mathcal{C}$, based on pole-polar duality, when $\mathbf{p}_i$ is located on the ellipse $\mathcal{C}$, its polar line ($\mathbf{l}_i$) is tangent to $\mathcal{C}$ at the point $\mathbf{p}_i$. The line $\mathbf{l}_i$ is then orthogonal to the gradient vector $\mathbf{g}_i$, and a vector orthogonal to $\mathbf{g}_i$ that lies on $\mathbf{l}_i$ can be found in homogenous coordinates as $\mathbf{g}_i^{\perp} = \left(g_{ix}^{\perp}, g_{iy}^{\perp}, 0\right)^{\mathrm{T}} = \left(-\frac{\partial I}{\partial y}(x_i, y_i), \frac{\partial I}{\partial x}(x_i, y_i), 0\right)^{\mathrm{T}}$ [42]. In the case of line ellipse, having n lines to fit with ellipse $\mathcal{C}$, the problem is to minimize the distance

$$E = \sum_{i=1}^{n} \left(G(\mathbf{p}_i, \mathbf{g}_i, \boldsymbol{\theta}_{\mathcal{C}})\right)^2 = \|\mathbf{Q}\,\boldsymbol{\theta}_{\mathcal{C}}\|^2, \tag{2.9}$$

where $G(\mathbf{p}_i, \mathbf{g}_i, \boldsymbol{\theta}_{\mathcal{C}})$, is the cost function and

$$\mathbf{Q} = \begin{bmatrix} g_{1x}^{\perp} x_1, \frac{1}{2} g_{1x}^{\perp} y_1 + \frac{1}{2} g_{1y}^{\perp} x_1, & g_{1y}^{\perp} y_1, \frac{1}{2} g_{1x}^{\perp}, \frac{1}{2} g_{1y}^{\perp}, 0 \\ & \vdots & \\ g_{nx}^{\perp} x_1, \frac{1}{2} g_{nx}^{\perp} y_1 + \frac{1}{2} g_{ny}^{\perp} x_1, & g_{ny}^{\perp} y_1, \frac{1}{2} g_{nx}^{\perp}, \frac{1}{2} g_{ny}^{\perp}, 0 \end{bmatrix}. \tag{2.10}$$

is the design matrix with $n \times 6$ dimensions.

2.3.3 Point-line ellipse

The positional equations in point ellipse can also be augmented with the gradient information in line ellipse by stacking equations (2) and (8), as proposed in [43]. In

this case, the minimization problem is defined by adding a second term of algebraic distance for the gradient contribution as

$$E = \sum_{i=1}^{n} \left\{ \left(F(\mathbf{p}_i, \boldsymbol{\theta}_c) \right)^2 + \left(G(\mathbf{p}_i, \mathbf{g}_i, \boldsymbol{\theta}_c) \right)^2 \right\} = \|\mathbf{M}\, \boldsymbol{\theta}_c\|^2, \tag{2.11}$$

subject to $h(\boldsymbol{\theta}_c) = 0$, where

$$\mathbf{M} = \begin{bmatrix} \{\mathbf{D}\}_{n\times 6} \\ \{\mathbf{Q}\}_{n\times 6} \end{bmatrix}, \tag{2.12}$$

is the new design matrix with $2n \times 6$ dimensions. The constraint proposed by Fitzgibbon et al. [39], $h(\boldsymbol{\theta}_c) = \boldsymbol{\theta}_c^{\mathrm{T}} \mathbf{N} \boldsymbol{\theta}_c - 1$, can now be applied to find a closed-form solution using the method of Lagrange multipliers.

2.3.4 Centroiding

The gray-weighted centroid method is widely used in photogrammetry applications and commonly serves as the benchmark for other algorithms. In this method, the pixel intensity, $I(x_i, y_j)$, is assigned to each pixel as the weight, and the target center coordinates, x_c and y_c, are found as:

$$\begin{pmatrix} x_c \\ y_c \end{pmatrix} = \frac{\sum_i \sum_j I(x_i, y_j) \begin{pmatrix} x_i \\ y_j \end{pmatrix}}{\sum_i \sum_j I(x_i, y_j)} \tag{2.13}$$

Now that we have reviewed the theoretical concepts and methodologies for circular target localization, in the next sections we compare the performance of these methods through simulations and experiments.

2.4 Accuracy and precision analysis

2.4.1 Physical targets

To study the performance of different target localization methods in terms of accuracy and precision, active and passive targets are considered. Physical targets are used as samples for creating a dataset of realistic and synthetically generated images of targets, each with a known center position, which serves as the ground truth. The active target is manufactured using a cylindrical bushing with an inner diameter of 12mm, a light diffuser layer, and an 850nm-120° IR LED, see Figure 2.3 (a). The passive target is a 12mm adhesive target with high reflectivity and black contour manufactured by AMETEK Inc., see Figure 2.3 (b). An example of the final test targets is shown in Figure 2.3 (c). Due to the difference between light intensities of each target, the diameters of the targets appear differently in the images. To perform a fair comparison, the images of each target are captured with their own specific exposure time, in this case 7ms for active and 27ms for passive targets, which results in equal diameters from each target in the images.

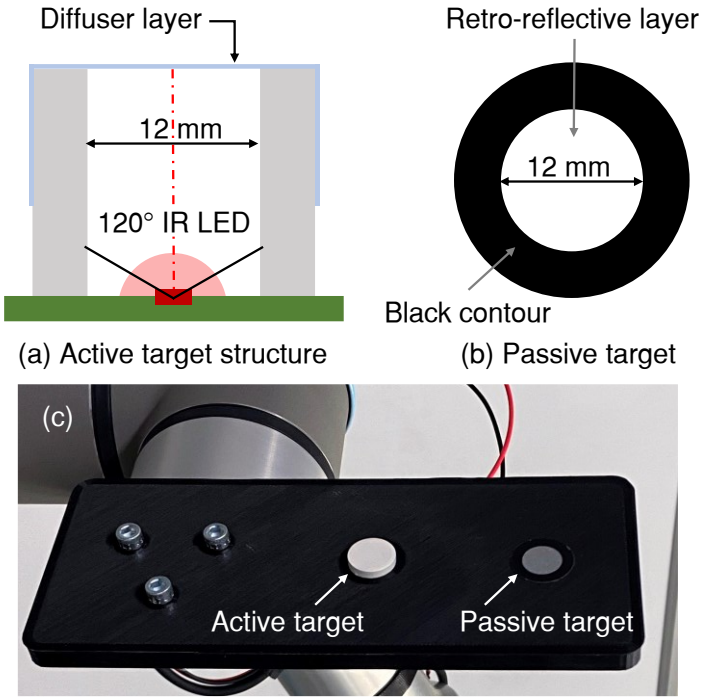

Figure 2.3: Schematics and dimensions of (a) active and (b) passive targets along with (c) final test targets.

2.4.2 Synthetic dataset

Here, we compare the accuracy and precision of different estimators using a dataset of synthetically generated images from circular targets with known center coordinates, which serves as the ground truth. To ensure high fidelity, the images of real targets in both active and passive cases were captured and used as benchmarks for generating the synthetic dataset. The synthetic targets were created by applying Gaussian blur and noise to an ellipse, which helped in creating a realistic transition profile between the target and the background. Figure 2.4 (a) shows the images of real active and passive targets along with the

synthetically generated target. These targets are also compared in terms of cross-sectional pixel values in Figure 2.4 (b).

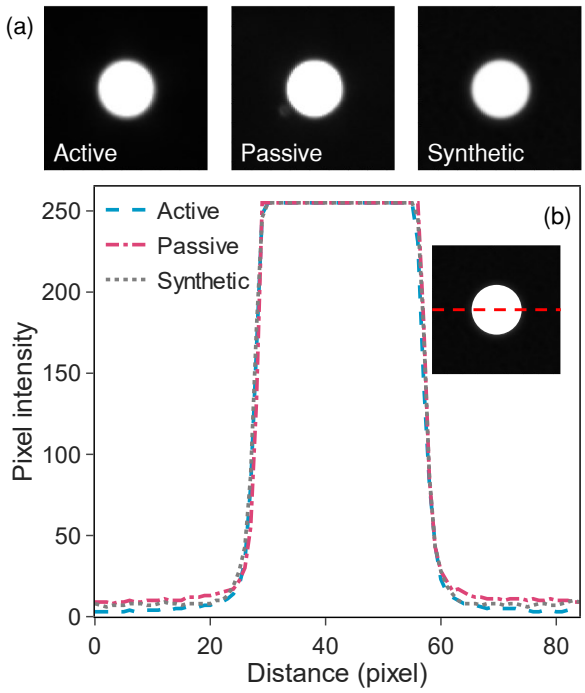

Figure 2.4: (a) Active, passive, and synthetic ellipses and (b) their pixel intensity profiles.

Here, we focus only on estimating the center coordinates since targets' center is the only parameter measured from images of a pose measurement system. Due to different environmental and imaging factors, such as dust, lighting and sensor characteristics, different levels of noise can appear in images. Furthermore, in practical scenarios, the camera-lens pair is used within a range of focus depths, leading to varying levels of blurriness at the edges of target images.

To investigate the effects of noise and blur on center coordinate estimation, we generated a dataset of 1,000 ellipses with varying levels of noise and blur. Each image in the dataset contains a single ellipse with randomly selected center

coordinates, axes, and angle. The axes and angle of the ellipse are randomly chosen from the ranges of [10, 20] pixels and [0, 179] degrees, respectively. The center coordinates of the ellipses are randomly selected within [-5, 5] pixels in the x and y directions from the center of the image. The images have dimensions of 85×85 pixels.

To create different levels of noise, synthetic images are contaminated by adding Gaussian noise with standard deviation of $\sigma = 0$ to 3 percent of the image intensity range (max(I) – min(I)) with the steps of 0.5 percent. The same dataset is generated with different levels of blur. Blur is generated in the levels 0 to 5 by applying a Gaussian filter with $\sigma = 0, 0.1, 0.5, 1, 1.5, 2$ and kernel size of $K = 0, 3, 7, 11, 15, 19$, respectively. Since the original images used for generating the synthetic dataset were captured in the best focus condition of the camera, the realistic dataset is considered as the 0-level blur and the higher levels of blur are added to it. After applying the blur filter, a Gaussian noise with $\sigma = 1\%$ is added to count for the noise introduced by the sensor.

2.4.3 Input data

In the methods that use the gradient information (line ellipse, and point-line ellipse), the input data points are extracted by calculating the image gradient. Since IR images are bimodal images of targets in almost constant lighting conditions, a gradient threshold can be set for selecting the pixel regions. These pixels include regions with high gradient values indicating the transition from the

target edge to the dark background. Although a higher number of data inputs (pixels) increases the amount of gradient information used in the solution, care must be taken to avoid introducing noisy data points that sacrifice the accuracy of the results. To determine the gradient threshold, T, a multiple M of the mean of gradient magnitudes is used as

$$T = M \times mean_{\mathrm{G}}, \tag{2.14}$$

where

$$mean_{\mathrm{G}} = \frac{\sum_{i=1}^{n}\sum_{j=1}^{m}\sqrt{\frac{\partial I}{\partial x}\left(x_i, y_j\right)^2 + \frac{\partial I}{\partial y}\left(x_i, y_j\right)^2}}{n \times m} \tag{2.15}$$

is the calculated mean value for gradient magnitudes in an $n \times m$ image. Based on our empirical tests, a threshold with $M = 3$ has been selected which results in a high overall accuracy by ensuring that only transition pixels are selected as inputs. In Figure 2.5 (a), a cross-sectional plot of normalized gradient magnitudes is shown for active, passive and synthetic targets. The highlighted pixel regions with high gradient magnitudes show the transition pixels on the edge of the targets (with $M = 3$). Figure 2.5 (b) shows the regions selected after removing the pixels below the threshold.

In point ellipse and corrected point ellipse methods, the edge pixels found by Canny edge detection are used as the input pixel data. Also, for the centroiding method, a binary threshold on the pixel values has been used to select the pixels on the edge and the area inside the target as the input data. Based on our empirical tests, a threshold of (pixel values higher than) 100 resulted in the

highest accuracy, and therefore has been used as the threshold for centroiding method.

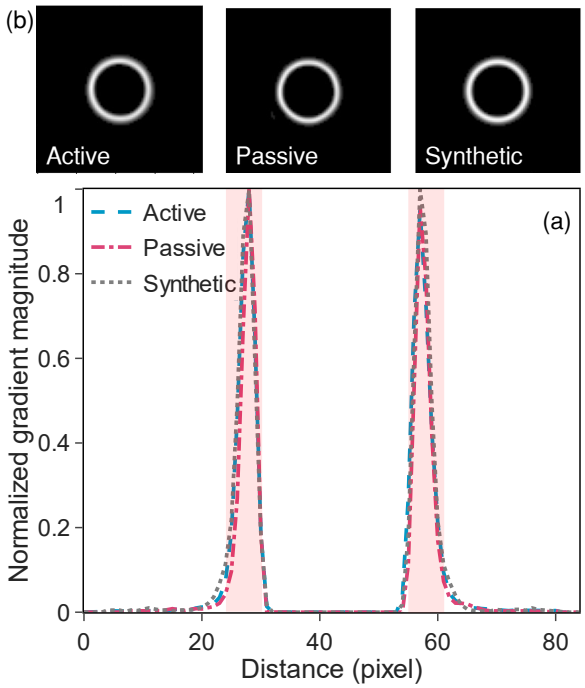

Figure 2.5: Transition edge between the target and the background in gradient magnitude of images from (a) active, passive, and synthetic targets along with (b) their normalized cross-sectional profiles and selected regions under red highlight.

2.4.4 Simulation results

Figure 2.6 and Figure 2.7 show the center position error for different methods subject to different levels of noise and blur, respectively. The mean and standard deviation (STD) of errors in center positions are calculated as a metric for accuracy and precision of estimations, respectively. As can be seen, the point-line ellipse and line ellipse methods show the best results in terms of accuracy and precision. The point-line ellipse method shows marginal improvement over line ellipse in higher levels of noise. However, in different levels of blur, the results seem

identical, and addition of point data does not improve the accuracy (see Figure 2.7). In the case of corrected point ellipse, although accuracy improvement is observed compared to the point ellipse method, both methods show lower overall accuracy compared to the centroiding method along with the fast degradation of accuracy by increasing the level of noise and blur. It should be mentioned that we also investigated the preprocessing and correction of edge points in point-line ellipse approach, and it does not improve the results in center localization.

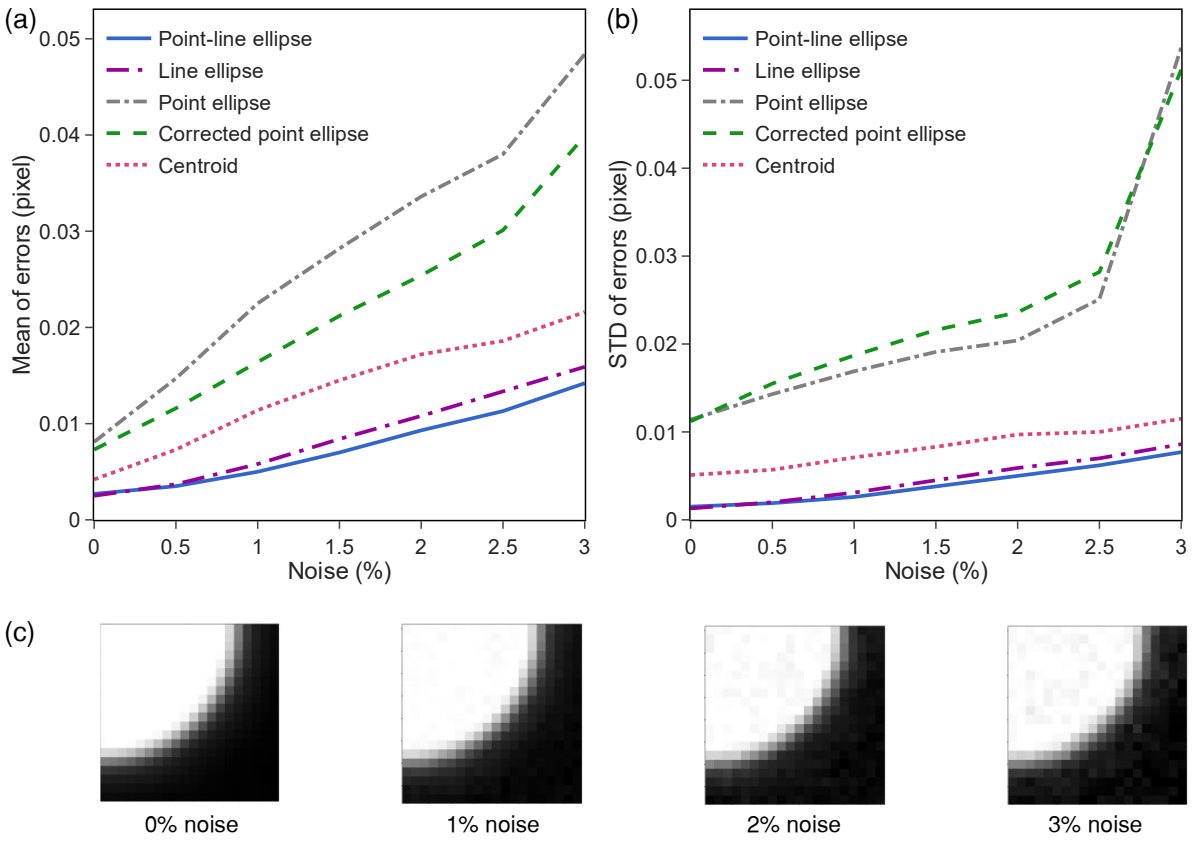

Figure 2.6: (a) Mean and (b) standard deviation of center position errors of different methods in different levels of noise.

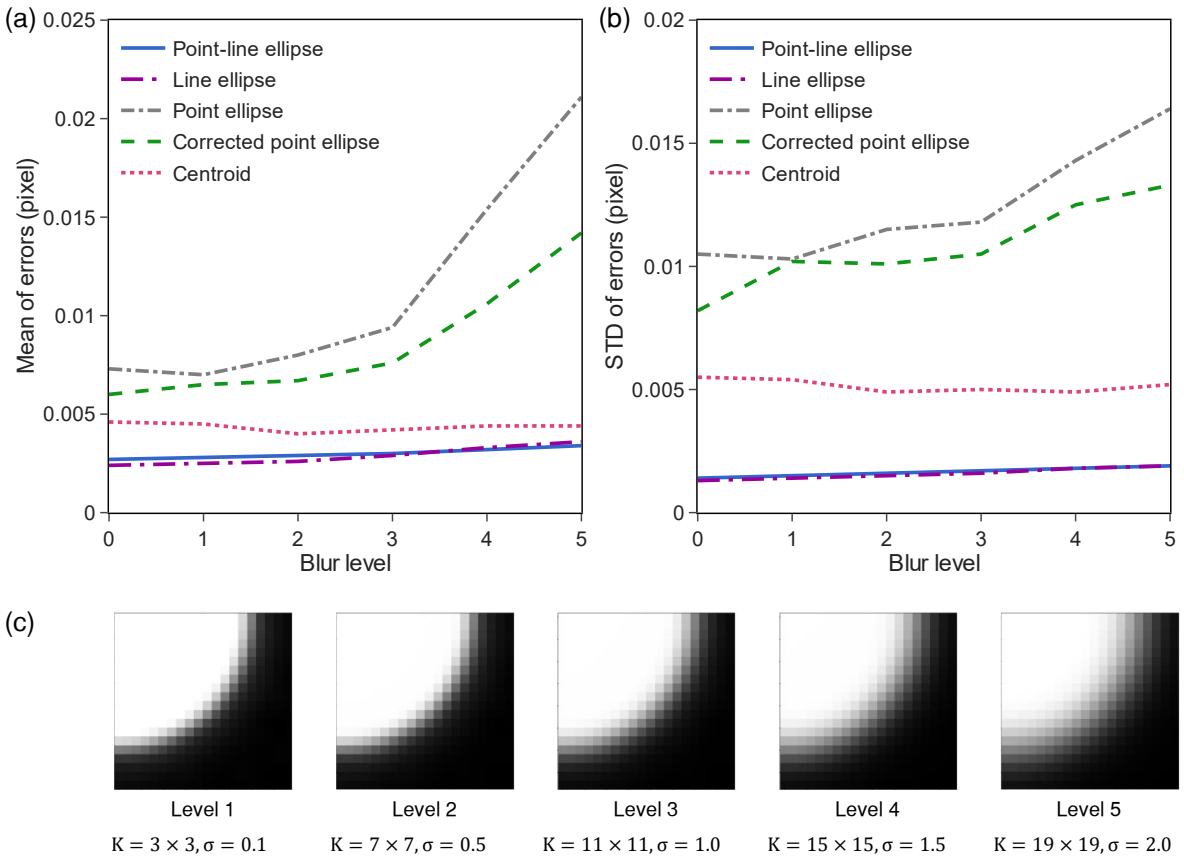

Figure 2.7: (a) Mean and (b) standard deviation of center position errors of different methods in different levels of blur.

2.5 Experimental evaluation in robotic applications

2.5.1 Experimental setup

An IR stereo vision setup has been developed as the robot motion tracking system to conduct experiments and compare different target localization methods in robot measurement scenarios. This system includes two 3.2 megapixel monochrome cameras paired with 12 mm lenses and 715 nm (near IR) long-pass filters. The cameras are synchronized using IEEE-1588 Precision Time Protocol (PTP) and

capture images in under 10 micro-seconds from each other. Each camera is also equipped with 850 nm IR projector rings. A Leica AT960 short-range laser tracker has been used as the reference measurement system for calibration of the stereo vision setup and evaluation of measurements (see Figure 2.8 (a)). The laser tracker has a positional uncertainty of ±30 µm in the operation range of ~2.5 meters used in this study. In all measurements, a UR5e collaborative industrial robot from Universal Robots has been used along with necessary attachments designed for holding the measurement targets (see Figure 2.8 (b)).

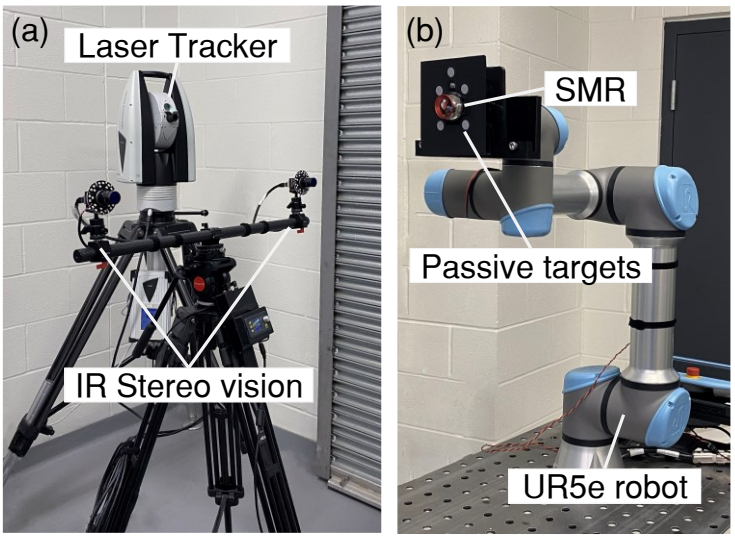

Figure 2.8: (a) IR stereo vision system, Leica laser tracker and (b) the Universal Robots UR5e robot with the measurement end-effector.

2.5.2 Camera calibration

To triangulate the targets with high accuracy, it is required to perform camera calibration and find the camera and lens parameters with high accuracy. A three-dimensional measurement volume of 400mm×400mm×400mm in robot workspace

has been selected for calibration of cameras and performing the measurement tests (see Figure 2.9). A calibration end-effector with nine passive targets is designed in a way that holds a laser tracker Spherically Mounted Retroreflector (SMR) at the barycenter of the targets (see Figure 2.9). The end-effector is then moved to a grid of 5×5×5 equally distanced points within the measurement volume using the robot. The position of the SMR is measured at each point. Each camera captures an image from each end-effector pose and the corresponding image point is found as the barycenter of the nine passive targets. Since the calibration end-effector is made by manually positioning the passive targets, the barycenter of targets cannot accurately align with the SMR center. However, this error only results in a translated camera coordinate system and does not affect the accuracy of calibration if the end-effector is moved between the calibration points with purely translational motion and with negligible rotations. The calibration is then performed using Bouguet's calibration method implemented in the OpenCV library. To increase the accuracy of target center localization in the presence of lens distortion, the Bouguet's method has been implemented in an iterative calibration procedure introduced in [44]. In this process, the camera parameters are recomputed and refined iteratively through successive undistortion and unprojection of images from the calibration artifact, which results in higher accuracy in center localization of each individual target. As for the target detection method, the point-line ellipse method is used for finding the center of each target, and the barycenter of all nine targets is then calculated and used as the image

point in the calibration process. For the camera model, a pinhole model is considered with 3 and 2 parameters for radial and tangential lens distortions, respectively.

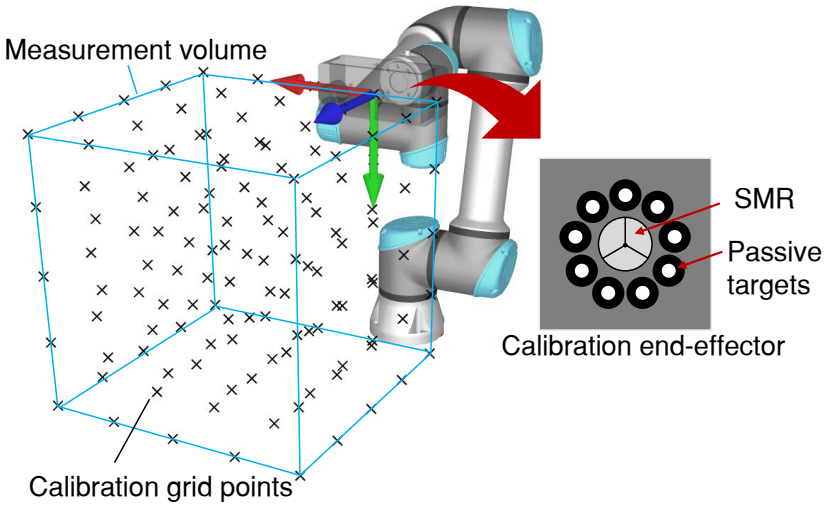

Figure 2.9: Measurement volume in robot workspace, calibration grid points, and calibration end-effector.

2.5.3 3D accuracy

The calibrated stereo vision system is used to estimate positions of the robot end-effector in the measurement volume. The laser tracker is also used to measure the end-effector positions at the same time and evaluate accuracy. This evaluation is performed using the distances between two successive positions. Using the distance as the accuracy metric eliminates the need for transformation between the laser tracker and vision system which itself can introduce additional errors to the estimations. For this test, the robot is moved to 100 pair of points that are 100 mm apart from each other and positioned randomly inside the 3D measurement

volume. The robot end-effector includes an SMR target and up to 4 passive targets around the SMR. Except for the case that includes only one passive target, the passive targets are mounted in a way that their barycenter coincides with the SMR center, see Figure 2.10 (c). At each point, the vision system was used to measure the 3D positions of the targets, and their barycenter was calculated when more than one target was used. Then, the Euclidean distance between the subsequent positions was calculated and compared with the distance measured by the laser tracker. This experiment has been performed for different number of targets from 1 to 4 and using different target localization methods. The result of experiments is shown in Figure 2.10 (a) and (b) in terms of mean and standard deviation of errors, respectively.

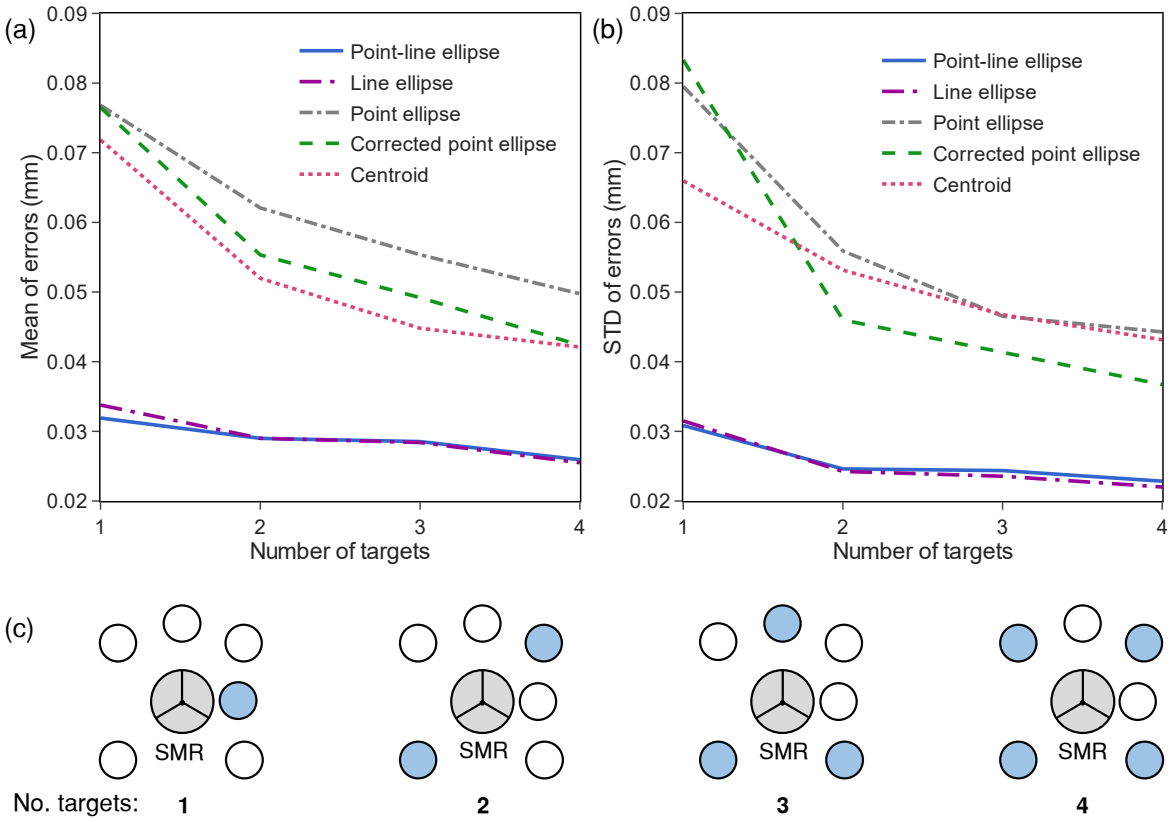

Figure 2.10: A comparison between (a) mean and (b) standard deviation of 3D distance errors from different target localization methods with different number of passive targets.

It can be seen that the 3D distance errors in line ellipse and point-line ellipse methods show higher accuracy and precision compared to other methods. Their mean error for a single target was found to be approximately 0.033 mm. As the number of targets increases, the error reduces gradually, reaching 0.025 mm for 4 targets. Other methods show a trend in accuracy which is somewhat close to each other, starting from around 0.075 mm for a single target to around 0.050 mm for 4 targets. This indicates that methods that use gradient information can achieve significantly higher distance accuracy with even a single target, compared to other

methods such as centroiding. Furthermore, it can be inferred that in practical robotic applications, the accuracy of these methods is bound to the accuracy of the calibration process and reference measurement device, i.e., the laser tracker, with the accuracy of 0.030 mm.

2.5.4 6D accuracy

The 6D pose of the robot is measured using the stereo vision setup and the laser tracker's 6D probe, Leica T-Mac. The T-Mac probe is equipped with an SMR and 10 additional IR LEDs, which enable precise and absolute 6D pose measurements. The measurements are compared to that of the laser tracker in terms of 3 positional and 3 orientational components. The T-Mac probe is rigidly attached to the robot flange, along with circular passive targets, as seen in Figure 2.11 (a). The passive targets are affixed to the flat panels located on the top and bottom of the T-Mac probe in a circular pattern with an average diameter of 160 mm. Different numbers of passive targets are used for pose estimation (4, 8, 18, and 30 targets) as shown in Figure 2.11 (b). The robot has been moved to 100 randomly selected locations in the measurement volume, each separated by a distance of 100 mm in 3D space. To establish the three orientations of the robot at each point, the end-effector roll, pitch, and yaw rotations are randomly selected from the range of [-15, 15] degrees. This allows for the experiment to be performed in a range of different orientations and positions, ensuring the accuracy of the pose measurement is thoroughly investigated inside the measurement volume.

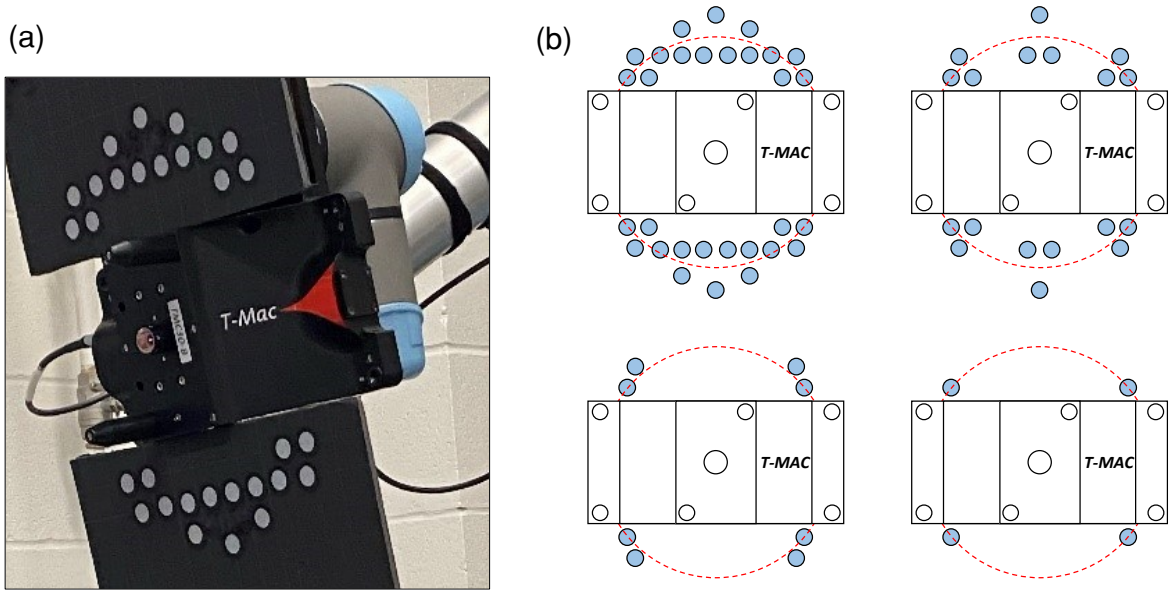

Figure 2.11: (a) T-Mac and passive targets attached to the robot flange in (b) different target configurations.

In this experiment, the measurements obtained from the laser tracker are the absolute pose of the T-Mac probe with respect to the laser tracker coordinate system. On the other hand, the measurements from the stereo vision are relative poses found by calculating rigid-body transformations using singular-value decomposition (SVD) between two subsequent positioning of targets with respect to the stereo vision coordinate system. To generate absolute measurements with the stereo vision system, an initial pose of robot at the center of measurement volume is set as the initial reference, where a frame parallel to the stereo vision coordinates is assigned to. This initial frame is shown as absolute transformation $\mathbf{A_0}$ in Figure 2.12. In Figure 2.12, the same pose is measured from the T-Mac probe using the laser tracker with absolute transformation $\mathbf{B_0}$. Since the stereo vision targets and T-Mac probe are rigidly attached to the robot flange, the rigid-body

transformation **X** between them can be found by solving the "**AX** = **XB**" calibration problem. Here, the Lie Group method proposed by Park and Martin [45] is used to find **X**, which is the transformation between the frame attached to the T-Mac probe and the frame assigned to the passive targets. In the calibration process, 40 pairs of measured poses ($\mathbf{A_{0i}}, \mathbf{B_{0i}}$) are used. To perform the comparison between the stereo vision and the laser tracker, the measured 6D poses from the stereo vision system, $\mathbf{A_{0i}}$, are compared with $\mathbf{XB_{0i}X^{-1}}$, which is the measured pose from the laser tracker expressed in stereo vision coordinate system. For each measured pose, the deviations of position and orientation are calculated from laser tracker as errors. The root-mean-square (RMS) values obtained from different methods of target localization in different number of targets are presented in Figure 2.13.

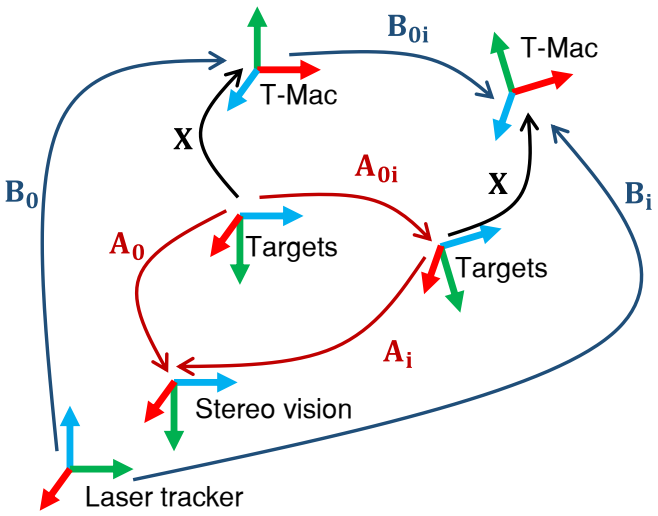

Figure 2.12: Calibration process between retro-reflective targets and T-Mac probe.

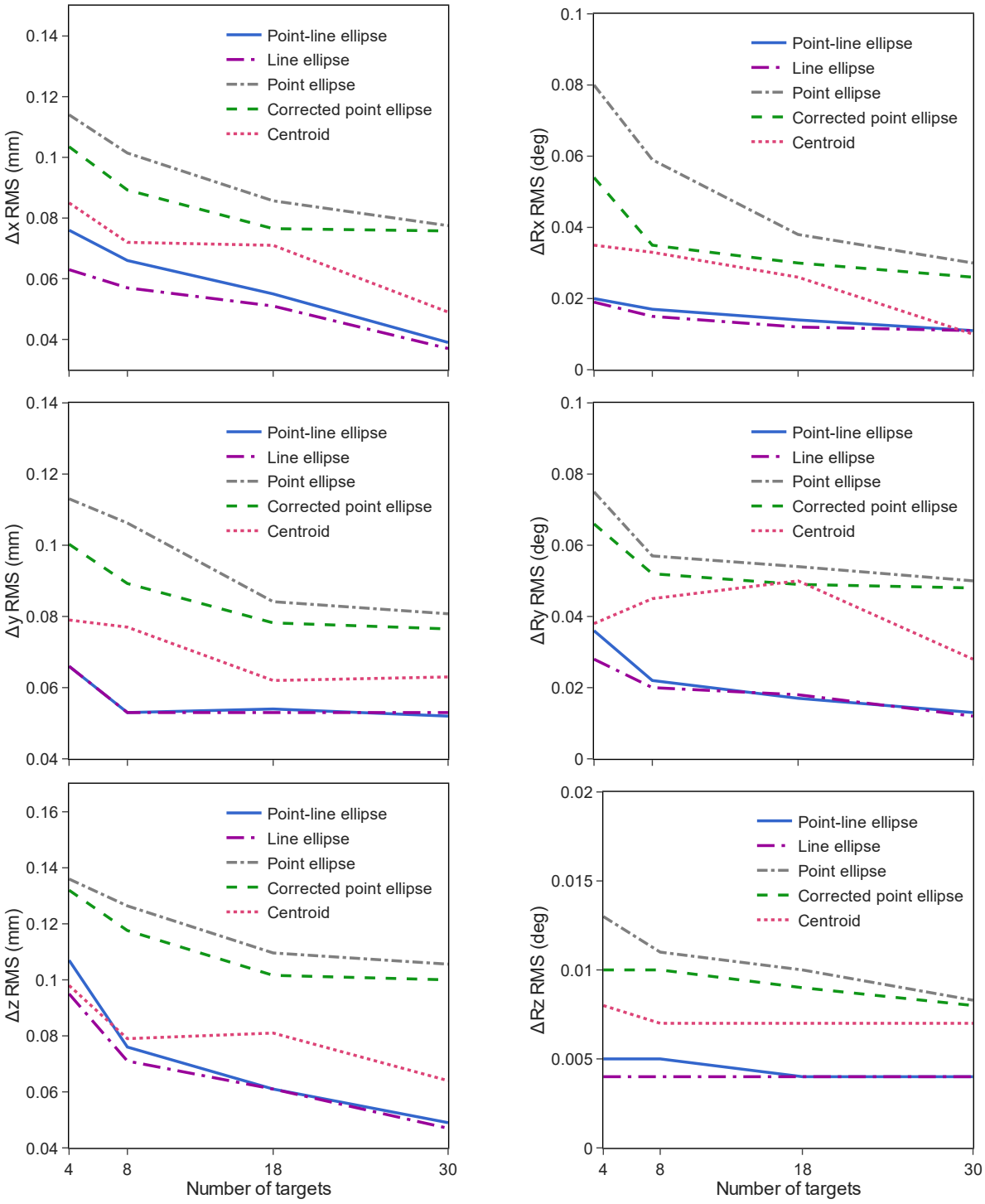

Figure 2.13: RMS errors of robot 6D pose estimation of different target detection methods with different number of targets.

From Figure 2.13, it is evident that the line ellipse and point-line ellipse methods exhibit the lowest level of errors in both positional and orientational elements. However, all the methods demonstrated a higher baseline error compared to the distance errors obtained in the previous section. This can be attributed to the inclusion of a calibration step between the laser tracker and vision system in the present analysis. Nevertheless, calibration errors are unavoidable in practical robotic applications, where multiple levels of calibration are necessary, such as calibration between the vision system and the robot base frame, and between the robot flange and tool center point (TCP). While the analysis of distance errors demonstrated a significant advantage of gradient-based methods over the centroiding method (as shown in Figure 2.10), from Figure 2.13, it can be inferred that the advantage is reduced in a more practical setup that includes calibration steps between different coordinate frames. Therefore, in a practical scenario, the disadvantage of the centroiding method in terms of accuracy can be neglected, as it offers a higher computational efficiency.

Table 1 presents the numerical values depicted in Figure 2.13, obtained with 30 targets, along with the robot errors measured using the laser tracker. To enable a comparison in terms of computation time, the average solution time for the synthetic ellipse dataset of 1000 is also included. The solution times were obtained on an AMD 5900X CPU, with the methods programmed in Python 3.7 using Numpy library version 1.21.6.

The results presented in Table 1 demonstrate that all of the tested methods can achieve a measurement accuracy that exceeds that of the robot itself. This improved accuracy has potential applications in fields such as robot calibration and error correction, robot guidance, and assembling sensory data. Furthermore, it is shown that the centroiding method, which is computationally more efficient than other methods (around 10 times), exhibits only a slightly higher error compared to the methods with the best performance, i.e., line ellipse and point-line ellipse methods. This difference in error may be acceptable in many practical applications when higher measurement frequency is required, or lower computational resources are available. Overall, the results presented in Table 2.1 provide insights into the trade-offs between accuracy and computation time for different target localization methods and can help guide the selection of an appropriate method for a given robotic application.

Table 2.1: A comparison of the computation time and error RMS obtained from the robot and different target localization methods, as measured by the laser tracker.

Method	Time (ms)	Δx (mm)	Δy (mm)	Δz (mm)	ΔR_x (deg)	ΔR_y (deg)	ΔR_z (deg)
Robot (UR5e)	—	0.539	0.774	0.388	0.126	0.036	0.046
Line ellipse	0.46	0.037	0.053	0.047	0.011	0.012	0.004
Point-line ellipse	0.46	0.039	0.052	0.049	0.011	0.013	0.004
Centroiding	0.04	0.049	0.063	0.064	0.010	0.028	0.007
Point ellipse	0.41	0.077	0.081	0.106	0.030	0.050	0.008
Corrected point ellipse	2.47	0.076	0.077	0.100	0.026	0.048	0.008

2.6 Summary

In this chapter, we presented a comparative study of target localization methods for high-precision vision-based measurement of robot pose. Through simulations and experiments conducted with a laser tracker, we evaluated the accuracy, precision, and computation time of five different methods, including centroiding, ellipse fitting with point data and gradient information, and ellipse fitting methods with augmented and corrected input data. Our analysis includes the evaluation of both 3D distance and 6D pose. Our results show that gradient-based methods exhibit the lowest level of errors in both position and orientation estimations. However, all the tested methods can achieve a measurement accuracy that exceeds that of the robot itself. The centroiding method, which is computationally more efficient than other methods, exhibits only a slightly higher error compared to the methods with the best performance. This trade-off between accuracy and computation time may be acceptable for many practical robotic applications where higher measurement frequency or lower computational power is required. Furthermore, this study highlights the importance of calibration in practical robotic applications and how it can affect the accuracy of pose measurement methods. The inclusion of calibration steps between different coordinate frames can reduce the advantage of gradient-based target localization methods over the centroiding method in terms of accuracy. It is worth noting that the calibration errors between the vision system and the robot base frame are inevitable and can affect the measurement accuracy. Therefore, when the highest levels of accuracy are required, proper calibration procedures must be performed

for different stages of calibration. In conclusion, the results presented in this study and the insights gained from our experiments can help guide the selection of an appropriate method for a given robotic application.

Chapter 3

Online Relative Pose Correction

This chapter aims to address the implementation of an online relative pose correction technique to improve the accuracy of industrial robots for drilling operations. A crucial requirement for successful pose correction is obtaining real-time and precise measurements of the end-effector's pose, i.e., the drilling spindle. In this context, this chapter focuses on pose estimation and correction by leveraging the capabilities of the developed optical measurement system presented in Chapter 2.

3.1 Background

Achieving high-precision drilling tasks using industrial robots requires careful consideration of both geometric and non-geometric sources of error. One approach to tackle this challenge involves the offline calibration of robots by re-evaluating

their kinematic model parameters and adjusting the nominal parameters within the robot controller [46]. However, accurate kinematic calibration is time-consuming and still may not provide the required accuracy at all robot poses [18]. Additionally, the calibration process often needs to be repeated depending on the payload of the robot, which further disrupts production operations. While specific tolerances may vary depending on the specific application and component being manufactured, aerospace machining operations generally aim for tight tolerances in order to ensure the integrity and reliability of assembled structures. In many cases, these operations strive to achieve tolerances within the range of 0.025-0.120 millimeters [47]. To meet these tolerances, robotic pose correction solutions have emerged to enhance the accuracy of drilling operations in aerospace manufacturing. This chapter discusses the development and implementation of a pose correction technique, aiming to significantly improve drilling accuracy and overall efficiency in the aerospace industry.

In the drilling process, the perpendicularity of drill bit relative to the surface plays a crucial role. Any deviation in the alignment of the tool axis and surface normal can lead to undesired outcomes, such as increased hole diameter and cutting forces. Consequently, this can negatively impact the quality of joints between aircraft components. To address this issue, various techniques have been developed to enhance the accuracy of normal adjustment in drilling processes.

Laser displacement sensors are extensively used to measure surface normal and ensure perpendicularity of drilling operations. These sensors are non-contact

measurement devices known for their numerous advantages, including high precision, low power consumption, and high reliability. Usually, three or four laser displacement sensors are installed on the robot's end-effector to perform surface normal measurements [48, 49].

Several studies [50–52] have proposed drilling end-effectors equipped with four laser displacement sensors to measure the surface normal during the drilling process. However, the potential assembly errors of these sensors were not taken into account in their approach. The presence of machining errors in the end-effector and assembly errors results in small discrepancies between the nominal and actual zero points, as well as the laser beam directions of laser displacement sensors. Therefore, researchers [53, 54], have proposed a calibration method to compute the errors of the zero point and laser beam direction of laser displacement sensors. They reported a reduction in average angular deviation down to 0.1°, which satisfied the established tolerances. While these methods effectively correct the robot's orientation with respect to the surface, they do not directly address the challenge of accurate robot positioning.

Laser trackers are an alternative solution that can be used as an external measurement system for enhancing the accuracy of drilling operations. Laser trackers, in essence, perform as a laser displacement sensor that follows a target attached to the robot, and can offer a maximum 3D positional accuracy of 0.015 mm [55]. They also offer additional 6D probes, e.g., Leica T-Mac, with orientational accuracy of 0.03° [56].

Liu et al. [57] used a laser tracker with several 3D targets on the end-effector to correct the robot's errors in a peg-in-hole assembly robot. In a similar approach, Posada et al. [58] proposed an error correction solution for robotic drilling using a laser tracker and three separate 3D targets attached to the drilling spindle. Although they reported positional errors smaller than 0.1 mm and rotational deviations of 0.2°, their solution cannot be used for real-time measurements as the targets must be measured once at a time using the laser tracker. Droll [59] developed a real-time path correction with direct end-effector feedback from a Leica laser tracker paired with a 6D probe. He reported an RMS error of 0.11 mm while driving the robot with 100 mm/s. Moeller et al. [60] used two laser trackers with 3D position targets and a 6D probe attached to the robot end-effector. They used real-time measurements from the laser tracker in a secondary controller to correct the robot's trajectory and achieved errors less than 0.25 mm in a milling operation. Chen et al. [61] combined a laser tracker positioning target with four laser displacement sensors to improve the positioning accuracy in robotic drilling based on co-kriging. Their method utilizes error similarity based on the kinematics of the drilling robot and laser tracker measurements to estimate positional errors at various points in the workspace. They reported a maximum/average error of 0.26/0.12 mm in the positional error. Fernandez et al. [62] proposed the implementation of a laser tracker with 6D probe along with a laser line scanner at the robot end-effector. With their suggested system, they achieved a positioning accuracy of 0.15 mm in the robot workspace. Wang et al. [63] proposed a solution

for real-time path error compensation in robots using a single 3D laser tracker target. They tested their methodology for robotic drilling tasks and showed a reduction in hole positional errors from 0.83 mm to 0.17 mm. Cvitanic et al. [64] investigated the effect of fusing the data from a laser tracker 6D probe with an inertial measurement unit (IMU) to improve the robot end-effector state estimation. They reported up to 95% improvement in velocity estimation accuracy, and up to 45% improvement in angular acceleration estimation from the laser tracker measured data. As previous studies suggest, laser trackers are undeniably suitable for precise position and orientation measurements in industrial robots. However, their high cost and challenges in implementation, particularly due to the bulkiness of 6D probes and limitations of 3D targets, position them as better suited for offline error compensation and robot calibration rather than online error correction.

Optical coordinate measurement machines (CMMs) are another alternative solution that can be used as an external measurement system for robotic applications. Schneider et al. [65] proposed an approach for position control of industrial robots using the pass-through between an industrial Computer Numerical Control (CNC) and servomotors. They used a CNC-controlled robot with an external optical CMM system paired with LED targets to create a closed-loop feedback control on the robot end-effector and improve the robot accuracy to less than 0.4 mm. Chen et al. [66] proposed an optical/inertial data fusion system for motion tracking of the robot manipulator. They used Kalman filter to fuse the

data from an IMU with measurements from optical trackers to achieve submillimeter accuracy. Gharaaty et al. [67] used an optical CMM system for pose correction of the robot end-effector. They used a root mean squared filtering approach coupled with a proportional-integral-derivative (PID) controller to correct robot's pose error in an online manner and reported an accuracy of 0.05 mm and 0.05° for position and orientation, respectively. Compared to laser trackers, optical CMMs are easier to implement for robot error correction but they offer lower accuracy.

Vision-based systems offer a practical and accurate measurement solution owing to their tailored approach and reasonable cost [9]. Several vision-based solutions have been proposed to enhance drilling positional accuracy. Zhu et al. [68] presented a 2-dimensional (2D) vision system designed to enhance the positional accuracy of drilling processes. They achieved this by localizing multiple reference holes in the part, which ultimately led to a positional accuracy of 0.1 mm. Frommknecht et al. [69] proposed a combined system of 2D vision detection along with laser distance sensors to detect and localize reference holes, and achieved an accuracy of 0.3 mm. In a similar approach, Mei and Zhu [70], combined laser displacement sensors with a 2D vision camera to position the tool using a set of reference holes. They reported a positioning accuracy of 0.05 mm. Similar approaches [71–73] have been used for positioning the drilling tool using different detection and localization methods for reference holes. Several studies [74–76] concentrated solely on achieving robust detection of circular holes through

contour refinement and model fitting, as it directly affects the precision of the drilling process.

Vision-based solutions based on part localization and feature detection have also been utilized in other manufacturing tasks [77]. Yang et al. [78] proposed a robotic multi-view pose estimation method for the pick-and-place operation of large parts with an accuracy less than 1 mm. Jiang et al. [79] developed a vision-based guidance solution for robotic peg-in-hole operations. They used two sets of binocular cameras, one with eye-to-hand configuration to localize the robot end-effector, and the other one installed on the robot (eye-in-hand configuration) for aligning the tool with reference holes. Ayyad et al. [80] developed a neuromorphic vision based solution for positioning of robotic drilling systems. In their solution, they first performed a multi-view reconstruction to estimate the workpiece's pose, and then, they refined the pose for a local region of workpiece by detecting reference circular holes. A positional accuracy of 0.2 mm is reported for their solution. Although above-mentioned vision-based solutions for robotic drilling can offer high accuracy in measurements, they mainly rely on pre-existing reference holes for correcting the robot pose. This constraint restricts their implementation in scenarios where drilling operations are required on large composite panels, where reference holes either are not visible or do not exist.

This study focuses on the development of a high-precision vision-based measurement system that can deliver accurate measurements in the robot workspace. By simultaneously measuring the robot pose and the part pose, we

combine part localization with robot pose error correction and propose an online relative error correction solution between the robot end-effector and the part. We compare different implementation methods for the proposed solution, and finally, we implement the developed solution on the test setup.

3.2 Pose measurement

As discussed in Chapter 2, each triangulated target is measured as a 3D point coordinate, (x, y, z), in the optical tracker's coordinate system, here called world frame, $\mathcal{F}_w$. Having the position of at least 3 non-colinear targets, a coordinate frame, $\mathcal{F}_s$, can be assigned to the drilling spindle. The pose of $\mathcal{F}_s$ with respect to the world frame $\mathcal{F}_w$ is measured as ${}^{w}\mathbf{T}_s$. Since the initial arrangement of the targets on drilling spindle are arbitrary, $\mathcal{F}_s$ is an arbitrary frame and the measurements are to be performed with respect to an initial pose of the drilling spindle. This type of measurement is called "relative" pose measurement, as the initial pose of the drilling spindle is assigned arbitrarily. Here, the initial frame of the spindle is assigned in a way that the center of spindle frame, $\mathcal{F}_s$, is positioned at the barycenter of the targets, in parallel with the coordinate frame of the optical tracker $\mathcal{F}_w$, see Figure 3.1. As long as the optical tracker observes the targets, it can measure the rigid body motion of the targets due to the motion of the drilling spindle.

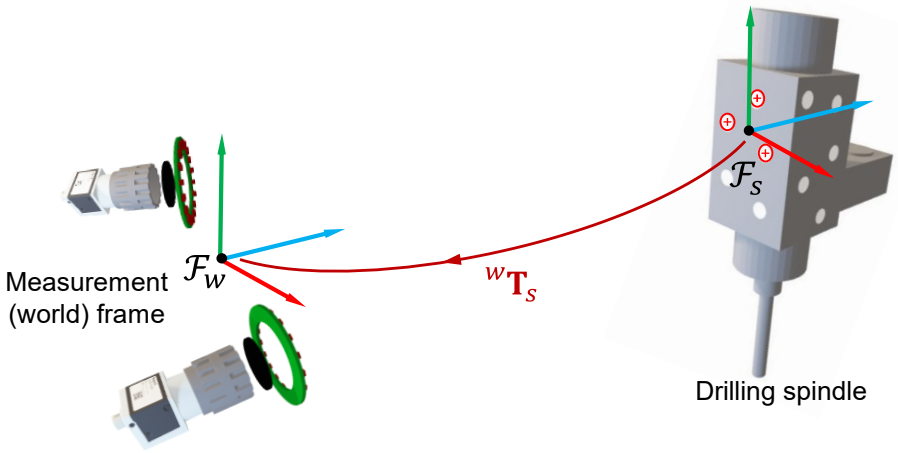

Figure 3.1: Measured pose of drilling spindle in the optical tracker (world) frame.

Here, the relative pose of two subsequent configurations of the targets in 3D space is found using the Least-Squares fitting method based on Singular Value Decomposition (SVD) proposed in [81]. It should be mentioned that since the initial drilling spindle frame is an arbitrary frame, it can be assigned to any point in space and in any orientation.

3.2.1 Software interface

To facilitate the online computation of poses, error calculation, controller implementation, and real-time error corrections, a software interface has been developed in Python (Figure 3.2). The software comprises two primary components. The first component, robot communication, enables sending corrective motion commands to the robot. The second component handles communication with the optical tracker and carries out the measurement tasks while allowing the user to set up the measurement parameters.

3.2.1.1 Robot communication

Communication with the robot controller is established using the RoboDK API via an Ethernet cable and TCP/IP protocol. Through this interface, users are able to access variables of the robot controller as well as the current pose of the robot relative to either the base frame or a user-defined frame. Additionally, through this interface, users can apply relative transformations to the robot end-effector.

3.2.1.2 Measurement system communication

The optical tracker interface enables users to perform several functions, including connecting to cameras, synchronizing the cameras, creating targets and frames (rigid bodies), saving and loading frames, calculating transformations between frames, and checking the status of camera sensors. The tracked targets and frames are visually presented through 2D camera views and 3D visualization within the RoboDK environment. Figure 3.2 provides a visual representation of the designed software interface.

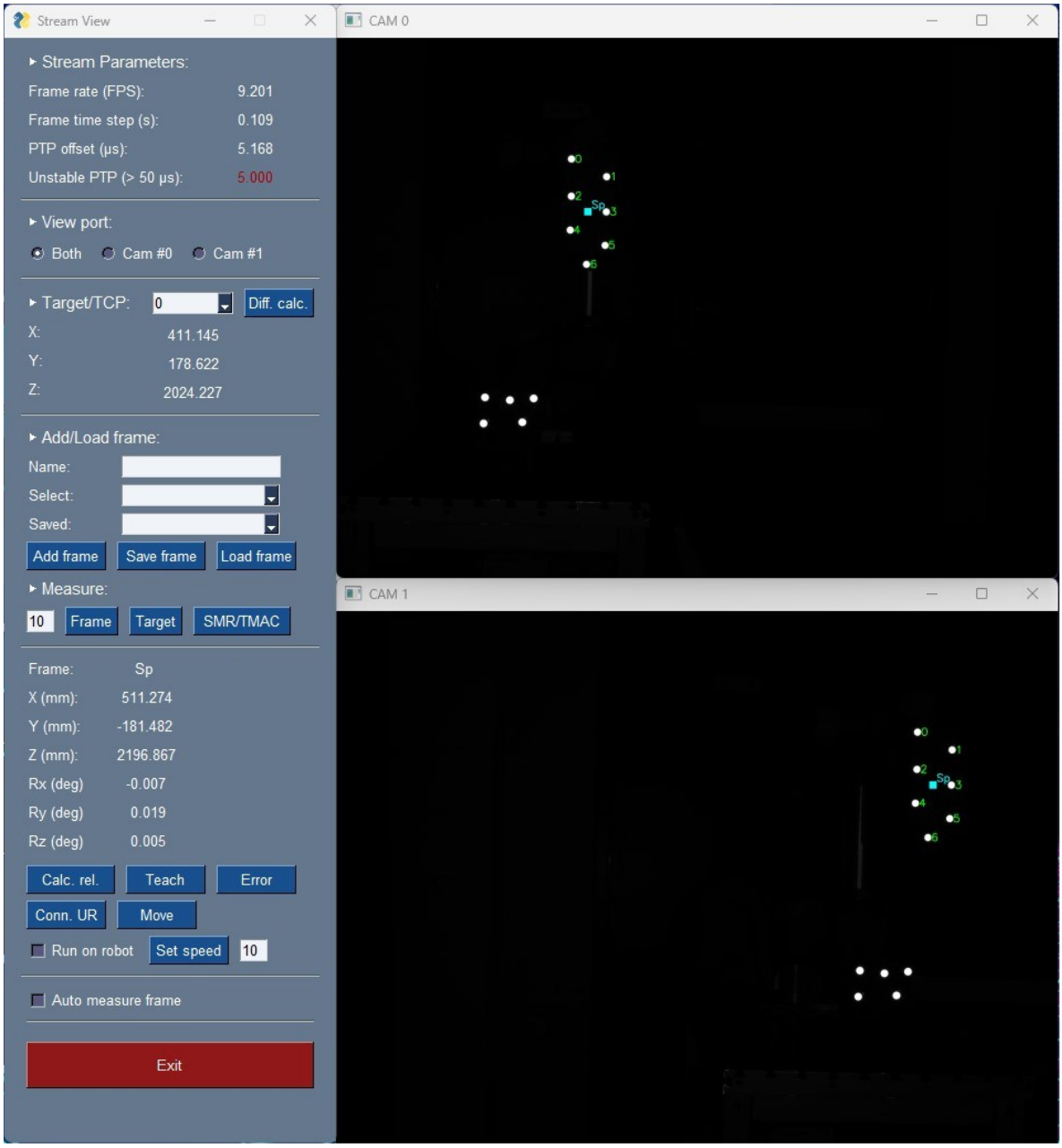

Figure 3.2: Developed software interface for vision measurement system and robot communication.

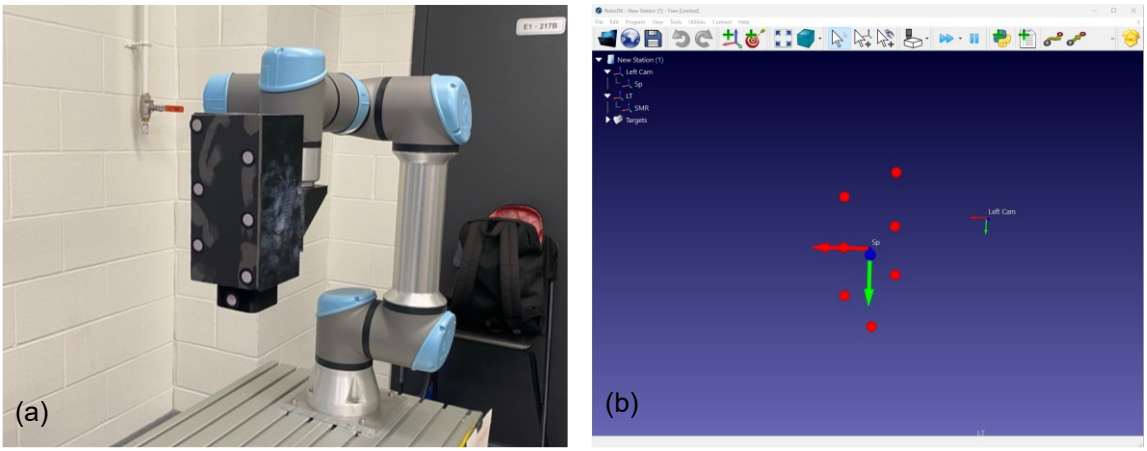

Figure 3.3: (a) Drilling spindle with targets and (b) assigned frame to the rigid body using the developed software and visualized in RoboDK.

To utilize the pose measurement data as feedback for robot error correction, it is necessary to combine and establish communication between the robot and measurement system interfaces. Targets and frames can be created or loaded into the software as needed. When the frames are loaded, the software continuously measures and displays their real-time position and orientation with the measurement frequency. Figure 3.4 depicts the connection between the measurement system, computer, i.e., software interface, and robot controller. As can be seen, the measurement system supplies the robot's pose, which is received by the computer. The software then generates the corrective motion commands to make corrections to the position and orientation of the drilling spindle using the robot controller.

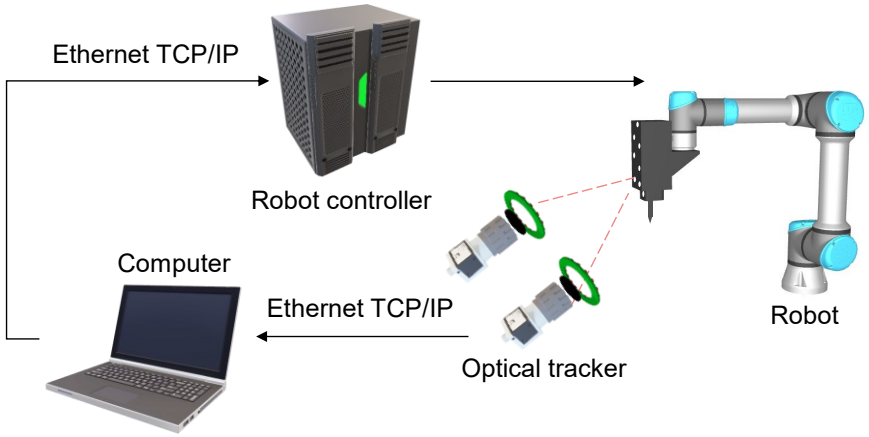

Figure 3.4: Schematic of connections between different elements.

3.3 Robot-vision calibration

Figure 3.5 shows the kinematic chain between the vision system and the robot. In this chain, the transformation between the robot base frame, $\mathcal{F}_b$, and the optical tracker is shown as ${}^{w}\mathbf{T}_{b}$. The transformation between the robot flange, $\mathcal{F}_f$, and the assigned frame to the spindle targets, $\mathcal{F}_s$, is presented as ${}^{s}\mathbf{T}_{f}$. Both ${}^{w}\mathbf{T}_{b}$ and ${}^{s}\mathbf{T}_{f}$ are unknown and required to be found so that the measurements can be used for correcting the robot's pose. This can be done by solving a calibration and pose estimation problem using the robot pose at different locations in the robot workspace and within the vision system's field of view.

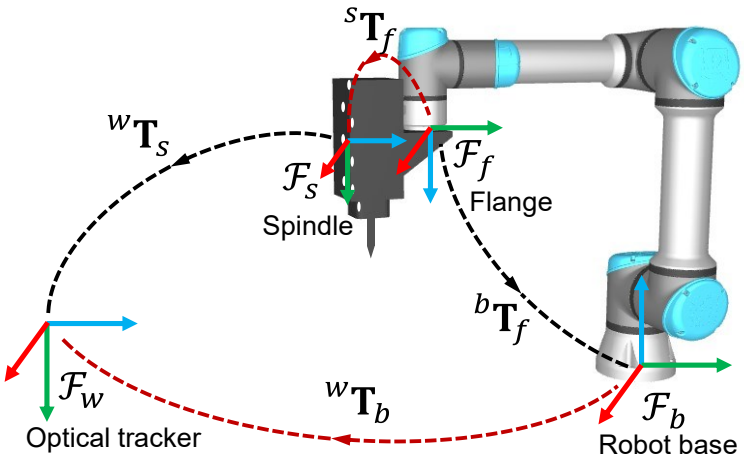

Figure 3.5: Kinematic chain between vision system and robot.

Here, there exist two sets of measurements from two separate systems: the vision system that measures the pose of frame attached to the spindle, and the robot controller that provides the flange pose. The spindle targets are rigidly attached to the moving spindle which is connected to the robot flange with a fixed unknown transformation, ${}^{s}\mathbf{T}_{f}$. The targets are tracked at poses j = 1, 2, ..., n by the vision system with the data being represented as ${}^{w}\mathbf{T}_{s}^{j}$, while simultaneously the robot controller is reporting the flange pose at j = 1, 2, ..., n with the data being represented as ${}^{b}\mathbf{T}_{f}^{j}$. Both ${}^{w}\mathbf{T}_{s}^{j}$ and ${}^{b}\mathbf{T}_{f}^{j}$ are measured with respect to their own reference coordinate frame, i.e., vision system and robot base, respectively. Therefore, if the unknown fixed transformations ${}^{w}\mathbf{T}_{b}$ and ${}^{s}\mathbf{T}_{f}$ are calculated, one can transform the measured pose from one coordinate frame, i.e., the vision system, to the coordinate frame of the other system, i.e., the robot. Thus, the data from the vision system can be directly compared with the robot's motion.

The unknowns ${}^{w}\mathbf{T}_{b}$ and ${}^{s}\mathbf{T}_{f}$ can be found by solving the calibration problem

$$ {}^{w}\mathbf{T}_{s}^{j}\ {}^{s}\mathbf{T}_{f} = {}^{w}\mathbf{T}_{b}\ {}^{b}\mathbf{T}_{f}^{j}, \tag{3.1} $$

at poses j = 1, 2, ..., n. This problem is commonly referred to as **AX** = **YB** in the context of calibration between two systems of measurement. If there is no sensor noise, theoretically, only 3 unique poses are required to find a solution. In practice, however, there is always measurement noise, making it necessary to use a higher number of measurements for more generalized solutions. The closed form solution based on Kronecker product and singular value decomposition proposed in [82] can be used to estimate both unknown transformations.

3.4 Relative pose correction

Here, the goal is to correct the robot pose errors using the developed measurement system. To do so, one can define a point (or path) in the measurement coordinate frame and guide the robot to the defined point. This requires finding the transformation between the robot and the measurement system, ${}^{w}\mathbf{T}_{b}$, and the transformation between the spindle targets and robot flanges, ${}^{s}\mathbf{T}_{f}$, to generate the robot commands that guides the robot spindle to the desired point. This solution, however, accumulates the errors from the calibration process of both robot and spindle and does not fully exploit the benefit of the developed measurement system.

To increase the accuracy of robot guidance, here, we rely on the ability of the vision system in measuring rigid body frames relative to each other in the same reference frame, i.e., the world frame. To do so, high-precision fixture targets are

assumed to guide the robot to the location of drilling in the robot workspace. These fixtures localize the part and act as a reference frame in which the drilling operation is performed. Figure 3.6 shows the schematics of the relative pose error correction between the drilling spindle moved by the robot with respect to the fixture frame, $\mathcal{F}_t$. In this case, ${}^{t}\mathbf{T}_{s}^{des.}$ is the desired relative pose between the spindle and the fixture, and the goal is to make the measured relative pose between the spindle and the fixture, ${}^{t}\mathbf{T}_{s}^{mea.}$, as close as possible to the desired one. Therefore, one can compute the error between the desired and measured poses as ${}^{des.}_{w}\mathbf{T}_{mea.}$ with respect to the measurement frame. This relative error can be corrected by sending a relative motion command to the robot flange which is represented as ${}^{des.}_{f}\mathbf{T}_{mea.}$ in Figure 3.6.

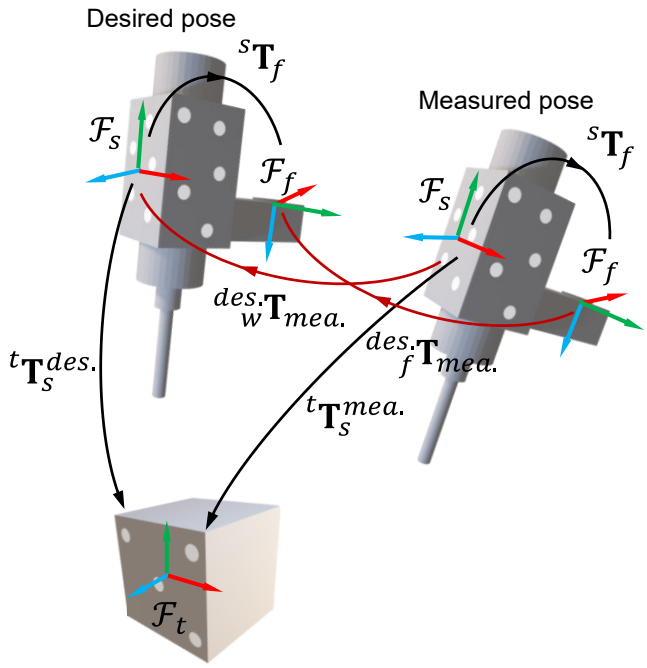

Figure 3.6: Relative pose error measurement between the drilling spindle and fixture.

We can compute the robot relative motion command, ${}^{des.}_{f}\mathbf{T}_{mea.}$, as

$$ {}^{des.}_{f}\mathbf{T}_{mea.} = {}^{s}\mathbf{T}_{f} \,.\, {}^{des.}_{w}\mathbf{T}_{mea.} \,.\, {}^{s}\mathbf{T}_{f}{}^{-1} \,. \tag{3.2} $$

Computation of ${}^{des.}_{f}\mathbf{T}_{mea.}$ requires the unknown transformation between the flange and spindle, ${}^{s}\mathbf{T}_{f}$, that can be found by solving the relative calibration problem of form $\mathbf{AX} = \mathbf{XB}$.

3.5 AX = XB calibration

Figure 3.7 shows the parameters involved in the computation of the transformation between the robot flange and the spindle frame, here shown as **X**, by solving the homogeneous transformation equation given by

$$ \mathbf{AX} = \mathbf{XB} \,. \tag{3.3} $$

A and **B** are the homogeneous transformation matrices representing the relative motion of the spindle frame, $\mathcal{F}_s$, with respect to the measurement frame and the relative motion of the robot flange, $\mathcal{F}_f$, with respect to the robot base frame, respectively.

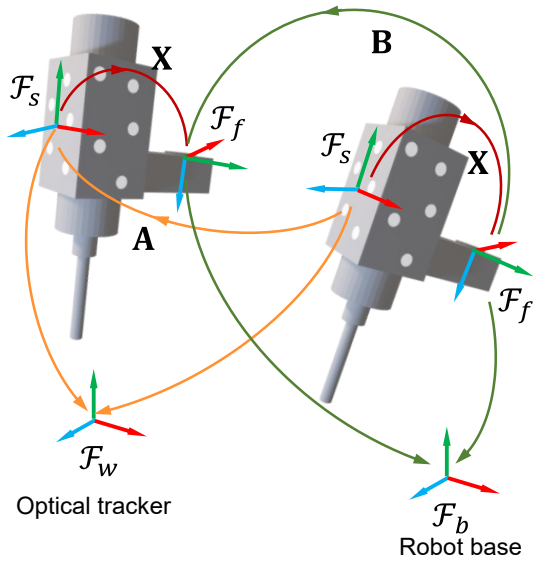

Figure 3.7: Schematics of **AX** = **XB** calibration between the spindle and flange frames.

Eq. (3.3) can be represented in the form of rotation and translation parts as

$$\begin{aligned} \mathbf{R}_A\mathbf{R}_X &= \mathbf{R}_X\mathbf{R}_B \,, \\ (\mathbf{R}_A - \mathbf{I})\mathbf{t}_X &= \mathbf{R}_X\mathbf{t}_B - \mathbf{t}_A \,, \end{aligned} \tag{3.4}$$

where **R** and **t** indicate rotation matrices and translation vectors, respectively. The subscripts A, B and X indicate to which homogenous transformation they belong. The calibration process involves measuring sets of poses from the robot and the measurement system at the same time.

Different methodologies have been proposed in the literature for solving the **AX** = **XB** calibration problem. These methods can be classified into separable and simultaneous solutions. In separable solutions, the rotation part of Eq. (3.4) is first estimated, and then, the translation part is estimated based on the computed rotation. Methods proposed by Chou and Kamel [83] and Park and Martin [45] can be categorized as separable solutions, where they used unit quaternions and Lie-

algebra respectively to represent the rotation parameters. The separable solutions suffer from the loss of the intrinsic relationship between the rotation and translation parameters since the estimated rotation is necessary for computation of the translation vector [84]. To overcome this limitation, simultaneous solutions have been devised. Daniilidis and Bayro-Corrochano [85] introduced the use of dual quaternions as a method for simultaneously estimating the rotation and translation components. Another simultaneous solution was developed by Lu and Chou [86], who formulated a linear system of equations using quaternions. In a different approach, Li et al. [87] incorporated the Kronecker product in their solution. Here, we perform an experimental comparison between these solutions for $\mathbf{AX} = \mathbf{XB}$ calibration problem.

3.5.1 Experimental evaluation

The performance of different calibration solutions is evaluated in the context of robot calibration with respect to the vision measurement system. A set of 100 poses inside the robot workspace are randomly selected (see Figure 3.8). To ensure the visibility of the targets from the cameras, each pose is constrained to the rotation angles in the range of [-15°, 15°] for pitch, roll, and yaw. The performance of different solution methods has been evaluated using the residual transformation error defined as

$$\mathbf{E}_i = \mathbf{A}_i - \mathbf{X}\mathbf{B}_i\mathbf{X}^{-1}. \tag{3.5}$$

This formulation evaluates how accurately the transformation measured from the measurement system can be transformed to the robot flange transformation using the estimated $\mathbf{X}$. The transformation error matrix at each location is then converted to a 6D pose vector in the form of $e_i = \left[x_e, y_e, z_e, r_{x_e}, r_{y_e}, r_{z_e}\right]^T$, and the error for each solution is defined as

$$e_m = \frac{1}{N}\sum_{i=1}^{N}\|e_i\|_2, \tag{3.6}$$

where e_m is the mean of error vector norms, e_i, over the N pair of measurements, $\mathbf{A}_i$ and $\mathbf{B}_i$, used for each solution. e_m is a unitless metrics that indicates how close $\mathbf{A}_i$ and $\mathbf{X}\mathbf{B}_i\mathbf{X}^{-1}$ are using the estimated transformation $\mathbf{X}$.

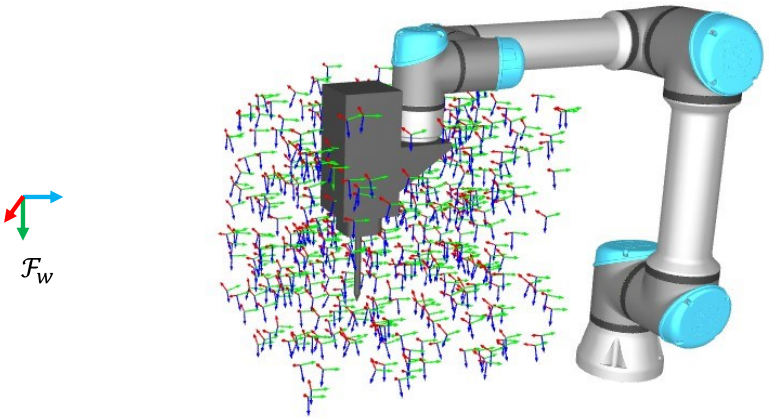

Figure 3.8: Simulation dataset of calibration poses inside the robot workspace.

Figure 3.9 shows the mean error values, e_m, from different calibration methods obtained from 5 to 100 measurements. As can be seen, the simultaneous solution proposed by Li et al. [87] based on Kronecker product yields in the lowest error values. This method also shows the fastest convergence rate as the results converge after 12 measurement points used as the input. Therefore, this method

has been selected in this study for estimating the transformation between flange frame and the spindle frame, ${}^{s}\mathbf{T}_{f}$.

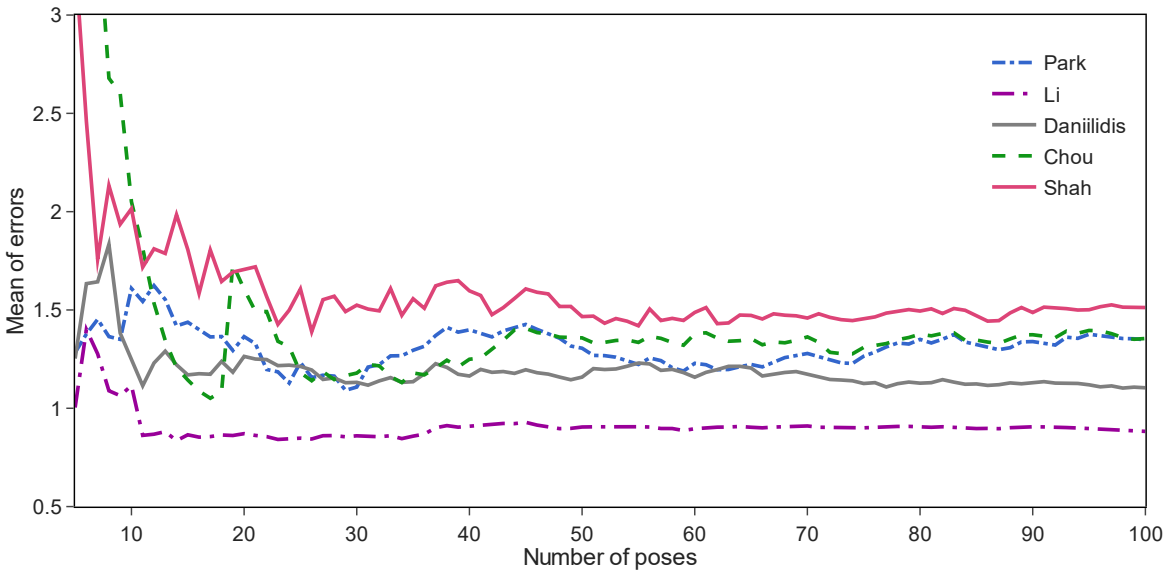

Figure 3.9: Mean error, e_m, obtained from experiment with different methods and different number of measurements.

3.6 Iterative online relative pose correction

Now, using the relative pose error formulation introduced in Section 3.4, we can correct the difference between the desired and measured relative poses to perform online pose correction. Figure 3.10 shows the flowchart of the algorithm used in this study for correcting the relative error in robot. First, the user sets the relative pose between the drilling spindle and the workpiece. The relative pose between the drilling spindle and the fixture is then measured and the robot motion is calculated using ${}^{s}\mathbf{T}_{f}$ transformation, i.e., **X** found in **AX** = **XB** calibration.

Next, the robot is relocated to correct the relative error, and the process continues until the accuracy requirement is met.

Since ${}^{s}\mathbf{T}_{f}$ has been estimated using relatively inaccurate robot poses, there will always be uncertainty in its estimation. As shown in Figure 3.9 even the best performing calibration method has residual errors. The inaccuracy in ${}^{s}\mathbf{T}_{f}$ along with other sources of inaccuracy in robot's motion result in residual error in the corrective motion that the algorithm applies. Therefore, the spindle error with respect to the fixture cannot be corrected in one corrective motion and it requires multiple iterations until the accuracy requirements are met.

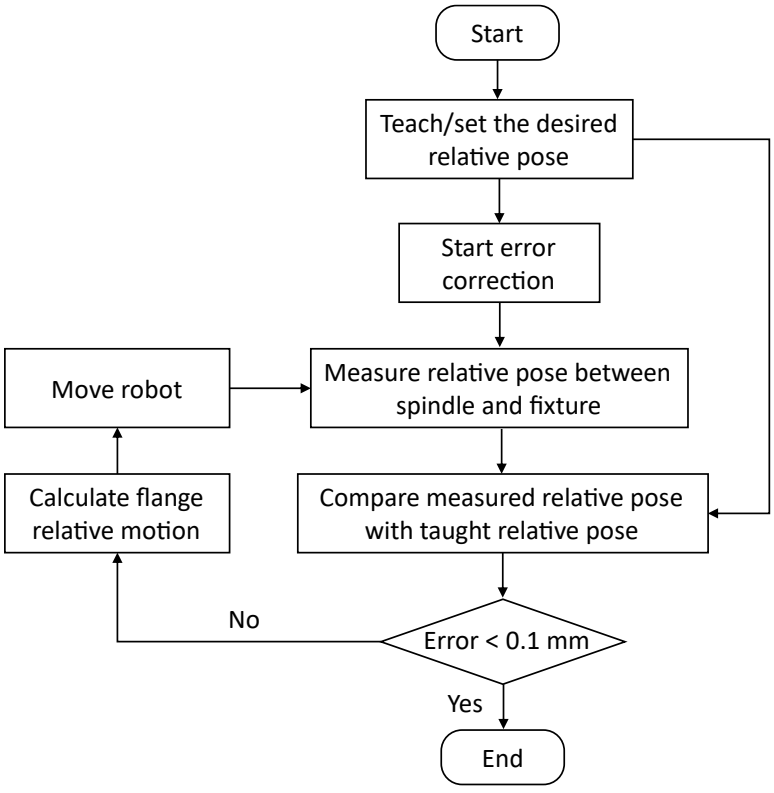

Figure 3.10: Flowchart of error correction algorithm.

3.6.1.1 Measurement noise level

In addition to robot's motion errors, the relative pose measurements from the measurement system, like any other sensory system, has a level of noise that contains the measurement noise from both drilling spindle and fixture. Figure 3.11 shows the noise level in positional and orientational measurements in relative pose between the drilling spindle and the fixture in 100 measurements. The Y-axis is the direction in which the depth estimation is performed using stereo-vision triangulation. Therefore, as can be seen in Figure 3.11, the positional noise level in Y-axis is higher than that of the other axes. Also, since the rotation around Y-axis is estimated from target motions in image plane, X and Z directions, the least amount of noise is visible in relative rotation around Y-axis, i.e., R_y.

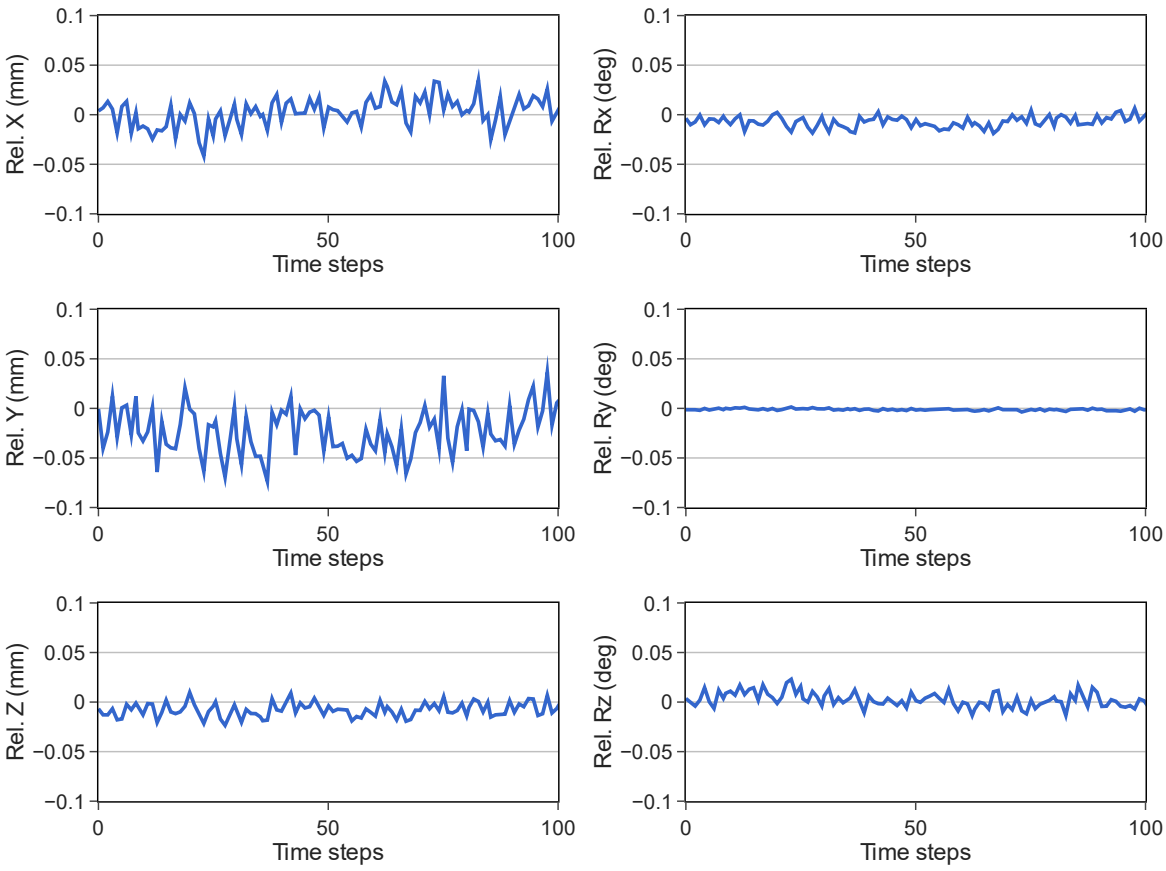

Figure 3.11: Relative positional and orientational measurement noise level in robot base.

3.6.1.2 Experimental evaluation

The developed vision system has been used to correct robot errors in an iterative approach until the positional and orientational component errors are less than 0.1 mm and 0.1°. Relative errors of about 0.1 mm and 1° are considered for the initial positional and orientational errors, respectively. The results of error correction are shown in Figure 3.12 for each individual motion component measured from the vision system. As can be seen, the robot errors are corrected with two steps that satissfies the accuracy requirements.

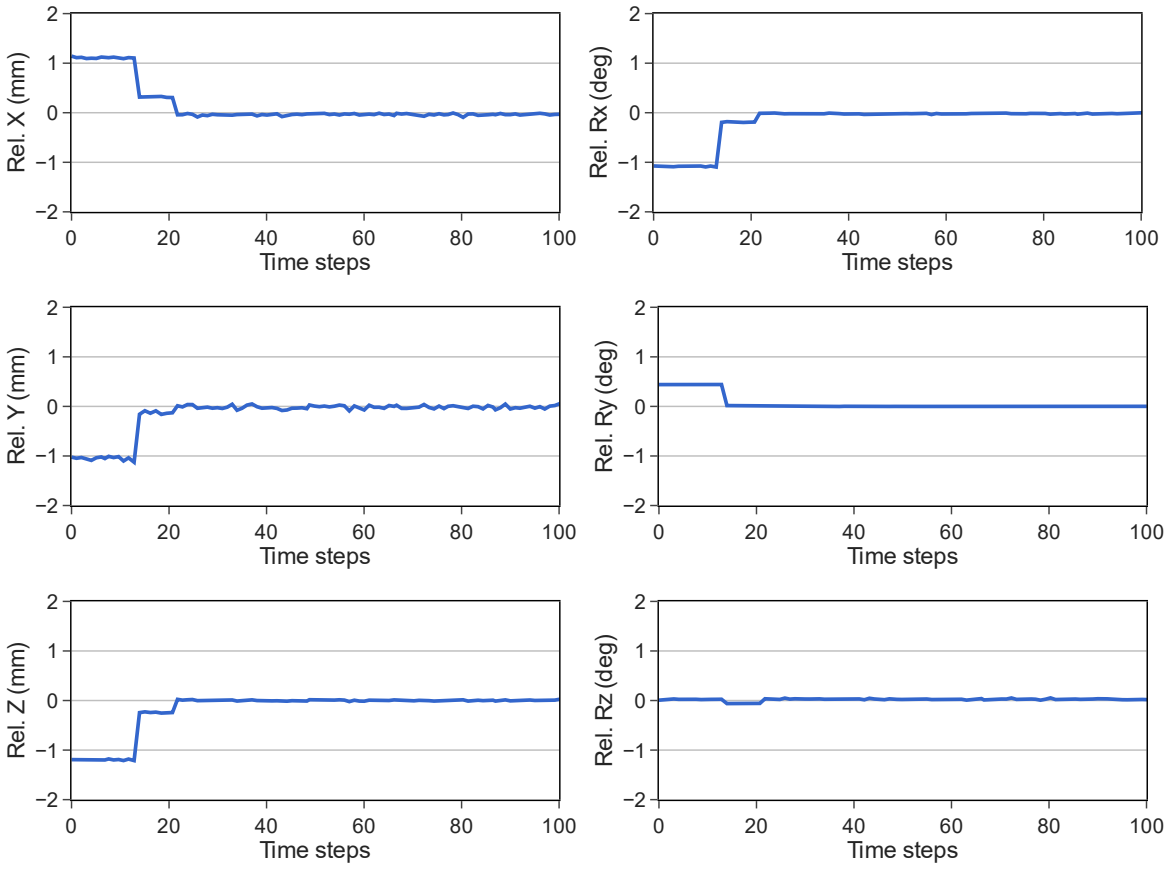

Figure 3.12: Relative positional and orientational error correction in an iterative approach.

3.7 Closed loop relative pose correction

The error correction can also be implemented in the form of a closed feedback loop for a smoother corrective motion. The error between the desired and measured relative poses can be considered as the feedback signal for the controller to perform online pose correction. As shown in Figure 3.13, the feedback signal from the vision system provides the measured pose of the spindle and fixture, which is then used for computing the relative pose signal, ${}^{t}\mathbf{T}_{s}^{mea.}$. The desired relative pose,

${}^{t}\mathbf{T}_{s}^{des.}$, is also set by user, i.e., reference input. The relative pose error signal, ${}_{w}^{des.}\mathbf{T}_{mea.}$, is then calculated in the measurement frame. The robot motion command, ${}_{f}^{des.}\mathbf{T}_{mea.}$, is then estimated and used as the input signal for the controller. ${}_{f}^{des.}\mathbf{T}_{mea.}$ is converted to the 6D vector format consisting of the translational and rotational rigid body motion components that need to be applied to the robot flange frame relative to its current pose. The controller output is the motion command sent to the robot controller and later to the robot.

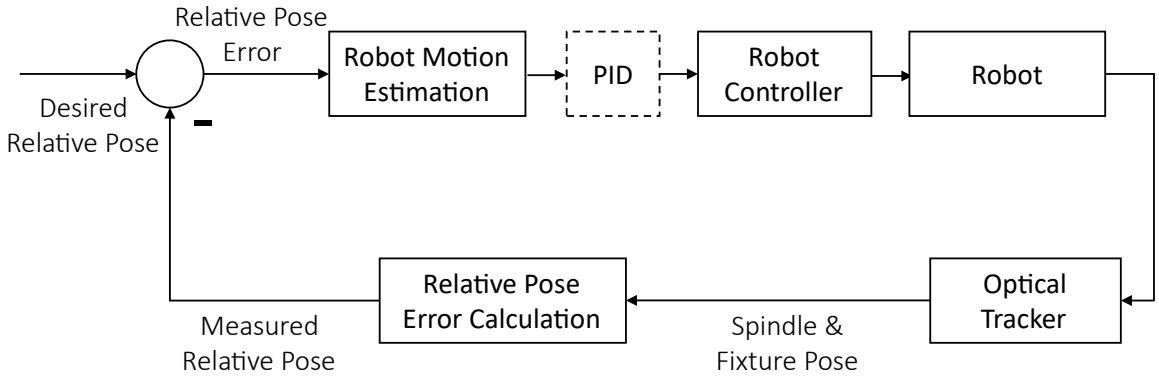

Figure 3.13: Relative pose error correction control loop.

The PID controller for the robot motion can be expressed as

$$u_i(t) = K_p e_i(t) + K_i \int e_i(t)\, dt + K_p \frac{de_i}{dt}, \tag{3.7}$$

where $u_i(t)$ is the PID output to the robot controller, and $e_i(t)$ is the relative pose error with $i = x, y, z, r_x, r_y, r_z$. The coefficients K_p, K_i, K_d are the proportional, integral and derivative gain values, respectively, that require tuning for a desirable performance. The selected controller gains after tuning are listed in Table 3.1.

Table 3.1: Controller gain values

Motion	K_p	K_i	K_d
x, y, z	0.6	0.05	0.05
r_x, r_y, r_z	0.4	0.01	0.05

3.7.1.1 Controller performance

In order to demonstrate the performance of the developed and tuned controller in online error correction, the robot error in each individual axis has been corrected. For this test, the drilling spindle has been placed in a distance from the desired pose in each axis, and then, the controller has been used to correct the error. Figure 3.14 shows the recorded axis errors over the time of experiment for positional and orientational errors. As can be seen, the controller is able to converge to errors less than tolerance requirements by gradually and smoothly approaching to the desired reference point in each individual motion axis.

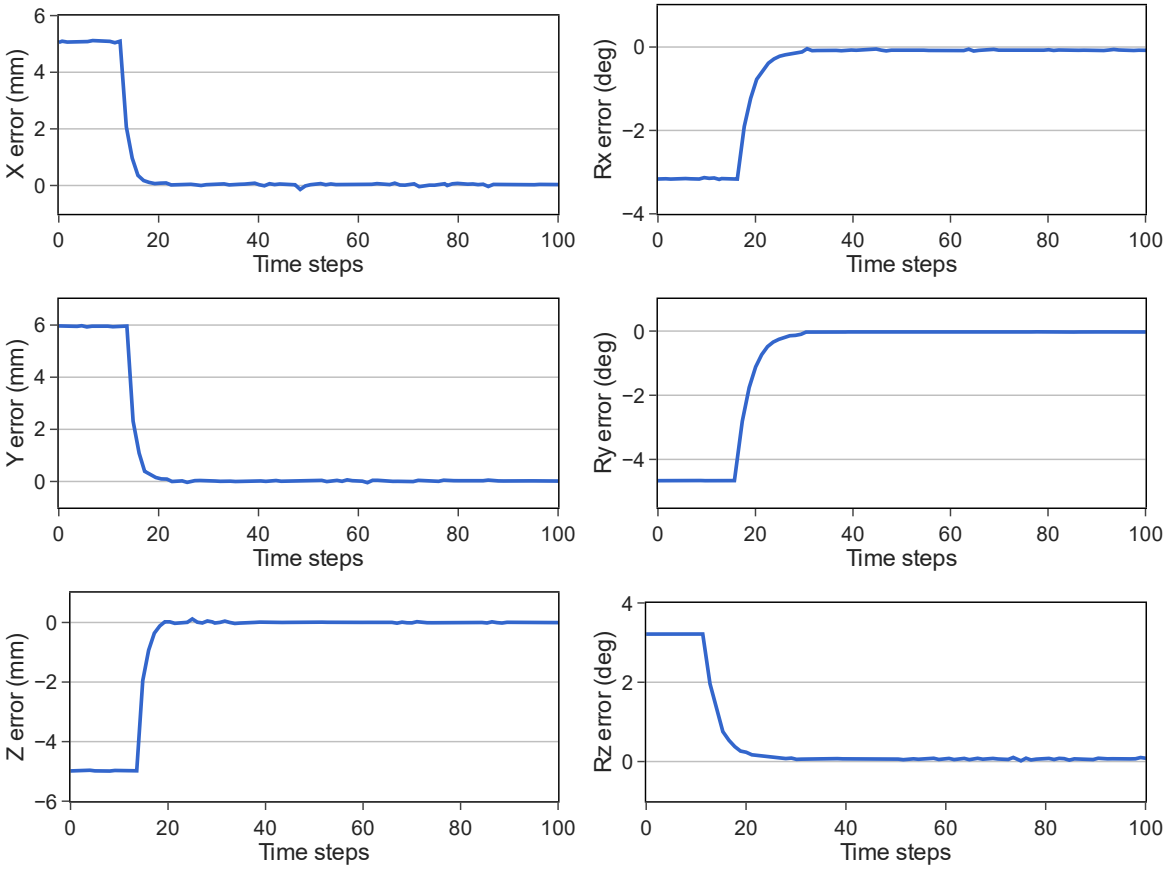

Figure 3.14: Relative positional and orientational error correction using PID controller.

3.7.1.2 Online 6D pose correction

Here, we test the developed system and controller in a general 6D error correction scenario. In this case, the robot is relocated with respect to the fixture to create positional and orientational errors of about 1 mm and 1°, respectively, in all axes of motion simultaneously. Figure 3.15 shows the result of error correction measured by the measurement system. As can be seen, the controller is able to correct the errors by gradual motion of spindle with respect to the workpiece.

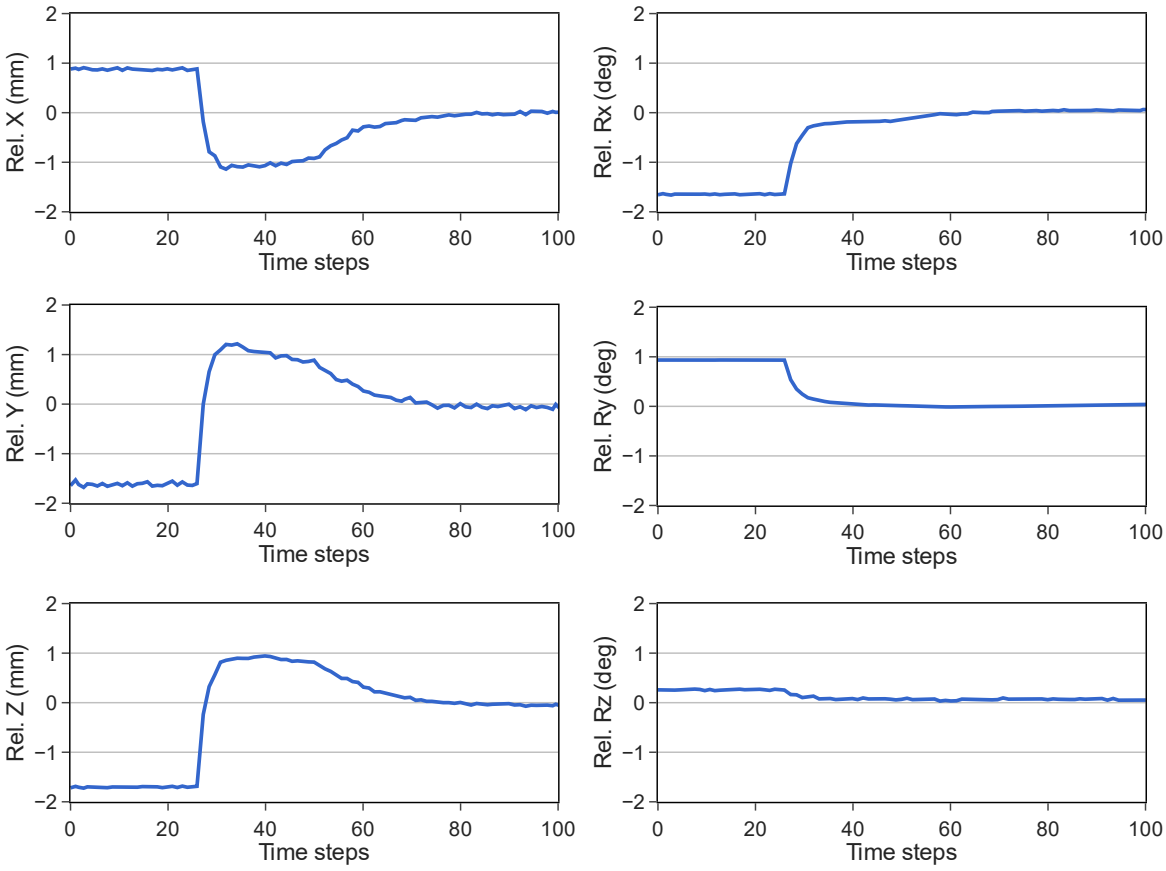

Figure 3.15: Relative positional and orientational error correction in 6D error correction using PID controller.

3.8 Summary

In this chapter, we presented an online vision-based error correction approach based on relative pose measurements between the drilling spindle and workpiece. We formulated the relative robot motion command from the data of the measurement system. We experimentally compared different calibration methods in the context of relative motion correction. We implemented an iterative approach to correct the robot's errors with respect to the workpiece, and showed the

performance of the solution using the developed vision system. To make the robot's corrective motion smoother, we implemented and tuned a PID controller that uses the measurement data from the vision system as the feedback signal. The PID controller has been tuned and the performance of the developed solution in correcting each individual motion axis and also general 6D errors has been demonstrated.

Chapter 4

Offline Relative Pose Correction

This chapter aims to address the implementation of the relative error correction technique to improve the accuracy of dual-robot drilling cells in the aerospace industry. This work has been performed during the author's internship at the National Research Council Canada (NRC) Aerospace Manufacturing Technologies Center (AMTC) in Montreal, QC, Canada.

Note: The robot pose data reported in this chapter were measured using a laser tracker available at the NRC. However, all the methods and analyses presented here can readily be used with the developed vision-based measurement system (presented in Chapter 2) as well.

4.1 Background

Large aerospace structures are usually carried and processed using multi-robot cells, i.e., cooperative systems. Multi-robot systems can be considered as an extension to the conventional mono-robots. They can reduce the technical limitations of manipulating large and complex parts. In these systems, multiple robots are in communication with each other. Usually, one robot acts as the "master" robot and controls other robots in the system, which are called "slave" robots. In this case, the information flows from the master robot to the slave robots, resulting in a minimal cost of synchronization as opposed to an all-to-all communication system [88, 89].

In a cooperative system, the master-slave configurations can be classified into two types in terms of robot motion [90]: (1) direct geometric coupling, and (2) indirect geometric coupling. In direct mode, the slave robots act in parallel with the master robot and track its motions. A good example of this type can be seen in load sharing tasks between two robots [91]. Figure 4.1 (a) shows an industrial application of load sharing between two robots in processing large aerospace body panels.

On the other hand, in the indirect geometric coupling case, the slave robot follows the master robot's basic motions, such as linear or circle arc, and performs its own superposed motion blocks too [90]. In this scenario, the robots' end-effectors can be physically disconnected from each other, e.g., the inspection process of large aircraft panel using vision systems [20], where the master robot holds the part and presents the inspection regions to the slave robot that holds the

inspection system. On the other hand, the cooperative task may require physical contact between the robots for tasks such as robotic machining. Figure 4.1 (b) shows an example of robot cooperation in drilling applications. The slave motion in this case is planned relative to the master tool in a way that ensures the establishment of a time-varying geometric coupling.

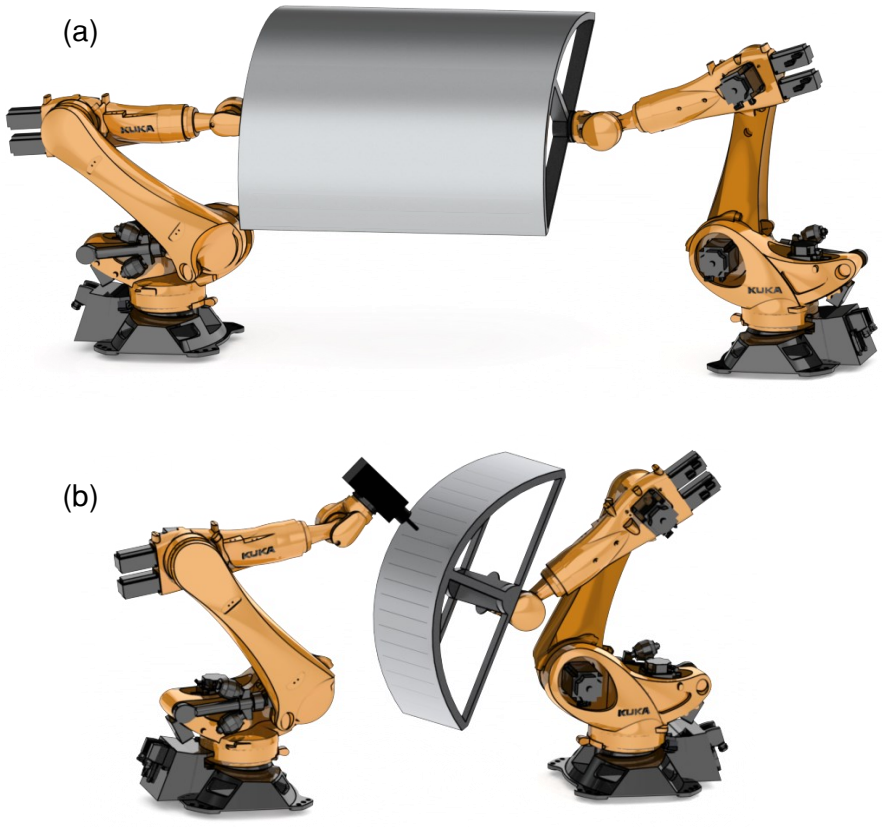

Figure 4.1: Examples of dual-robot cooperative system in (a) indirect and (b) direct geometric coupling modes

The general layout of a robotic cell and the performed operations can be simulated in a virtual environment using CAD models, often referred to as off-line programming. However, due to differences between the actual robot physics and

the nominal behavior simulated using off-line programming software, there are always inaccuracies between the programmed and actual robot trajectories. These inaccuracies can reach values up to several millimeters in a mono-robot operation. In multi-robot systems, these errors can further add up depending on the number and relative positioning of robots with respect to each other. In load sharing tasks, even small errors in the relative positioning of the robot end-effectors may lead to loosening or over-tightening of the fixtures. This poses the risk of dropping the part or creating a large amount of internal forces that can damage the part or robots. Similarly, in cooperative tasks involving the execution of a process by one robot on a part manipulated by the second robot, the relative inaccuracy between the robots can directly degrade the tolerance of the manufactured parts. To prevent these issues in multi-robot cooperative settings, our main objective in this chapter is to ensure that the slave robots follow the master's trajectory as accurately as possible. This is achieved by directly reducing the relative errors between the robots instead of focusing only on improving the accuracy of each robot individually. We investigate the direct and indirect geometric coupling modes (Figure 4.1), where the slave robot must accurately operate in master robot's coordinate frame.

To achieve a high level of accuracy in robotic operations, it is crucial to perform robot calibration. Calibration methods can be categorized into model-based and model-free methods [92]. In model-based approach, the sources of error in robot parameters are considered as model parameters. For example, finding the

deviation of kinematic parameters from the nominal robot model, so called geometric calibration, is a common model-based approach [93]. More complex model-based methods that consider non-geometric parameters, e.g. structural compliance [94, 95], are also developed to improve the accuracy of robots. While model-based methods require fewer number of measurements, their modeling becomes quickly complex in case of including non-geometric parameters.

On the other hand, model-free methods do not require a mathematical model to recognize the deviations from the nominal robot. They directly compensate the error from measurement data, which indeed include all sources of error, i.e., geometric and non-geometric, in the real robot. The process of model-free approach normally starts with measuring robot end-effector errors in the robot workspace on pre-defined grids [96, 97] or sampled points [98]. Then, a regression model is used to find the best fit for the error function of the robot, given the measured data points. Different methods have been used to find the best fit model, including fuzzy interpolation [96], polynomial approximation [99], and ANNs [100]. A simulation study by Bai and Wang [101] shows that ANNs can achieve a higher performance in model-free calibration of mono-robots in comparison with popular fuzzy and polynomial approaches.

In recent years, several studies have been carried out in mono-robot calibration based on ANNs [101–106]. ANNs have been applied to either uncalibrated robots [101, 105], or calibrated robots [97, 100, 103] for correcting their residual errors. In our work, ANNs are used for correcting dual-robot

cooperative systems in both scenarios of a) correcting uncalibrated robots, which compensates for both geometrical and non-geometrical errors and b) compensating residual errors after geometric calibration, which mostly concerns non-geometrical error effects. Two ANN models are trained to predict the error functions of the robots in two stages. The first stage aims to correct the relative error between the robots by compensating the slave robot pose with respect to the master robot. This stage ensures that the slave robot follows the master robot with high accuracy. This approach directly reduces the error of handling a part or cutting tool with respect to the master robot. Then, a secondary ANN model learns to correct the master robot as a mono-robot, to ensure the absolute accuracy of the cooperative system. Also, by comparing the compensation results from uncalibrated robots and geometrically calibrated robots, the ability of ANN models in improving the geometrically calibrated robots in a cooperative system is investigated. In addition, when training a neural network, the optimal number of trainable parameters and architecture are not known in advance. This can result in a larger than required network which reduces the ability of generalization on a given set of training data. We have used a five-fold cross-validation process for selecting the optimal network parameters to address this issue for the two stages of calibration.

The contributions of this chapter are as follows:

(1) A novel calibration method for master-slave collaborative systems is suggested, which is based on the decoupling of relative master-slave errors and absolute master robot errors, thus allowing for accuracy improvement

compared to the calibration of each individual robot in a collaborative system.

(2) An experimental methodology and setup for the simultaneous calibration of a dual-robot master-slave configuration using a laser tracker have been developed.

(3) An error compensation approach based on ANNs is implemented for relative and absolute error compensation in a dual-robot system.

(4) Furthermore, the proposed compensation strategy is applied to pre-calibrated robots to predict the residual errors in a collaborative system.

The remainder of this chapter is structured as follows: Section 4.2 discusses the methodology for decoupling master-slave collaborative errors into relative and absolute parts. Section 4.3 presents the experimental methods and procedures for data measurement. Section 4.4 covers the process of developing ANNs for error prediction and finding the optimal network and training parameters. Section 4.5 provides the simulation results and analysis. Section 4.6 draws conclusions based on our research.

4.2 Methodology

Figure 4.2 shows the schematic of a dual-robot system with the kinematic chain in a master-slave configuration. The location of the robots' frames is represented using a homogeneous transformation matrix as

$$ {}^{j}\mathbf{T}_i = \begin{bmatrix} \mathbf{R} & \mathbf{t} \\ \bar{0} & 1 \end{bmatrix}_{4\times4}, \text{ where } \quad {}^{j}\mathbf{T}_i^{-1} = {}^{i}\mathbf{T}_j = \begin{bmatrix} \mathbf{R}^T & -\mathbf{R}^T\mathbf{t} \\ \bar{0} & 1 \end{bmatrix}_{4\times4} \tag{4.1} $$

where ${}^{j}\mathbf{T}_i$ is the homogeneous transformation from frame i to j, which transforms the positional vector of any point in frame i, denoted as $\mathbf{v}_i$, to the coordinates in frame j using $\mathbf{v}_j = {}^{j}\mathbf{T}_i\mathbf{v}_i$. In Eq. (1), $\mathbf{t}$ is the 3×1 translation vector, defined as $[x, y, z]^T$. $\mathbf{R}$ is the 3×3 orthogonal rotation matrix and here is expressed with Euler angles defined as

$$ \mathbf{R}(a, b, c) = \mathbf{R}_z(a) \,.\, \mathbf{R}_y(b) \,.\, \mathbf{R}_x(c) \tag{4.2} $$

with a, b and c as Euler angles. Alternatively, a homogeneous transformation ${}^{j}\mathbf{T}_i$ can be expressed in the form of a 'pose' vector ${}^{j}\mathbf{p}_i$ with 6 independent variables (3 translations and 3 orientations) as

$$ {}^{j}\mathbf{p}_i = [x,\ y,\ z,\ a,\ b,\ c]^T. \tag{4.3} $$

In the dual-robot system shown in Figure 4.2, important Cartesian frames that form the kinematic chain are noted as the world frame, $\mathcal{F}_w$, slave flange, $\mathcal{F}_{sf}$, slave base, $\mathcal{F}_{sb}$, master flange, $\mathcal{F}_{mf}$, and master base, $\mathcal{F}_{mb}$. We should add that Tool-Center-Point calibration has become a standard and well-mastered procedure in the commercial off-the-shelf mono-robot calibration software, and it is therefore not addressed in this work for the sake of simplicity. The world frame, $\mathcal{F}_w$, is a fixed coordinate frame which is usually assigned to the measurement device. In our case, the laser tracker coordinate frame is the world frame. Using the laser tracker and a base frame calibration process [107, 108], we can determine the pose

of master and slave base frames with respect to $\mathcal{F}_w$, i.e., ${}^{w}\mathbf{T}_{mb}$ and ${}^{w}\mathbf{T}_{sb}$, respectively.

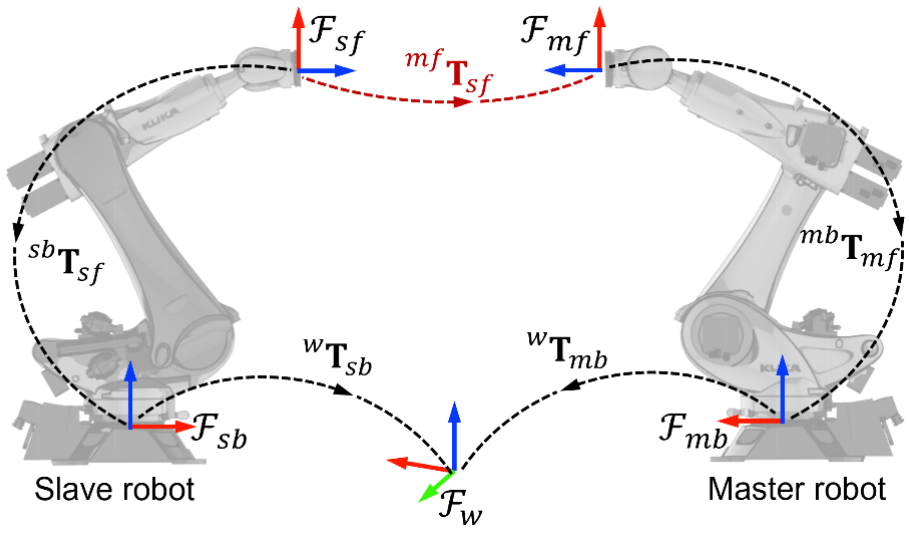

Figure 4.2: Important frames in a dual-robot system forming a kinematic chain

${}^{mf}\mathbf{T}_{sf}$ is the nominal transformation between the slave flange and the master flange, and can be found as

$$ {}^{mf}\mathbf{T}_{sf} = {}^{mb}\mathbf{T}_{mf}^{-1}.\ {}^{w}\mathbf{T}_{mb}^{-1}.\ {}^{w}\mathbf{T}_{sb}.\ {}^{sb}\mathbf{T}_{sf}, \tag{4.4} $$

where ${}^{mb}\mathbf{T}_{mf}$ is the transformation matrix from master flange frame, $\mathcal{F}_{mf}$, to master base frame, $\mathcal{F}_{mb}$, and ${}^{sb}\mathbf{T}_{sf}$ is the transformation from slave flange frame, $\mathcal{F}_{sf}$, to slave base frame, $\mathcal{F}_{sb}$. It is assumed that the robots are rigidly mounted and do not move with respect to each other during the operation. Therefore, ${}^{w}\mathbf{T}_{mb}$ and ${}^{w}\mathbf{T}_{sb}$ are constant transformations.

In the case of direct geometric coupling, i.e., load sharing, the robot flanges are geometrically coupled through the workpiece. It is assumed that the workpiece is a rigid body and there is no relative motion between the robot flanges and the

workpiece. Hence, ${}^{mf}\mathbf{T}_{sf}$ is nominally constant during the cooperative task. On the other hand, in the indirect geometric coupling, the slave robot is not rigidly connected to the master robot and can perform its own motion blocks to process the workpiece that is held by the master robot. In this case, the process is planned in the workpiece coordinate frame. It is assumed that the workpiece is rigidly attached to the master flange. Hence, the master flange can be used as the process coordinate frame. Therefore, ${}^{mf}\mathbf{T}_{sf}$ can be considered as the time-varying relative constraint in the master-slave motion in the indirect geometric coupling tasks. Whether ${}^{mf}\mathbf{T}_{sf}$ is constant or time-varying, making sure that the transformation ${}^{mf}\mathbf{T}_{sf}$ is maintained as accurately as possible during cooperative tasks is crucial for the safety and accuracy of processed workpiece. To this end, we propose a new approach that considers both absolute and relative accuracy of master-slave robot configurations. The detailed methodology is presented in the following sections.

4.2.1 Relative vs absolute accuracy

In this work, we propose a two-stage approach for calibration of a master-slave robot configuration to correct: (a) the master-slave relative errors and (b) the master robot's errors. By correcting the master-slave relative error, one can ensure the safety of the workpiece carried by the robots and the accuracy of the cooperative operation. On the other hand, when the goal is to increase the absolute accuracy of the whole system, both stages of calibration are required to be done. In this work, the two stages of calibration are dealt with in parallel due to

operational constraints that limited the possible number of measurements. Therefore, the system has been studied using two sets of measurements on both robots at the same time, once before and once after geometric calibration of the robots. However, in an industrial setting, it would be more practical to first calibrate the master robot, and then perform a new set of measurements on both robots using the calibrated master robot data to perform the proposed second stage of calibration.

4.2.2 Master-slave relative error

Let's define ${}^{mf}\mathbf{T}_{sf}$ as the homogeneous transformation between the nominal slave flange, $\mathcal{F}_{sf}$, and nominal master flange, $\mathcal{F}_{mf}$. This transformation is known from the motion blocks (constant or time-varying) that the slave robot needs to generate in a frame moved by the master robot. However, in real robots there are geometric and non-geometric sources of error that cause deviation from the nominal behavior. To calculate these errors, we need to measure the real pose of both robot flanges using a measurement device, e.g., a laser tracker. Assuming $\mathcal{F}_w$ as the reference coordinate frame of the laser tracker, the measurement process gives us the "measured" master flange and slave flange frames, $\mathcal{F}_{mf'}$ and $\mathcal{F}_{sf'}$, respectively, with respect to the measurement frame, $\mathcal{F}_w$. Figure 4.3 shows a schematic view of the measured and nominal robot postures and their corresponding flange frames. For the sake of illustration, the scale of errors in Figure 3 is exaggerated. The errors normally do not exceed a couple of millimeters in mid-size robots. The

measured poses of $\mathcal{F}_{mf'}$ and $\mathcal{F}_{sf'}$ with respect to $\mathcal{F}_w$ can be expressed as transformation matrices ${}^{w}\mathbf{T}_{mf'}$ and ${}^{w}\mathbf{T}_{sf'}$. We can calculate the measured flange-to-flange transformation ${}^{mf'}\mathbf{T}_{sf'}$ as

$$ {}^{mf'}\mathbf{T}_{sf'} = {}^{w}\mathbf{T}_{mf'}^{-1}.\,{}^{w}\mathbf{T}_{sf'}, \tag{4.5}$$

which is the measured flange-to-flange transformation when the master and slave robots are given the joint angles $\boldsymbol{\theta}_m = [\theta_1 \ \dots \ \theta_6]_m^T$ and $\boldsymbol{\theta}_s = [\theta_1 \ \dots \ \theta_6]_s^T$, respectively. Our goal is to keep ${}^{mf}\mathbf{T}_{sf}$ as accurately as possible for any given set of joint angles, $\boldsymbol{\theta}_m$ and $\boldsymbol{\theta}_s$, as the system inputs. In other words, the slave flange must always maintain the nominal pose of ${}^{mf}\mathbf{T}_{sf}$ with respect to the master flange even if there are path generation errors in each robot individually. Now, to decouple the error compensation problem, we assume that the slave robot is responsible for correcting the deviation of the measured flange-to-flange transformation, ${}^{mf'}\mathbf{T}_{sf'}$, from the nominal one, ${}^{mf}\mathbf{T}_{sf}$. Concretely, given the set of input joint angles, $\boldsymbol{\theta}_m$ and $\boldsymbol{\theta}_s$, and regardless of the pose error of the master flange, the slave flange must be relocated to have a pose of ${}^{mf}\mathbf{T}_{sf}$ with respect to the master flange. This assumption allows us to define a compensated pose for the slave flange in the master flange frame. To find this compensated pose, we use the difference between measured and nominal transformations as

$$ \mathbf{E}_{rel.} = {}^{mf}\mathbf{T}_{sf}^{-1}.\,{}^{mf'}\mathbf{T}_{sf'}, \tag{4.6}$$

where $\mathbf{E}_{rel.}$ is the master-slave relative pose error.

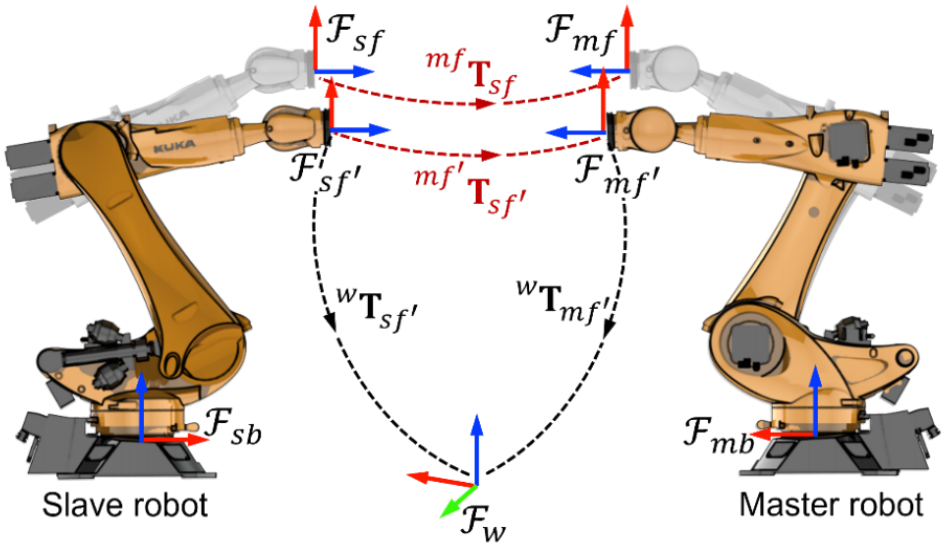

Figure 4.3: Schematics of nominal and measured robot postures and their corresponding flange frames

By subtracting $\mathbf{E}_{rel.}$ from the nominal flange-to-flange transformation matrix, ${}^{mf}\mathbf{T}_{sf}$, we generate a compensated pose in the measured master flange that corrects the relative error between the master and slave robots:

$$ {}^{mf}\mathbf{T}_{sf_{comp.}} = \mathbf{E}_{rel.}^{-1} \,.\, {}^{mf}\mathbf{T}_{sf} \tag{4.7}$$

Figure 4.4 shows the diagram of correcting the master-slave relative error by relocating the slave flange to the compensated pose in the master frame, ${}^{mf}\mathbf{T}_{sf_{comp.}}$. As shown in Figure 4.4, the slave flange is moved to a new compensated pose, $\mathcal{F}_{sf_{comp}}$, so that it retains the nominal transformation ${}^{mf}\mathbf{T}_{sf}$ between the robot flanges. It should be noted that the master-slave relative error, $\mathbf{E}_{rel.}$, is a function of the joint space in both robots, $\boldsymbol{\theta}_m$ and $\boldsymbol{\theta}_s$, and the contribution of each robot in forming the relative error varies in its own joint space. However, since the goal is to correct the relative error by only relocating the slave flange with respect to the

master flange, as per our decoupling assumption, we can assume that the measured and nominal master flanges coincide, $\mathcal{F}_{mf} \approx \mathcal{F}_{mf'}$.

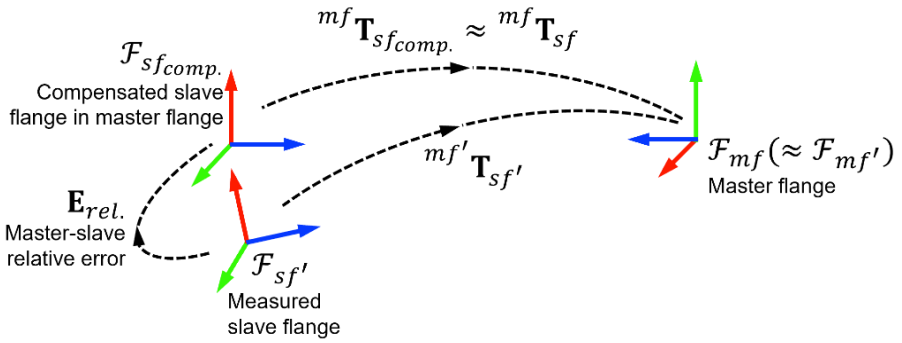

Figure 4.4: Slave flange compensation relative to the master flange

As shown in Eq. (4.7), relocating the slave flange requires the nominal flange-to-flange transformation, $^{mf}\mathbf{T}_{sf}$, and the master-slave relative error, $\mathbf{E}_{rel.}$. $^{mf}\mathbf{T}_{sf}$ is given in the robot program to create the motion of the slave robot with respect to the master robot. We only need measurements to find $\mathbf{E}_{rel.}$ for each pair of coordinates in the master-slave joint space. Here, we rewrite $\mathbf{E}_{rel.}$ in the vector form, $\boldsymbol{\varepsilon}_{rel.}$, as the master-slave relative error vector:

$$\boldsymbol{\varepsilon}_{rel.} = \left[\varepsilon_x,\ \varepsilon_y,\ \varepsilon_z,\ \varepsilon_a,\ \varepsilon_b,\ \varepsilon_c\right]^T_{rel.}. \tag{4.8}$$

In the next section, we train an ANN regression model based on a set of training data that can predict $\boldsymbol{\varepsilon}_{rel.}$ given the joint space of both robots as the input.

4.3 Master-slave absolute accuracy

By compensating for the master-slave relative error, we correct the slave path to ensure the safety and accuracy of cooperative tasks. Now, to increase the absolute

accuracy of the system, e.g., in placing the part on a fixture, we perform the absolute calibration process for the master robot only. Here, to calibrate the master robot, we use another ANN model that predicts the errors of master robot for different joint angle inputs. Figure 4.5 shows the transformations necessary for calibration of the master robot. The goal is to find the deviation of the measured master flange, $\mathcal{F}_{mf'}$, from the nominal master flange, $\mathcal{F}_{mf}$, with respect to the master base, $\mathcal{F}_{mb}$, in which we program the master robot motion blocks. This deviation can be found by forming a kinematic chain from the measured master flange to the nominal master flange, see Figure 4.5, and calculating the pose error transformation matrix, $\mathbf{E}_m$, as

$$\mathbf{E}_m = {}^{mb}\mathbf{T}_{mf}^{-1}.\ {}^{w}\mathbf{T}_{mb}^{-1}.\ {}^{w}\mathbf{T}_{mf'}, \tag{4.9}$$

where ${}^{mb}\mathbf{T}_{mf}$ is the nominal pose and given, ${}^{w}\mathbf{T}_{mb}$ is a constant transformation and found through base calibration, and ${}^{w}\mathbf{T}_{mf'}$ is the measured pose of the master flange. Here, we rewrite $\mathbf{E}_m$ in the vector form, $\boldsymbol{\varepsilon}_m$, as the master robot error vector:

$$\boldsymbol{\varepsilon}_m = \left[\varepsilon_x,\ \ \varepsilon_y,\ \ \varepsilon_z,\ \ \varepsilon_a,\ \ \varepsilon_b,\ \ \varepsilon_c\right]_m^T. \tag{4.10}$$

$\boldsymbol{\varepsilon}_m$ is a function of the master joint space. In the next section, we train an ANN model to predict the master robot error, $\boldsymbol{\varepsilon}_m$, given the master joint angles, $\boldsymbol{\theta}_m$, as input.

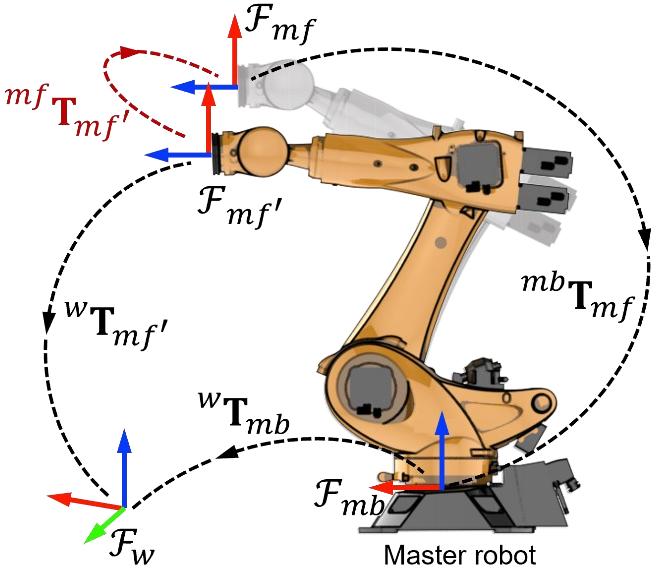

Figure 4.5: Schematics of master flange pose error calculation

4.4 Experimental data measurement

To train ANN models for a multi-robot cooperative system, we need to measure the pose errors of master and slave robots in their shared workspace at different points. In this work, a dual-robot cell comprising two identical KUKA KR300 R2500 ULTRA robots were used, see Figure 4.6. For measurements, a Leica LT800 laser tracker along with spherically mounted reflector (SMR) targets were used. As shown in Figure 6, the measurement tooling for both robots is made of steel plates housing three SMR targets, each of which facing the laser tracker.

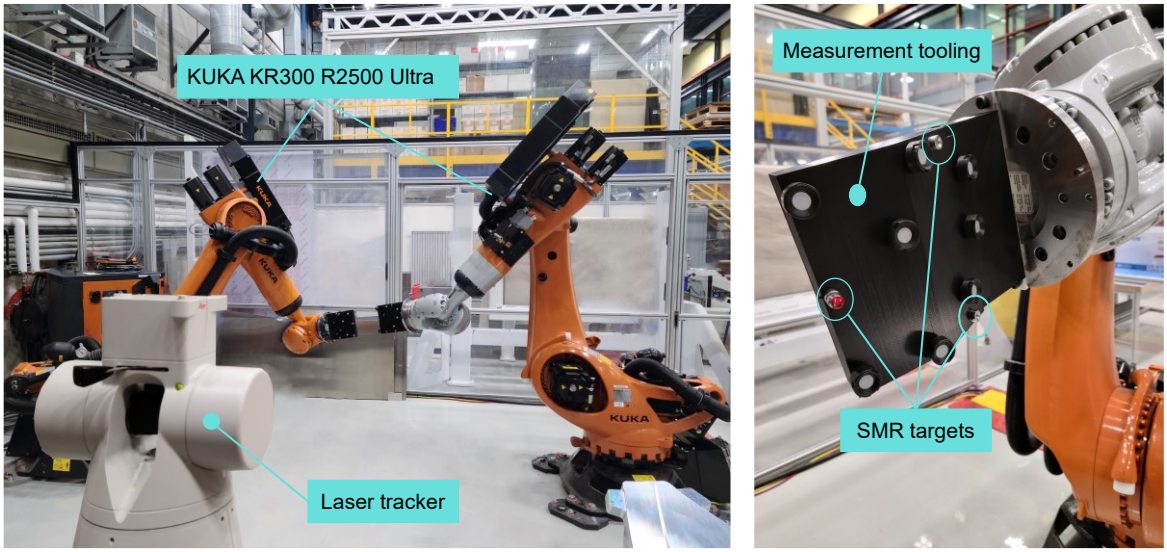

Figure 4.6: Experimental master-slave setup including two KUKA KR300 R2500 ULTRA robots, Leica LT800 laser tracker, measurement tooling, and SMR targets

4.4.1 Identification of base frame and flange frame

Before starting the data measurement process for robot calibration, the base and flange frames of each robot must be identified. For both cases, circle point analysis (CPA) [109] was used in RoboDK. RoboDK is a commercial software for robot offline programming. Using the CPA method, the best fit plane is found by measuring at least 3 points on a circle created by moving a joint (Figure 4.7). The normal to the plane at the center of the circle identifies the joint axis. We can identify the base and flange frames by performing the CPA method for perpendicular axes. For the case of base frame, the CPA method was applied to the first and second joints. A magnetic SMR housing was attached to the body of the robot on links 1 and 2. The SMR was located on the housing, and joints 1 and 2 were moved to the extent that the SMR position could be measured by the laser

tracker. To increase the accuracy of results, between 12 to 22 points on each circle were measured, depending on visibility of the SMR to the laser tracker. These points are then used to find the center, axis, and plane of the circle for constructing the base frame. A similar process is performed for the flange frame. The SMR has been attached to the measurement tooling and moved by joints 5 and 6, see Figure 4.7.

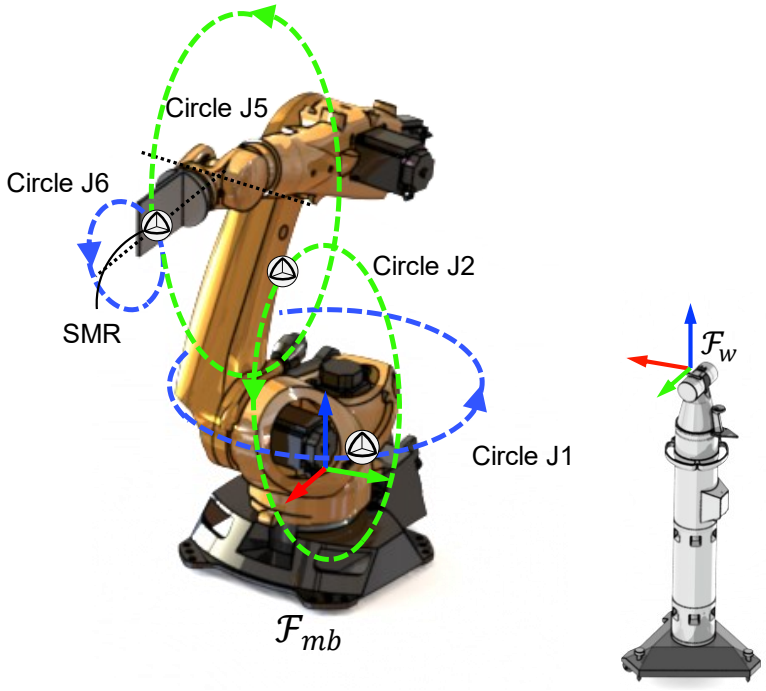

Figure 4.7: Identification process for robot base frame and flange frame in the world frame

4.4.2 Measurement procedure

The measurement process includes sending different pose commands in the Cartesian space to master and slave robots and measuring the real pose reached by the robots at the flange frame. The list of joint angles and their corresponding

measured flange poses are then used for generating the training data. The robot configurations in the joint space have been defined in a way that are representative of the target cooperative tasks. Each pair of points represent a measurement case in which the two flanges are aligned and placed at a distance ranging from 75 cm to 95 cm. The point pairs are distributed in the workspace to cover the calibration volume required in the desired collaborative tasks. As a result, the master and slave flanges are positioned in their common workspace, where the robots are expected to work on the same workpiece. Both robots are sent to 1000 randomly generated points in their common workspace, as shown in Figure 4.8.

The points are generated in a way that ensures the feasibility of the measurements by checking for the direct line of sight between the laser tracker and the reflectors. The robots and the workspace volume are built in a RoboDK environment as a digital twin and all points are simulated to check for the robot reachability and collision. The robots are then sent to the points and their flange pose is measured at each point using the laser tracker. After measuring the uncalibrated robot poses, both robots are geometrically calibrated using the axis calibration module in RoboDK, and the calibrated robots are again measured at 1000 points. In the next sections, the collected data from uncalibrated and geometrically calibrated robots will be used to train ANN models for robot error correction.

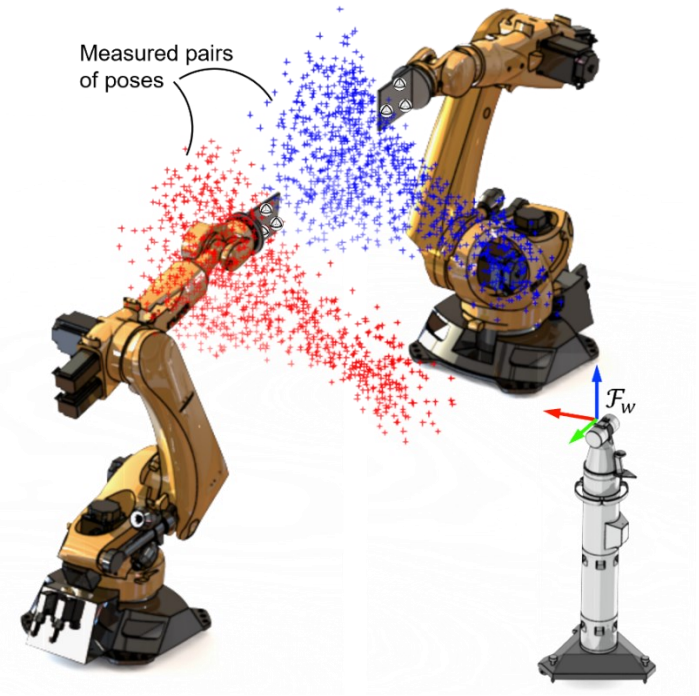

Figure 4.8: Schematics of data measurement volume and the pair of measurement poses

4.5 Neural network architecture

In this work, we employ fully dense multi-layer perceptron (MLP) neural network architectures for predicting the master-slave relative errors, $\boldsymbol{\varepsilon}_{rel.}$, and master's absolute errors, $\boldsymbol{\varepsilon}_m$. An MLP network consists of a series of fully connected layers, called input, hidden, and output, see Figure 4.9. In the context of robot calibration, the joint angles are used as the input neurons and the error parameters as the output neurons. Each hidden layer includes trainable parameters, weights and biases, followed by a nonlinear function, activation layer. As the activation function, Rectified Linear Unit (ReLU) function has been used for the hidden

layers, and the final layer of all architectures has a linear activation layer, as commonly used for regression models.

Out of 1000 pairs of points in the dataset, 200 of them are selected as the test data and 800 for the training phase, after randomly re-sorting the point pairs. Also, before selecting the test and training data splits, the inputs and outputs are both normalized. The normalization process scales and translates each feature individually, i.e., each joint angle in the input and each error parameter in the output, between its minimum and maximum values such that the values are between zero and one. Normalization of input data improves the convergence of the model and guarantees equal importance of all joint angles as input. On the other side, normalizing the output data guarantees that all output features, here error parameters, are contributing equally to the minimization of the loss function.

The network architectures are implemented in Keras (v2.6.0) with TensorFlow backend. The networks are configured to use Adaptive Moment Estimation (ADAM) optimizer for minimizing the loss function, in our case the mean squared error (MSE), through several training epochs. The learning rate of 0.0003 was used in all cases, and as the rest of parameters, the default values of Keras were used.

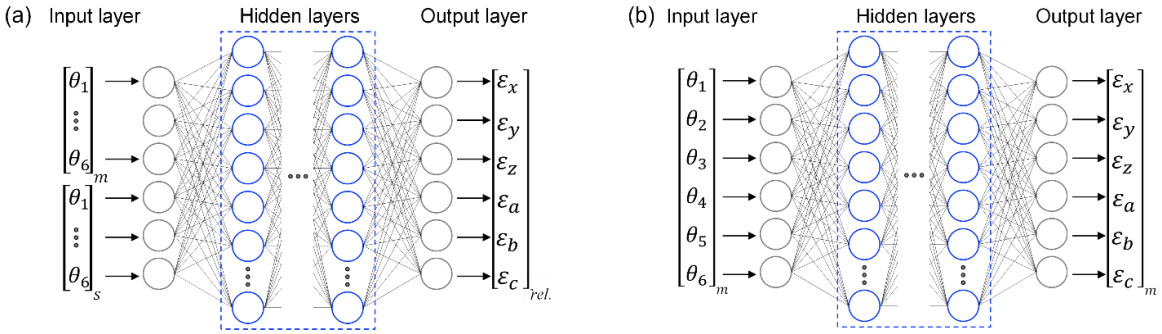

Figure 4.9: Schematics of ANN aechitectures for prediction of (a) master-slave relative error $\boldsymbol{\varepsilon_{rel.}}$ and (b) master absolute error $\boldsymbol{\varepsilon_m}$

4.5.1 Optimal MLP architecture

Due to dealing with a relatively small dataset (1000 measurements), it is crucial to avoid overfitting without having to use a portion of the dataset as validation data during the training phase. Using k-fold cross-validation helps in finding the optimal parameters such that the trained network has the best performance in predicting the unseen test data. In this method, the training data is divided into k number of folds. Then, the training process is performed k times by holding one of the folds as validation set and using the rest of the data for training. Figure 4.10 shows different splits of data during the process of data preparation.

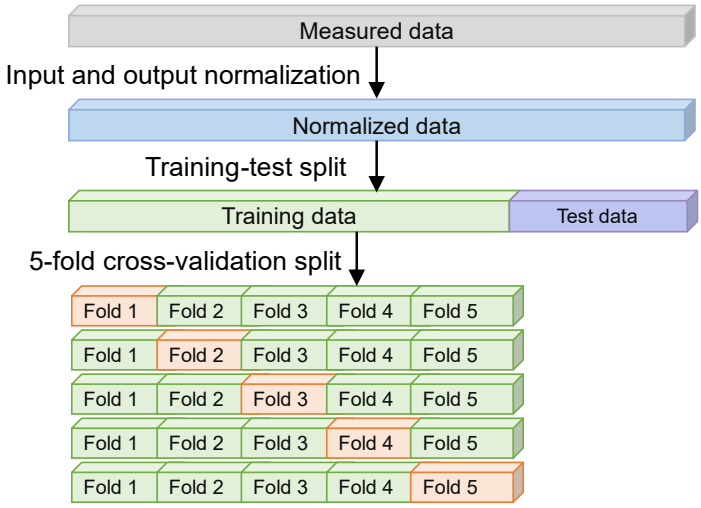

Figure 4.10: Data preparation and splits for training, test and 5-fold cross-validation

Since estimating the performance using k-fold cross-validation can be noisy, we performed repeated k-fold cross-validation with 5 repeats and 5 folds, k=5, and reported the mean performance across all repeats and all folds.

To find the optimal architecture, we studied from shallow fully dense networks with one hidden layer to deep neural networks with up to 4 hidden layers. The number of neurons per layer varies between 32 to 1024. Since by changing the depth and width of the network the convergence speed varies, we implemented an "Early stopping" mechanism [110] to give equal chance to all models to converge. This mechanism stops the training process for each model after reaching the minimum validation loss value and not showing improvement for a pre-defined number of epochs, in our case 100. Figure 4.11 (a) and (b) show the cross-validation results for master-slave relative pose error and master pose error, respectively, for the case of uncalibrated robots. As can be seen, networks with 3

hidden layers each having 256 neurons, (3×256), outperform other architectures and generate the lowest mean loss on the validation folds. It should also be noted that when the number of neurons is relatively low (less than 512), increasing the number of hidden layers allows the model to capture more complex patterns in the data, as shown in Figure 4.11. However, as the number of neurons increases beyond 512, the results indicate that the networks already have sufficient capacity to learn from the data. Adding more hidden layers leads to overfitting, where the model starts memorizing noise in the training data, causing it to perform poorly on unseen data.

The same graphs are shown in Figure 4.12 (a) and (b) after geometric calibration of the robots. In this case, one layer of 512 neurons, (1×512), outperforms other networks and provides the best performance in terms of mean validation loss.

Having the optimal architecture for each case, the same cross-validation process is performed for different batch sizes and epoch numbers. Figure 4.13 (a) and (b) show the performance of (3×256) network in predicting master-slave relative error and master error, respectively, for uncalibrated robots. In both cases, the batch size of 32 is selected for the training process, as the batch sizes higher than 1 are usually recommended to increase the speed and stability of the training process [111]. For the case of geometrically calibrated robots, shown in Figure 4.14 (a) and (b), the batch size of 16 shows a good performance in terms of

validation loss for both cases of master-slave relative error and master error. Therefore, the batch size of 16 has been selected for the (1×512) architecture.

Knowing the batch size, the models are trained for different epoch numbers from 200 to 2000, see Figure 4.13 and Figure 4.14. It is shown that 800 and 600 epochs lead to the lowest mean validation loss before the model starts overfitting for master-slave relative error, shown in Figure 4.13 (a), and master error, shown Figure 4.13 (b), respectively. After the geometric calibration, the (1×512) model, shows the best performance at 600 and 800 epochs for master-slave relative error, shown in Figure 14 (a), and master error, shown in Figure 4.14 (b), respectively.

Finally, knowing the optimal training epoch number, we can train the final models on all training data, without withholding any fold of data as validation, and use the trained model as the final optimal model on the test data.

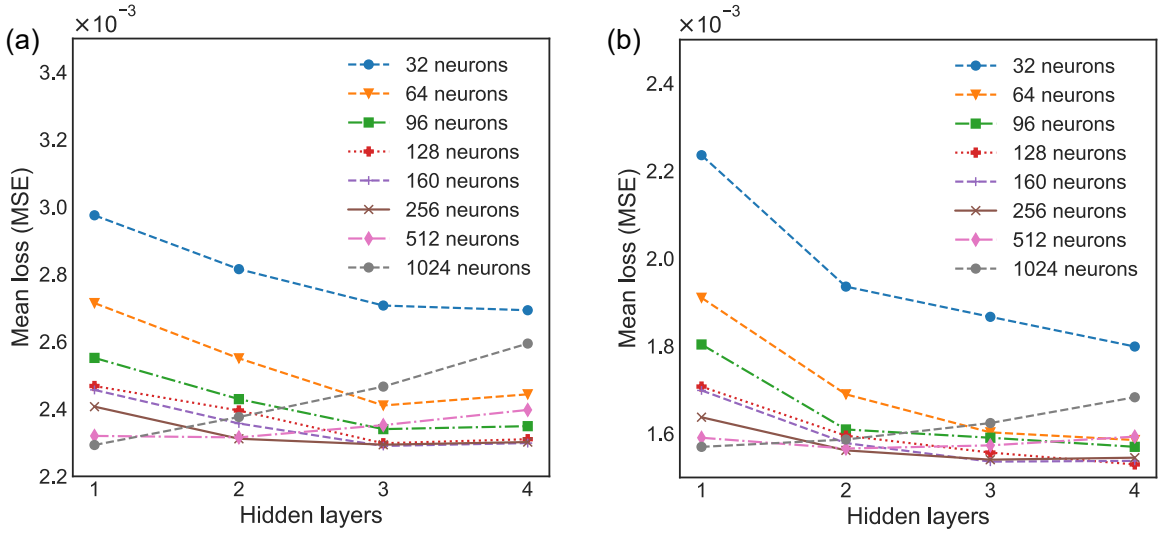

Figure 4.11: 5-fold cross-validation results of (a) master-slave relative pose error, $\boldsymbol{\varepsilon_{rel}}$, and (b) master error, $\boldsymbol{\varepsilon_{m}}$, before geometric calibration subject to different number of layers and neurons

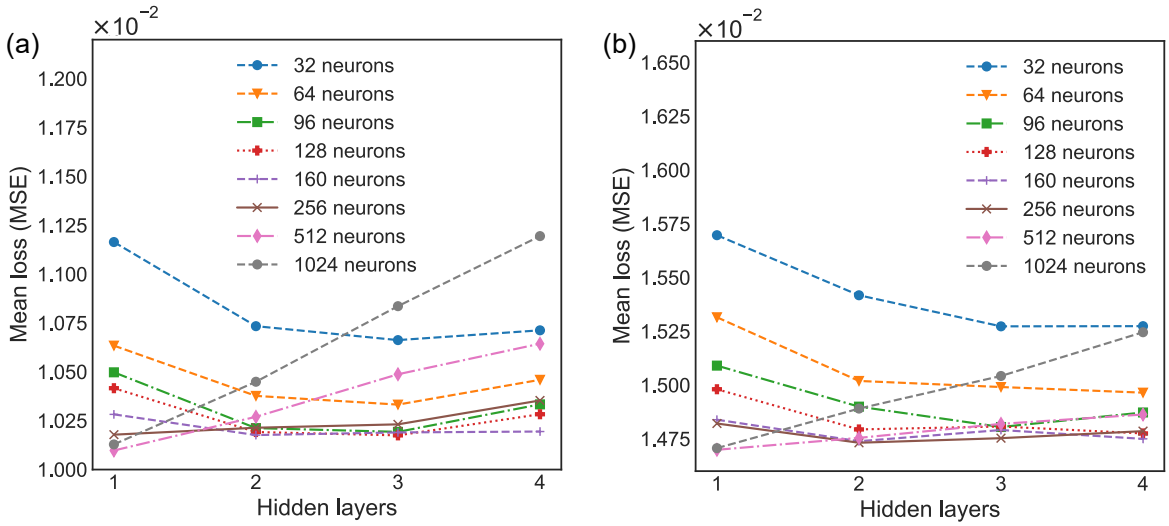

Figure 4.12: 5-fold cross-validation results of (a) master-slave relative pose error, $\boldsymbol{\varepsilon_{rel.}}$, and (b) master error, $\boldsymbol{\varepsilon_m}$, after geometric calibration subject to different number of layers and neurons

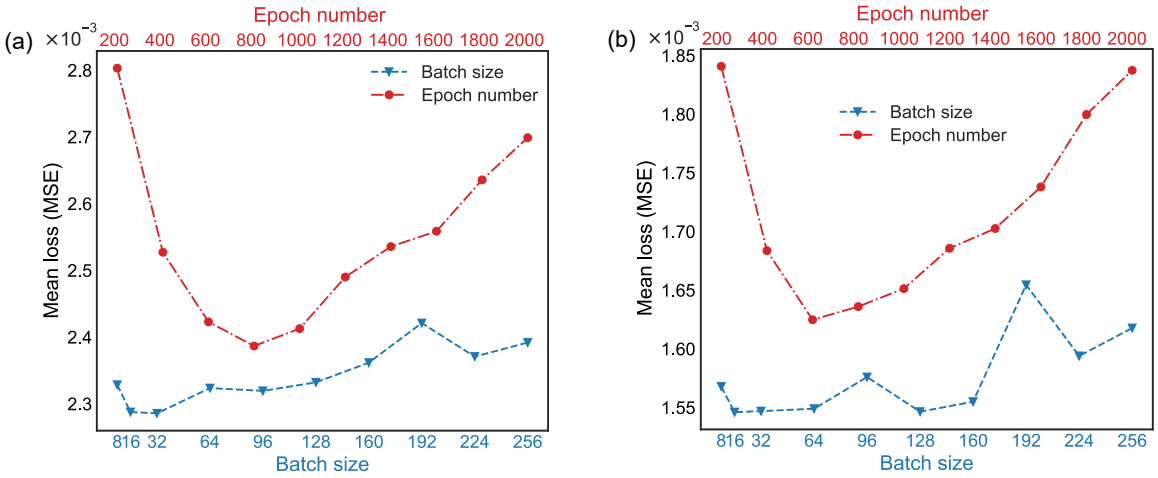

Figure 4.13: 5-fold cross-validation results of (a) master-slave relative pose error, $\boldsymbol{\varepsilon_{rel.}}$, and (b) master error, $\boldsymbol{\varepsilon_m}$, before geometric calibration subject to different batch sizes and epoch numbers

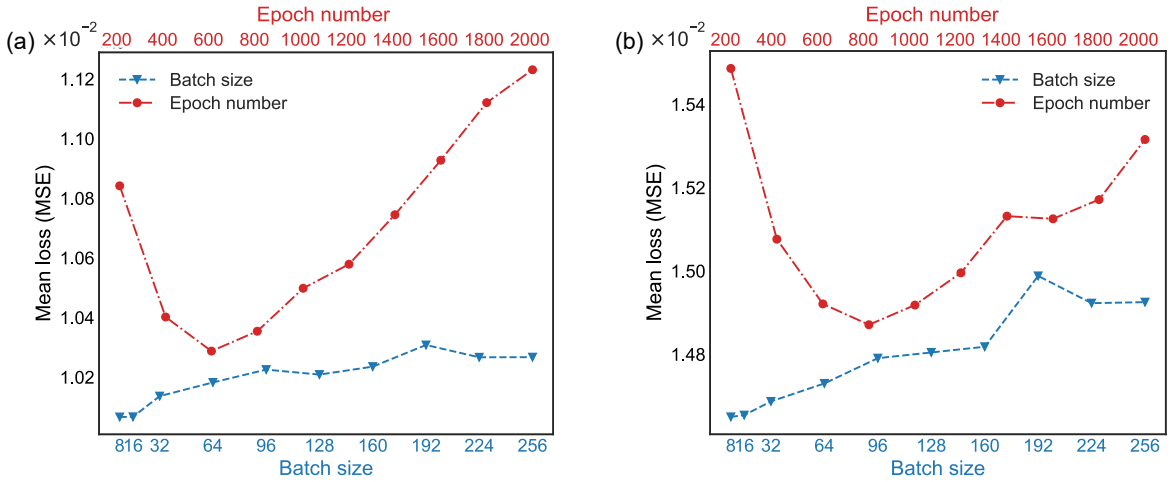

Figure 4.14: 5-fold cross-validation results of (a) master-slave relative pose error, $\boldsymbol{\varepsilon_{rel.}}$, and (b) master error, $\boldsymbol{\varepsilon_{m}}$, after geometric calibration subject to different batch sizes and epoch numbers

4.6 Results and discussion

The test data is used to evaluate the performance of the trained models on unseen data. All of these data points were experimentally measured. As shown in Figure 10, 200 points are kept apart from the data used for cross validation and training processes. The error parameters in terms of position and orientation are predicted in these points and compared with the measurements in both cases of uncalibrated and geometrically calibrated robots.

Figure 4.15 and Figure 4.16 show the calibration effect on the master-slave relative position and orientation errors, respectively. Table 4.1 also lists the numerical values for maximum, mean, and standard deviation (STD) of errors before and after compensation. As can be seen, the Max./Mean values of relative position error were reduced from 2.45/1.04 mm in uncalibrated robots to 0.43/0.185 mm (82% / 82% reduction) after error compensation. Figure 4.15 (a) and Figure

4.16 (a) show that the ANN can further improve the result of geometric calibration by compensating for the non-geometric sources of error. Even after geometric calibration, the ANN model was able to reduce the Max./Mean values of relative position error from 0.70/0.25 mm to 0.37/0.15 mm (47% / 40% reduction). A similar trend can be observed for relative orientation errors. Also, the distribution of relative errors for both uncalibrated and geometrically calibrated robots is shown in Figure 4.15 and Figure 4.16 (b) and (c), which shows a clear reduction in mean and spread of the errors. It can be seen that after geometric calibration of both robots, Figure 4.15 (c) and Figure 4.16 (c), the relative error between the robots can be as large as 0.7 mm and be reduced to almost half by the trained ANN.s

Table 4.1 Maximum, mean, and standard deviation of master-slave relative errors

Master-slave relative error	Position (mm)			Orientation (deg)		
	Max.	Mean	STD	Max.	Mean	STD
Uncalibrated robots						
Before compensation	2.452	1.043	0.417	0.163	0.058	0.034
After compensation	0.431	0.185	0.087	0.069	0.020	0.010
Improvement (%)	82	82	79	58	66	71
Geom. calibrated robots						
Before compensation	0.701	0.348	0.136	0.078	0.048	0.011
After compensation	0.374	0.148	0.068	0.045	0.014	0.008
Improvement (%)	47	57	50	42	71	27

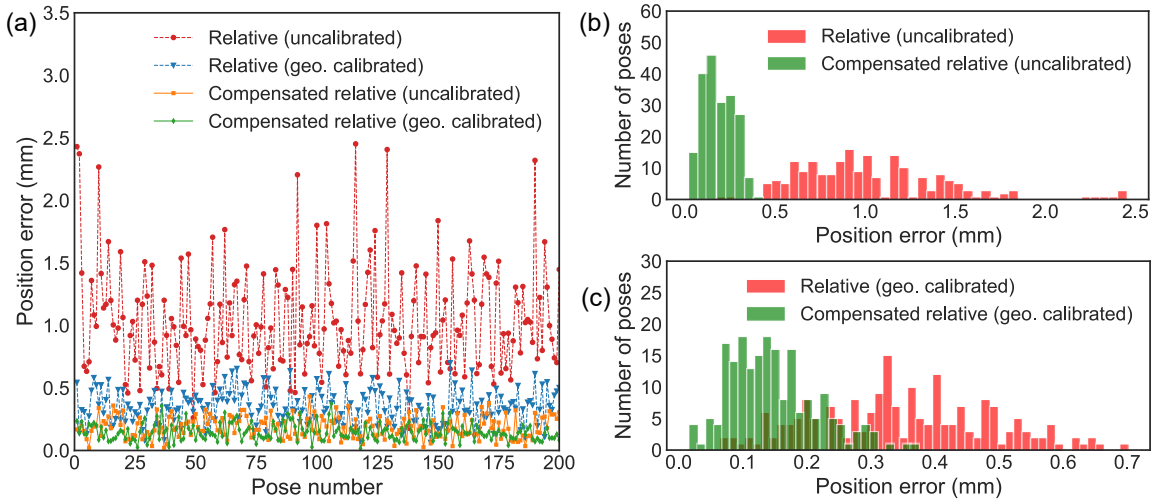

Figure 4.15: Simulation results for master-slave relative position error on test data for (a) uncalibrated and geometrically calibrated cases and their distributions in (b) and (c), respectively

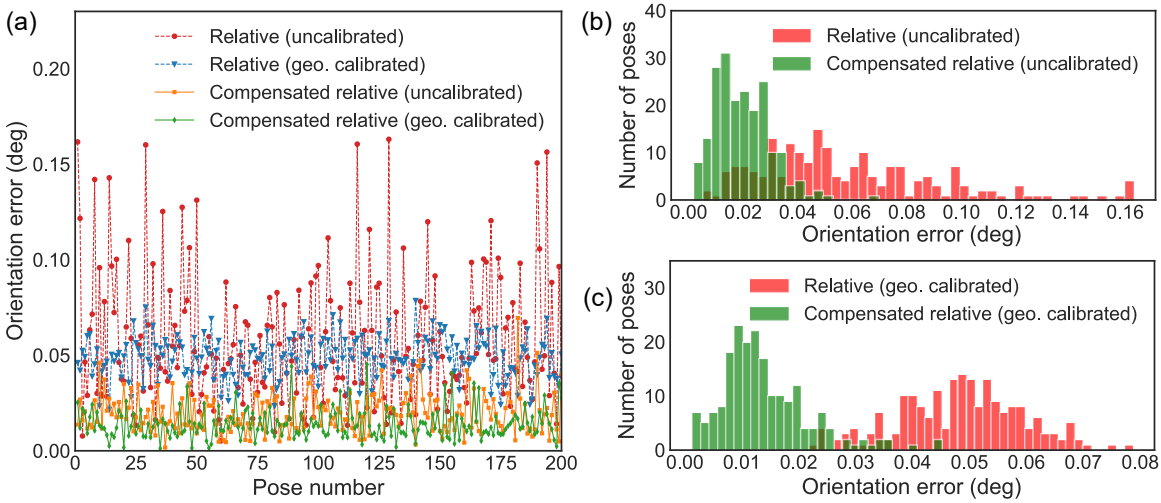

Figure 4.16: Simulation results for master-slave relative orientation error on test data for (a) uncalibrated and geometrically calibrated cases and their distributions in (b) and (c), respectively

For master robot errors, the numerical results of before and after error compensation are listed in Table 4.2. The ANN model was able to successfully reduce the Max./Mean values of master position error from 1.46/1.20 mm in uncalibrated robot to 0.17/0.07 mm, i.e. about 89% / 94% improvement. Figure 4.17 (a) and Figure 4.18 (a) compare the position and orientation errors before and after compensation. For both cases of position and orientation errors in master robot before geometrical calibration, the ANN is able to reduce the errors significantly. Even for the case of geometrically calibrated master robot, the ANN can further compensate the Max./Mean position error values from 0.35/0.16 mm to 0.16/0.07 mm, which is similar to what was achieved for uncalibrated master robot. This indicates that, if designed carefully, the ANN have the capacity to discover both geometric and non-geometric errors of a single robot with no pre-calibration. Figure 4.17 and Figure 4.18 (b) and (c) show the error histograms for uncalibrated and geometrically calibrated master robot, respectively. The massive difference between the mean and spread of the distribution in both cases of position and orientation errors shows the effectiveness of the trained ANN in predicting the master robot errors.

Table 4.2 Maximum, mean, and standard deviation of master errors

Master error	Position (mm)			Orientation (deg)		
	Max.	Mean	STD	Max.	Mean	STD
Uncalibrated robots						
Before compensation	1.460	1.198	0.115	0.131	0.046	0.026
After compensation	0.167	0.067	0.033	0.030	0.009	0.005
Improvement (%)	89	94	71	75	80	81
Geom. calibrated robots						

Before compensation	0.349	0.157	0.061	0.059	0.037	0.010
After compensation	0.159	0.067	0.029	0.020	0.007	0.004
Improvement (%)	54	57	52	66	81	60

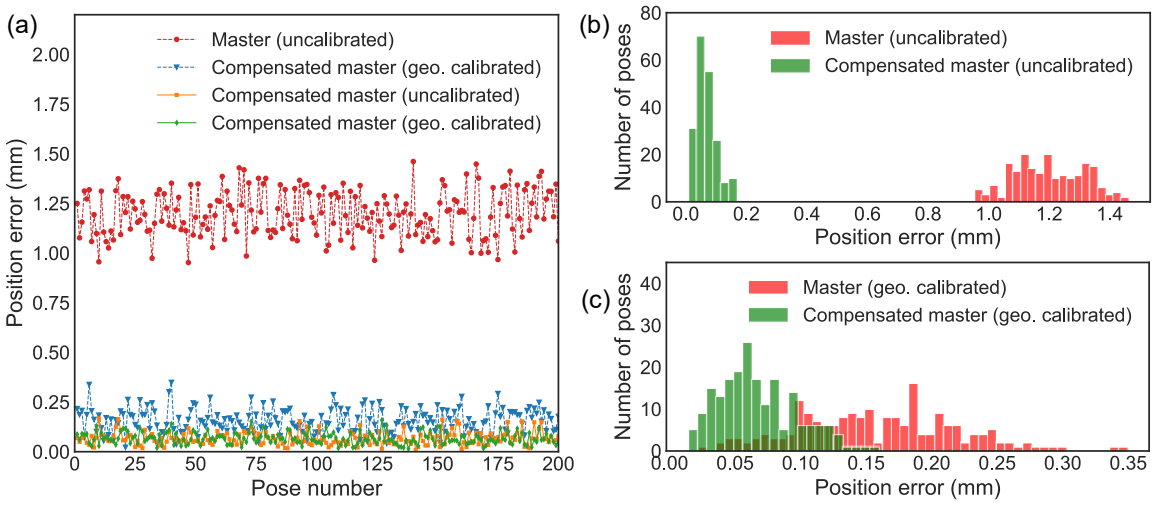

Figure 4.17: Simulation results for master position error on test data for (a) uncalibrated and geometrically calibrated cases and their distributions in (b) and (c), respectively

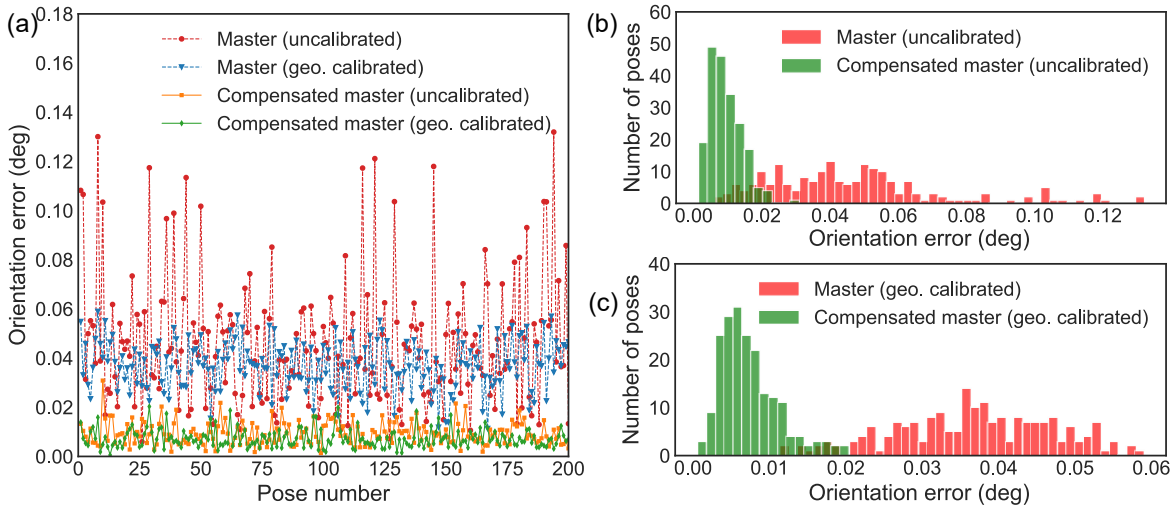

Figure 4.18: Simulation results for master orientation error on test data for (a) uncalibrated and geometrically calibrated cases and their distributions in (b) and (c), respectively

Considering the presented data in Table 4.1 and Table 4.2, it can be seen that without the prior calibration of each robot, the neural network resulted in slightly higher errors after compensation. Since the ANN model compensates for both non-geometric and geometric errors by minimizing the loss function over the given training data, this can suggest that some of these sources of error can cancel each other's effect out in the process of minimizing the loss function. In other words, the complexity of the relationship between the input parameters and output errors cannot be fully recovered given the training data. This suggests the limited capacity of the neural network in recovering the error function using the given number (or resolution) of training data. This is particularly true in the case of master-slave relative error (see Table 4.1) due to its more complex nature, with a higher number of input parameters and contributing errors from both robots. Therefore, it may be beneficial to increase the resolution of measurements for the case of relative master-slave errors or separate these sources of error into two steps when the number or resolution of training data is limited.

4.7 Summary

This paper aims to improve the accuracy of multi-robot systems in performing cooperative tasks. A calibration method was first proposed to directly compensate for the relative errors of the slave flange with respect to the master flange. A neural network was trained to compensate for the relative errors given the joint space of master and slave robots simultaneously. The absolute accuracy of the

cooperative system was then addressed by calibrating the master robot using a secondary neural network. A set of 1000 cooperative robot configurations were measured to form the training and test datasets. The effect of prior geometric calibration was also investigated by performing the same measurements after geometric calibration of master and slave robots. Two neural network models were trained for the compensation of master-slave relative errors and master errors in each case of uncalibrated and geometrically calibrated robots. The optimal parameters of the neural networks were found using repeated 5-fold cross validation method. The simulation results for error compensation show that the trained neural network can reduce the maximum positional relative error from 2.45 mm to 0.43 mm, 82% reduction, in case of robots without prior calibration. A similar trend was observed after the geometric calibration of each robot, where the neural network reduced the maximal relative position error from 0.7 mm to 0.37 mm, 47% reduction. Without prior calibration of each robot, the neural network resulted in slightly higher errors after compensation in the case of master-slave relative errors. However, in the case of the master robot, the difference between compensated errors for uncalibrated and geometrically calibrated robot was negligible. This indicates that the collective geometric and non-geometric errors of cooperative robots together are more difficult to predict in relation with the given inputs, i.e., the combined joint space of both robots. Therefore, in the case of master-slave relative errors, a prior geometric calibration or increasing the resolution of training data can be beneficial to further improve accuracy.

Chapter 5

Vision-Based Inspection Automation

Inspection of drilled holes is a critical step to ensuring the safety and reliability of aircraft structures. This chapter presents the design and implementation of a vision-based framework for the automated inspection of drilled holes using multi-light imaging and deep learning techniques.

5.1 Background

Carbon fiber composites are widely used in the aerospace industry due to their versatility and superior strength-to-weight ratio. During the manufacturing process, hundreds of thousands of holes must be drilled in composite structures for attachments and acoustic purposes [14, 112]. Due to the large stress concentration at hole locations, cracks and damages around drilled holes can quickly lead to part failure and disintegration of the aircraft assembly [6]. As such, it is crucial to

inspect holes for damages such as cracks and delamination after the drilling process.

Anisotropic behavior of composites leads to complex damage mechanisms which demand specialized Non-Destructive Testing (NDT) methods for quality control. Different measurement technologies for assessment of drilling damages can be categorized into three main classes: vision-based [19], ultrasound [113], and X-ray radiography [114]. The main advantage of X-ray and ultrasound methods is their ability to penetrate the workpiece and therefore, obtain information about the inner layers of composites. However, these techniques typically require highly specialized equipment and a controlled environment, and therefore may not be a suitable option for rapid inspection of drilled holes on production lines. Vision-based techniques, on the other hand, offer a fast and cost-effective inspection method which can easily be implemented in a fully automated manner using industrial robots. The outer layers of composites are most susceptible to showing drilling damages particularly at the tool entry (peel-up) and exit (push-out) [14]. Therefore, while vision-based methods cannot inspect inner layers of composites, they can provide a practical and rapid solution for flagging damaged holes directly on production lines.

Vision-based inspection is typically performed using either optical microscopy [115–119] or digital photography [120–125]. Optical microscopes can provide ultra high resolution images which can be used for detailed analysis and modeling of damages in CFRPs and fiberglass composites [115–119]. However, due to their

small field of view, optical microscopes cannot be used efficiently for rapid inspection of drilled holes [126]. Digital photography can provide a practical trade-off between resolution and field of view by selecting a proper camera-lens combination. As such, it has been chosen as the preferred imaging method in this work.

Researchers have used digital cameras along with different lighting techniques to investigate drilling induced damages in composites. Caprino and Tagliaferri [120] measured the delamination area in drilled fiberglass using a strong light source positioned behind the specimen. Stone and Krishnamurthy [121] utilized a CCD camera along with a visual digitizer. Khashaba [122] developed an inexpensive color flatbed scanner for inspection of quasi-transparent fiberglass composites. Seif et al. [123] used a shadow moiré laser based technique combined with a CCD camera to investigate the effect of different cutting conditions on drilling damages. Faraz et al. [124, 125] used a stereomicroscope with a special built-in illumination to measure the delamination area of drilled holes in CFRPs.

Vison-based inspection of carbon fiber composites is particularly challenging due to their dark, textured, and semi-specular surface [14]. Lighting conditions and the imagining angle can significantly affect the visibility and appearance of damages. Therefore, sophisticated image processing techniques are required to reliably detect damages such as cracks, and to distinguish them from the natural background pattern of CFRPs. Caggiano et al. [127] proposed an image analysis

algorithm based on hue saturation value (HSV) and with the aid of a green backlight through the hole. Hrechuk et al. [118] developed a segmentation method using the Delaney triangulation algorithm. However, the challenge with classical image processing techniques is that they require extensive manual calibration of lighting conditions and imaging parameters in order to reliably detect damages in CFRPs. To mitigate this problem, the present research proposes deep neural networks that can learn to detect drilling damages directly from raw captured images.

Deep learning techniques have been successfully implemented in different manufacturing applications for feature recognition, process monitoring, etc. [128–131]. Creating a deep learning model requires the selection of a proper network architecture, a cost function to assess the model performance, and a suitable optimization algorithm. There are mainly three deep learning approaches that can be used for inspection: object detection, regional image classification, and semantic segmentation. Object detection techniques can recognize items and localize them in an image using a rectangular frame. As an example, these techniques have been implemented for the inspection of thermal insulations [132]. The second approach is based on regional image classification. In these methods, the image is divided into smaller areas and each area is evaluated to determine if it contains the objects of interest. Cha et al. [133] used this method for detection of cracks in concrete structures. In their case, they classify each small area of the image using a Convolutional Neural Network (CNN). However, the above-mentioned

approaches cannot provide a pixel-level accuracy (better than 50 um) for calculating the geometrical parameters, e.g., area, perimeter, and length, required for the inspection of fine damages around drilled holes. Therefore, this research uses the third approach, i.e., pixel-level semantic segmentation. In this method, all pixels in the image of a drilled hole are classified into the damage area, hole area, and background texture.

Fully Convolutional Networks (FCNs) [134] are an ideal deep learning architecture for semantic segmentation problems. In FCNs, the fully connected layers of CNNs are no longer present, and therefore FCNs are comprised of only convolutional layers. U-Net is a highly efficient FCN with an encoder-decoder architecture originally designed by Ronneberger et al. [135] for segmentation of biomedical cells. In the encoder path, U-Net enables capturing high-level features through successive convolutional steps followed by pooling layers which gradually reduce the input dimensions. In the decoder path, it avoids information loss of pooling operations through skip connections between encoder and decoder steps. U-Net has been able to provide higher accuracy in segmentation of images with smaller training datasets compared to CNNs [136]. This makes U-Net an ideal option for automatic inspection tasks in industrial environments, where collecting training datasets is a challenge. Therefore, this research uses U-Net as the chosen architecture for the deep learning models.

5.1.1 Overview of the proposed framework and contributions

Figure 5.1 shows a schematic of the proposed inspection system. The developed framework comprises three main modules, which form the contributions of this paper:

i) A multi-light imaging end-effector has been carefully designed to allow capturing images from drilled holes subject to different lighting conditions. Captured images are then fused to provide a higher contrast between damages and the background.

ii) Image processing algorithms are implemented to extract the damage area, hole profile, and crack lines. The extracted regions serve as output labels (masks) for training of the deep learning model.

iii) A deep U-Net model is designed and trained on the masks generated by the image processing module. The trained U-Net is able to autonomously detect the damage area, hole profile, and crack lines from raw images of holes without any explicit image processing.

The developed system has been successfully tested for inspection of drilled holes in CFRPs as presented in the results section.

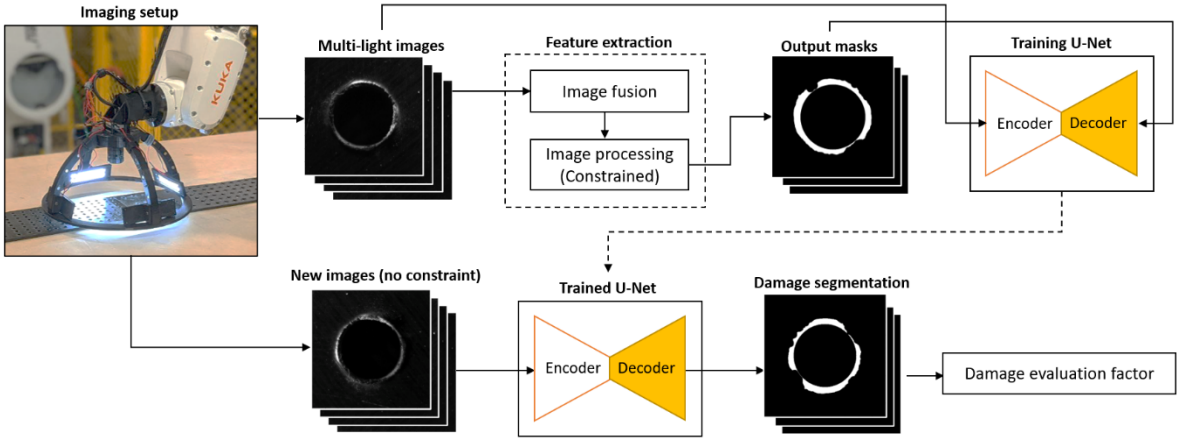

Figure 5.1: A schematic of the proposed method based on U-Net.

5.2 Multi-light setup and image fusion

Multi-light imaging can be defined as capturing sets of digital images from the same viewpoint with varying lighting conditions [137]. In a set of captured images, each pixel coordinate (x, y) is associated with one fixed physical point in the field of view while correlating with different light rays and brightness values. Such sets of images increase the amount of visual information that can be extracted through image processing. This section outlines the developed robotic end-effector for multi-light imaging and the subsequent image fusion steps for enhancing the contrast between the damage area and the background.

5.2.1 Multi-light imaging setup

Figure 5.2 shows a schematic diagram of the imaging and lighting setup. The lighting system comprises four LED lights which are fired sequentially one at a time to illuminate the inspected hole from different directions. The lights are

positioned with the same angle and distance with respect to the surface of the hole. Due to the semi-specular surface of CFRPs, dark-field (low angle) illumination has been used with an angle of 18 degrees from the surface. Low angle lighting can more effectively highlight uneven areas such as damages without intensifying the background patterns of the specimen. A single camera positioned above the hole and perpendicular to the surface captures four images, Img_1 to Img_4, while the hole is sequentially illuminated with lights 1 to 4, respectively. As detailed in Section 5.2.2, pixel-level image fusion is then used to combine the four images into a single image to increase the contrast of the damage area relative to the background.

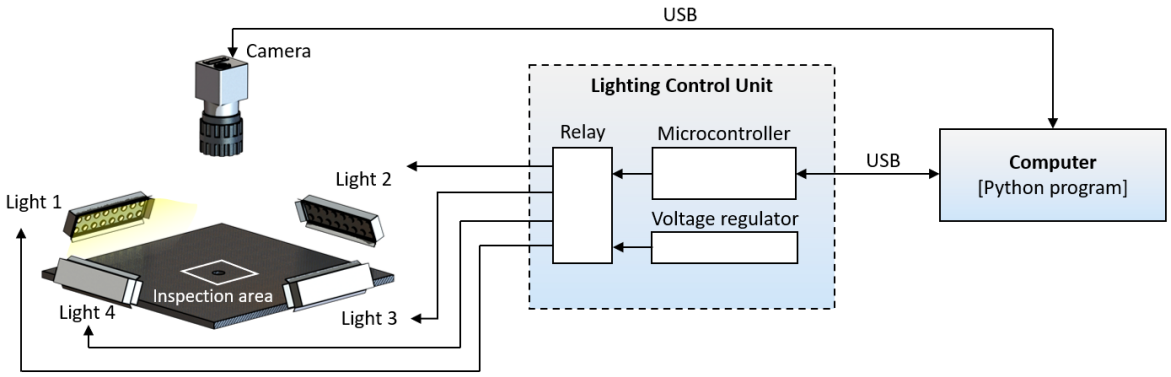

Figure 5.2: Overview of the multi-light imaging system.

Table 5.1 provides the detailed specifications of the imaging system. A Basler acA1440-220uc camera combined with a TECHSPEC C Series 25mm 2/3" Fixed Focal Length lens has been used as the imaging system, providing a resolution of 14 µm at a working distance of 100 mm. Each of the four LED lights (Figure 5.2) contains 30 SMDs of type 3025. To provide high intensity and uniform

illumination across the field of view, the SMDs in each light are packaged in a 2×15 rectangular array with approximate dimensions of 15×75 mm. The camera is controlled via a Python program through Pypylon, the official Python interface for the Basler cameras. The program controls the lighting system through an Arduino Uno microcontroller equipped with a 4-channel relay interface board, which is programmed to switch lights on and off followed by signals from a serial communication port. An adjustable step-down voltage/current regulator is used as the lighting power supply to control the light intensity. The light intensity must be controlled along with the sensor exposure time in a way that the resulting inspection images do not show pixel saturation in the bright damage areas. Pixel saturation causes white areas that do not have texture information, resulting in loss of information and inaccurate measurements.

Table 5.1 Technical specifications of the inspection camera and lens.

Camera	Model	Basler ace acA1440-220um USB 3.0 Monochrome Camera
	Camera Sensor Format	1/2.9"
	Pixels	1,440 × 1,080 (1.6 MP)
	Pixel Size	3.45 × 3.45 µm
Lens	Model	TECHSPEC C Series 25mm 2/3" Fixed Focal Length Lens
	Max. Image Circle	2/3"
	Working Distance (WD)	100 mm - ∞
	Field of View	20 mm (with a 1/3" sensor)
	Focal Length (FL)	25 mm

5.2.2 Image fusion for enhancing contrast of the damage area

Figure 5.3. (a) shows a sample set of four images, Img_1 to Img_4, captured by the multi-light imaging system from a hole with a diameter of 6 mm. Assuming grayscale images, the light intensity received by each pixel from the workpiece surface includes the combined reflection of light from the ambient and the lighting system. Therefore, each of the captured images, Img_i (i=1,2,3,4), can be written as:

$$\mathrm{Img}_i(x,y) = \mathrm{Img}_{\mathrm{Amb}}(x,y) + \mathrm{Img}_{\mathrm{L}_i}(x,y), \tag{5.1}$$

where (x,y) is the pixel coordinates, $\mathrm{Img}_{\mathrm{Amb}}$ is the contribution of ambient light in the formation of Img_i, and $\mathrm{Img}_{\mathrm{L}_i}$ is the added light intensity from the ith light. The ambient part of the images, $\mathrm{Img}_{\mathrm{Amb}}$, can be considered constant when the environment lighting does not vary rapidly. Therefore, the subtraction of images with opposite lighting, i.e., $\mathrm{Img}_i(x,y)$ and $\mathrm{Img}_{i+2}(x,y)$ where i=1,2, can be found as:

$$\mathrm{Img}_{\mathrm{Sub}(i,i+2)} = \mathrm{Img}_i - \mathrm{Img}_{i+2} = \mathrm{Img}_{\mathrm{L}_i} - \mathrm{Img}_{\mathrm{L}_{i+2}}, \tag{5.2}$$

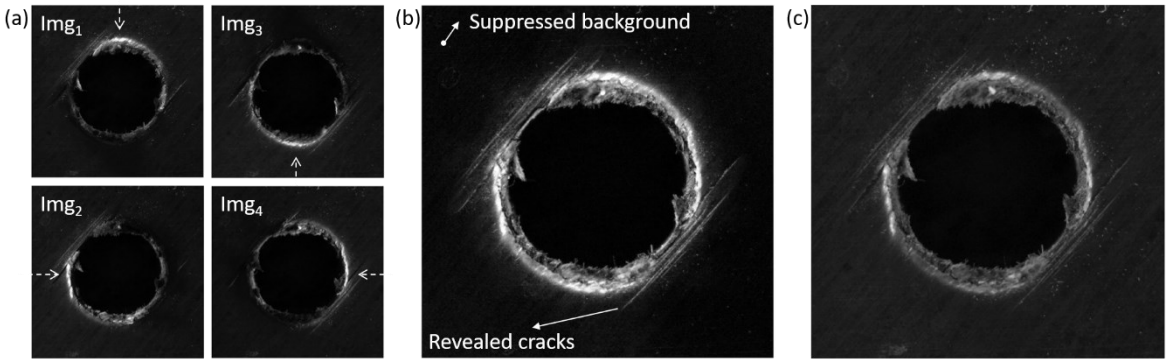

Figure 5.3: (a) Four captured inspection images from the hole (diameter: 6 mm), (b) fused image $\mathrm{Img}_{\mathrm{FD}}$ (proposed) in comparison with (c) image captured with dark-field lighting (conventional).

where $\text{Img}_{\text{Sub}(i,i+2)}$ is the subtraction of Img_{i+2} from Img_i. The resulting $\text{Img}_{\text{Sub}(i,i+2)}$ shows only the difference between the reflections of Light i and Light $i+2$ from the workpiece surface. As illustrated in Figure 5.4, since the lights are positioned on the opposite sides of the hole, each light generates more reflection on the side of the damaged surface where surface normals make smaller angles with the light rays. Since the background texture receives similar lighting from the two sources, subtraction of images, i.e., $\text{Img}_{\text{Sub}(i,i+2)}$, can effectively suppress the background. However, this subtraction may also result in losing the bright pixels (damages) of the $\text{Img}_{\text{L}_{i+2}}$ since negative brightness values turn to zero (black) in the subtracted image. To recover the bright regions of the $\text{Img}_{\text{L}_{i+2}}$, the opposite of this process, i.e. subtraction of Img_{L_i} from $\text{Img}_{\text{L}_{i+2}}$, is also calculated:

$$\text{Img}_{\text{Sub}(i+2,i)} = \text{Img}_{i+2} - \text{Img}_i = \text{Img}_{\text{L}_{i+2}} - \text{Img}_{\text{L}_i} . \tag{5.3}$$

Equations (5.2) and (5.3) applied on the four images in Figure 5.3 (a) result in four new images, i.e. $\text{Img}_{\text{Sub}(1,3)}$, $\text{Img}_{\text{Sub}(3,1)}$, $\text{Img}_{\text{Sub}(2,4)}$, $\text{Img}_{\text{Sub}(4,2)}$. Each pair of these images in one direction, $\text{Img}_{\text{Sub}(i,i+2)}$ and $\text{Img}_{\text{Sub}(i+2,i)}$ (i =1,2), are then combined to form a single image by keeping the maximum pixel values at each pixel coordinate. The resulting image in each of the two directions are then added to each other to form the final "fused image" as:

$$\text{Img}_{\text{FD}} = \sum_{i=1,2} \max\left(\text{Img}_{\text{Sub}(i,i+2)}, \text{Img}_{\text{Sub}(i+2,i)}\right) . \tag{5.4}$$

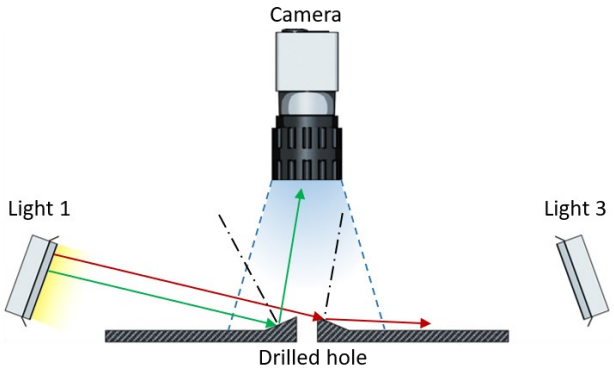

Figure 5.4: Light reflections on the damage surface.

Figure 5.3 (b) demonstrates the result of the proposed image fusion method, Img_{FD}, applied on the four images shown in Figure 5.3 (a). It can be seen clearly that the damage area around the hole has been intensified significantly compared to images obtain through conventional dark-field lighting (Figure 5.3 (c)). As a more quantitative measure, Figure 5.5 compares the gray level values (0-255) of the center column pixels in the proposed fused image (Figure 5.3 (b)) and conventional dark-field lighting (Figure 5.3 (c)). To have a fair comparison, the dark-field lighting intensity is set such that it creates approximately the same gray level in the background regions as the fused image. As can be seen in Figure 5.5, in the case of the fused image, the damage area shows significantly higher gray values indicating increased separation (contrast) between the damages and the background. The proposed image fusion technique results in a highly suppressed background as well although the effectiveness may vary depending on the composite material, fiber lay-up, visibility of fibers, and lighting conditions.

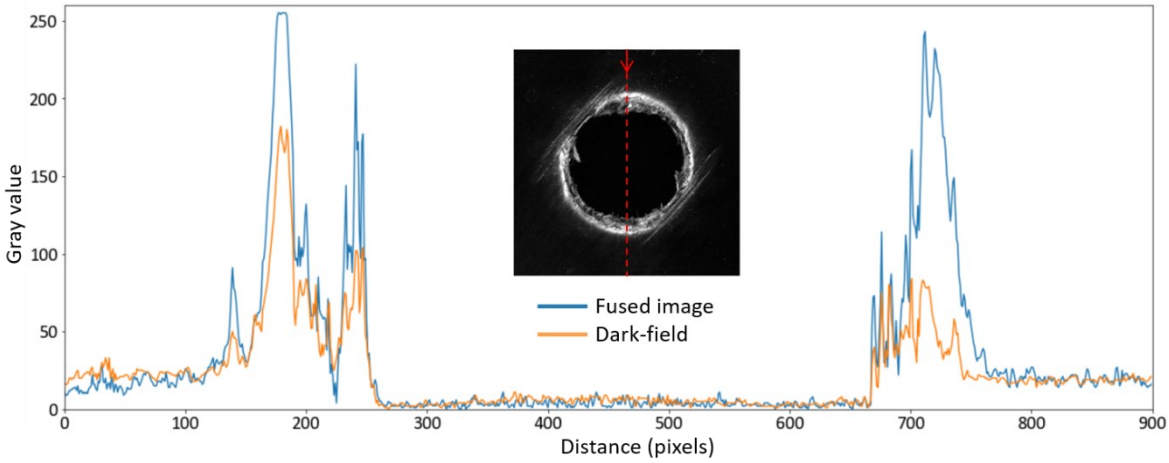

Figure 5.5: Comparison between gray level profile of the center column pixels in the proposed fused image (Figure 5.3 (b)) and conventional dark-field lighting (Figure 5.3 (c)).

5.2.3 Image fusion for extracting the hole profile

In order to segment an image of a hole for inspection, it is crucial to accurately detect the hole profile. Unlike in section 5.2.2 where the goal was to intensify the damage zone, here we need to clearly separate the hole itself from the surrounding area. To facilitate the segmentation of the hole profile, many researchers have used backlight illumination [117, 118, 127]. In this method, a diffused light source is placed behind the hole so that the light passing through the hole can clearly separate the hole geometry from the background. However, in rapid inspection of large aircraft panels on production lines, it may not always be feasible to shine light from behind holes. Therefore, instead of backlight illumination, this study takes advantage of dark-field imaging features provided by the developed multi-light system.

As illustrated schematically in Figure 5.6 (a), most incident rays from low angle lights that strike the inner wall of the hole scatter through the hole and therefore do not reach the camera's sensor. As a result, the hole area appears nearly black in captured images. The light rays reflected by the workpiece surface generate bright pixels in the image. Therefore, maximum separation of the hole area (black) from the surrounding area can be achieved by adding all four multi-light images in Figure 5.3.(a):

$$\mathrm{Img}_{\mathrm{FH}} = \sum_{i=1}^{4} \mathrm{Img}_i, \tag{5.5}$$

where $\mathrm{Img}_{\mathrm{FH}}$ is the fused image that will be used for extracting the hole profile. Adding the four images together combines the reflected light from all directions, resulting in higher brightness of the damage area and workpiece surface. It should be noted that the resulting image from Equation (5.5) can reach pixel values higher than 255 in the damage areas. Since the goal here is to separate the hole area (black) from damage areas (white), the saturated values over 255 are simply set to 255. Figure 6.b shows the fused image for hole profile extraction, $\mathrm{Img}_{\mathrm{FH}}$. The gray level profile (0-255) of the center column pixels is shown in Figure 6.c. As can be seen, the combined reflection of lights from all directions results in near saturation brightness of the damage area and relatively higher gray level in the background texture, thus providing a clear separation between the hole area and the surrounding regions.

The two fused images, Img_{FD} and Img_{FH}, obtained in Sections 5.2.2 and 5.2.3 are used for the extraction of the damage area and hole profile, respectively, as described in the following section.

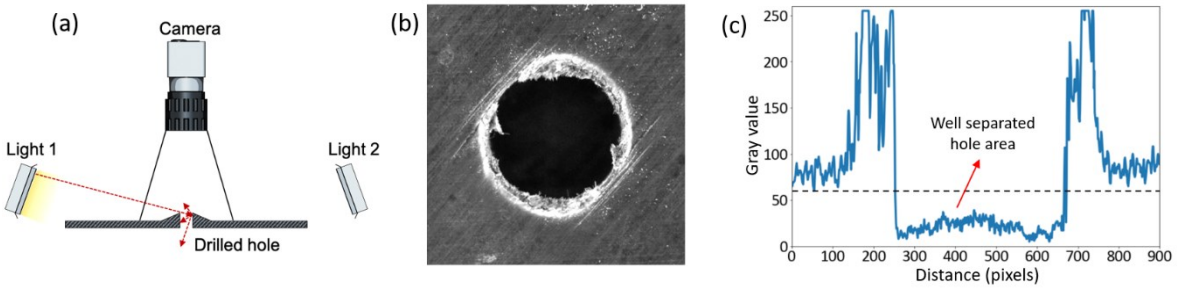

Figure 5.6: (a) Light rays hitting the inner wall of the hole, (b) fused images for extracting the hole profile, Img_{FH}, (c) gray level profile of the center column pixels in Img_{FH}.

5.3 Image processing and segmentation

After multi-light imaging and fusion steps, image processing algorithms are used to extract (segment) the hole profile, damage area, and cracks. The fused images can be segmented more reliably due to the intensified damage zone and the suppressed background. Segmented images are then used as output labels (masks) for training the deep learning model. The following sections describe the proposed image segmentation steps, which have been implemented using the OpenCV library in Python. It should be mentioned that all of these steps are fully automated and can be scaled to any number of images.

5.3.1 Segmentation of hole area

Figure 5.7 shows the procedure for extracting the hole profile. The fused image, Img_{FH}, is first inverted and binarized using thresholding with two values, 0 (black) for the background and 1 (white) for the region of interest, i.e., the hole area. Next, closing morphological transformations [138] are used to fill minor gaps and make the hole area more uniform. The closing transformations are applied as dilation steps followed by erosion steps with the same structuring element (kernel). The remaining small objects and noises are then removed based on a minimum threshold on area size, and the contour around the hole area is extracted. Finally, the hole profile is obtained by enclosing the minimum circle around the hole contour.

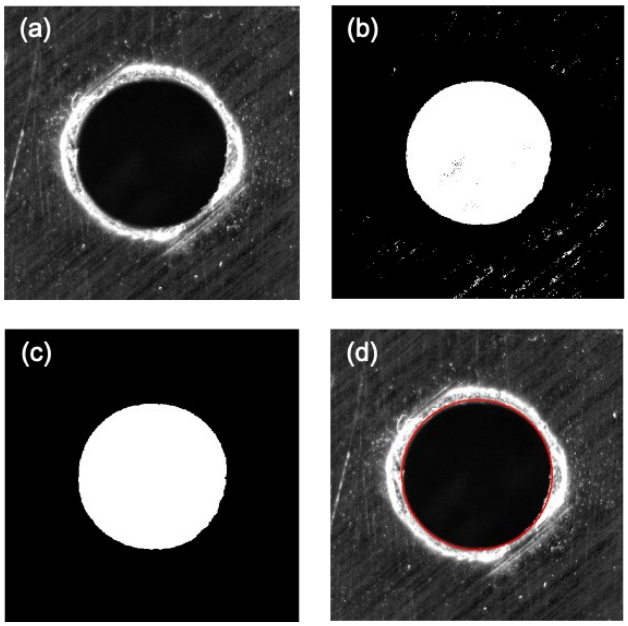

Figure 5.7: (a) Fused image Img_{FH}, (b) inverted and binarized image, (c) binary image after closing and removing noise, (d) fused image with minimum enclosing circle.

5.3.2 Segmentation of damage area

Figure 5.8 shows the steps for segmentation of the damage area. First, Img_{FD} is binarized using thresholding. The hole area obtained from the previous step (Section 5.3.1) is then added to the binary image. The closing transformation is applied, and small objects and noises are removed. The cv2.fillPoly function in OpenCV is used to fill the obtained areas, and the outer contour of the resulting area is marked as the damage envelop (Figure 5.8 (d)).

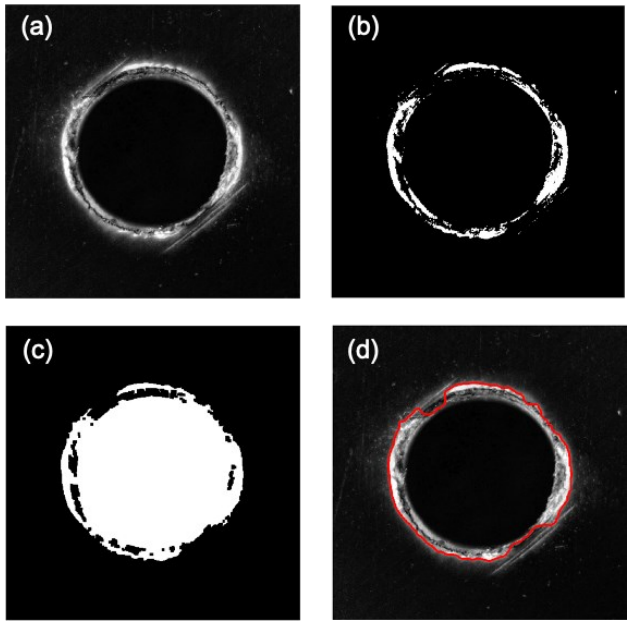

Figure 5.8: (a) Fused image Img_{FD}, (b) binarized image, (c) binary image after adding the hole area and closing filter, (d) fused image with outlined damage area.

5.3.3 Segmentation of cracks

Detecting cracks and distinguishing them from the natural texture of CFRPs is a challenging task and requires more advanced image processing techniques. Figure 5.9 shows the proposed steps for accurate segmentation of cracks in the fused image, Img_{FD}. First, to isolate the cracks, the damage contour obtained in the

previous step (Section 5.3.2) is subtracted from the fused image. Sobel filters are then used to emphasize the crack lines in the image. Sobel filters are 3×3 discrete difference operators commonly used for edge detection and emphasizing lines in images [139]. Since the cracks lie in the fibers' orientation, which is 45° in this case, two successive diagonal Sobel filters are convolved with $\mathrm{Img_{FD}}$. The filter kernels are shown in Equation (5.6).

$$S_1 = \begin{bmatrix} -2 & -1 & 0 \\ -1 & 0 & 1 \\ 0 & 1 & 2 \end{bmatrix}, \quad S_2 = \begin{bmatrix} 2 & 1 & 0 \\ 1 & 0 & -1 \\ 0 & -1 & -2 \end{bmatrix}. \tag{5.6}$$

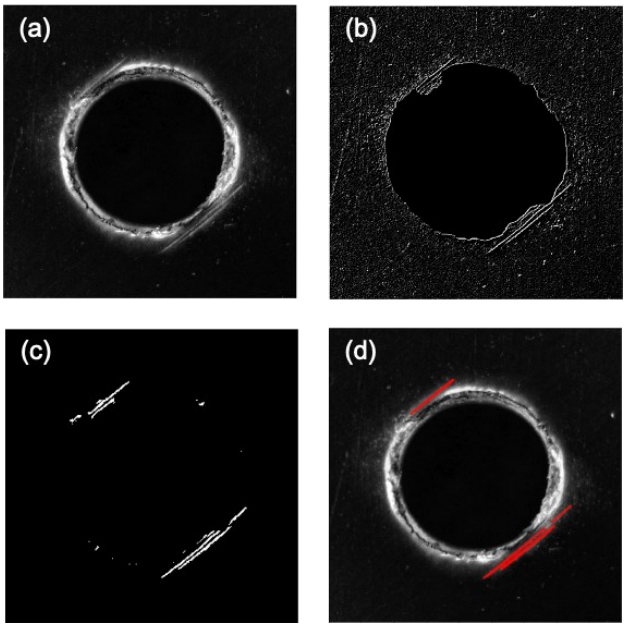

Figure 5.9: (a) Fused image $\mathrm{Img_{FD}}$, (b) image after subtracting the damage contour and applying Sobel filters, (c) binarized image after median blur filter, (d) fused image with segmented cracks (red).

Next, two median blur filters of size 3×3 are applied to reduce the remaining noise from the background texture. With a 3×3 window of median blur, the pixel

in the middle of a window will become the median of all pixels in the window. The image is then binarized and the remaining small objects and noise are removed. To create more connected crack pixels, a diagonal 5×5 dilation kernel shown in Equation (5.7) is applied:

$$D = \begin{bmatrix} 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 1 & 1 \\ 0 & 1 & 1 & 1 & 0 \\ 1 & 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 \end{bmatrix}. \tag{5.7}$$

Application of this dilation kernel connects the crack lines in the 45-degree direction and thickens them to facilitate the line detection process. The remaining edge of the damage area is removed by thickening the damage contour (Figure 5.8 (d)) and subtracting it from the image. Probabilistic Hough Line Transform, cv2.HoughLinesP, is used to detect straight lines from the remaining lines in the image. This algorithm provides the pixel coordinates of the end points of detected cracks. Figure 9.d shows the detected cracks (red) in the fused image, Img_{FD}.

It should be noted that in the case studied here, the cracks lie in the 45° direction, which is aligned with the orientation of fibers. Depending on the fiber lay-up (e.g., 0 or 90°), corresponding Sobel filters must be used. The proposed image processing steps can be extended to detect other types of damage. For example, since the model extracts the hole profile, the uncut fibers and resins within the hole area can be detected using basic image processing methods. Considering that our image fusion method increases the contrast of uneven surfaces compared to the background texture, the proposed image fusion steps are

expected to account for spalling damages as well. Finally, the proposed framework can be extended to other types of laminates such as woven composites. The lighting conditions and image fusion steps may need to be re-adjusted for those cases based on the background texture of the material.

The segmented images obtained as described in Sections 5.3.1-5.3.3 are used as output labels (masks) for training a deep U-Net model. Once trained, the U-Net model can autonomously detect the hole profile, damage area, and cracks in any direction directly from raw images without any image processing needed. The design and training of the deep learning model is presented in the following section.

5.4 Deep learning model for autonomous inspection

To identify damages around the drilled hole using multi-light imaging, a U-Net model is trained to implement image fusion and image processing algorithms in a fully autonomous manner. The following sections explain the architecture and training of the U-Net model.

5.4.1 U-Net architecture

As shown in Figure 5.10, the U-Net model comprises a down-sampling (encoding) path to extract the image context and an up-sampling (decoding) expansive path to enhance the localization resolution of segmentation. The gray layers in the figure represent input and output layers. The black solid arrows indicate the

computational operations inside the neural network. The input layer includes four 320×320 single-channel images, Img_i (i=1,2,3,4), captured by the multi-light imaging system. After the computational operations, the final output is a set of three single-channel images of the same size, i.e., 320×320. A pixel in each output layer can have a score between [0, 1], indicating the probability that the pixel is inside the label of that channel. In this case, the output will be a three-channel distribution of scores for the hole area, damage area, and crack lines.

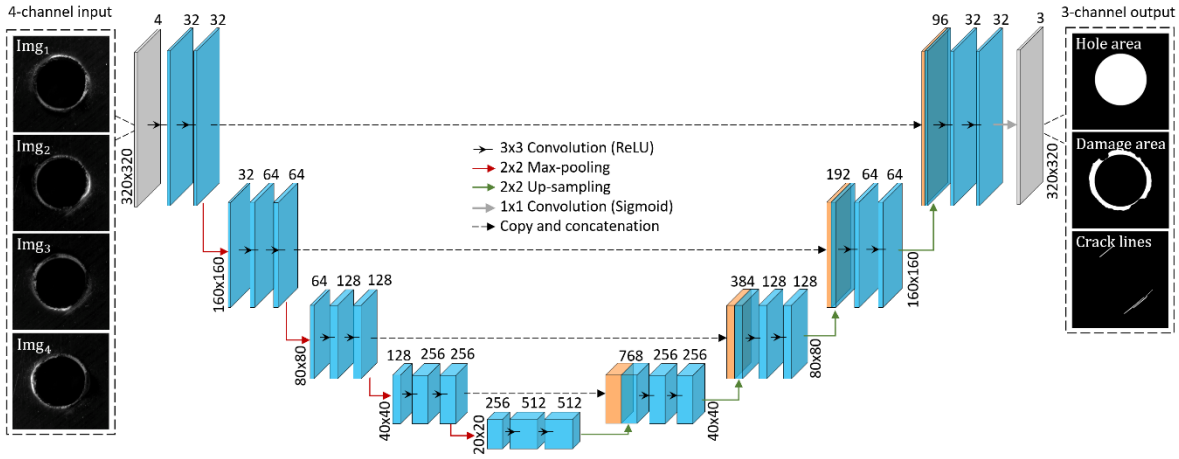

Figure 5.10: Architecture of the developed U-Net neural network for autonomous hole segmentation.

In the down-sampling path, the common architecture of CNNs, including several convolutional layers along with Rectified Linear Unit (ReLU) [140] activations are used. The convolutional layer includes a set of learnable filters (convolution kernels) where each filter captures different features from the image, e.g., low-level features such as textures and edges and high-level features such as cracks and damage regions. The weights in these filters are tuned during the

training process. A kernel size of 3×3 is used for convolution layers and each layer is followed by an activation function. Activation functions are nonlinear and differentiable functions that enable the network to model highly nonlinear systems. ReLU is a commonly used function in neural networks and is defined as:

$$\mathrm{ReLU}(x) = \max(0, x). \tag{5.8}$$

After extracting features at each convolutional layer, the spatial size of the input is reduced by a pooling layer [141]. Application of pooling layer helps with the extraction of higher-level features in subsequent convolutions. Pooling layers also reduce the number of learnable parameters and, therefore, prevent overfitting to the training data. In this research, a commonly used max-pooling layer with a 2×2 kernel is adopted for spatial size reduction. This pooling layer outputs the maximum value of each 2×2 window to the next layer. The encoding path has 4 blocks each comprising two convolutional layers and one max-pooling.

On the right side, the up-sampling layers are used to increase the resolution of the extracted features in a progressive manner. By copying and concatenating the high-level down-sampled layers with their up-sampled counterparts, the accuracy of local information in each class is ensured. After two convolutional layers at each decoding block, an up-sampling layer is used. Up-sampling layers perform the reverse operation of pooling and double the resolution of the feature map. Decoding blocks reconstruct the dimension of the input image step by step to output the segmentation results with the same scale and size.

A 1×1 convolution is implemented in the final layer to generate the outputs. Sigmoid function is used as the activation function of the last layer for binary labelling of each output layer. The sigmoid function is defined as:

$$\sigma(x) = \frac{1}{1 + \exp(-x)}. \tag{5.9}$$

The network outputs a 320×320×3 sigmoid activation map representing the three output masks, i.e., hole area, damage area, and crack lines.

During the training process, a neural network evaluates the difference between the actual and predicted values of the output by calculating a loss function. Higher loss values indicate a larger prediction error and thus must be minimized as the goal of training. Binary cross-entropy is commonly employed as the loss function in pixel-level classification tasks. It calculates the performance of a network with probability output values between [0,1] and is defined as:

$$H = -\frac{1}{n}\sum_{i=1}^{n}(y_i \log \hat{y}_i + (1 - y_i)\log(1 - \hat{y}_i)), \tag{5.10}$$

where n is the total number of labels, y_i is the ground truth of pixel i, and $\hat{y}_i$ is the predicted probability for the corresponding pixel. Class imbalance, which happens when one class occupies a smaller portion of the image, is a common issue that affects the performance of the cross-entropy loss. Since in drilling applications the area of the damage zone and cracks is significantly smaller than the area of the background, network predictions can be more biased toward background pixels [142]. One method to mitigate this problem is to use the Dice coefficient, also

known as the F1 score, as the metric for evaluating the accuracy of image segmentation. Dice factor is defined as:

$$Dice = \frac{2|A \cap B|}{|A| + |B|}, \tag{5.11}$$

where $|A \cap B|$ represents the common pixels between sets A and B. Dice coefficient produces a measure of similarity between the predicted results and the ground truth and is not affected by the ratio of labelled pixels to the background. In discrete data, such as image pixels, Dice coefficient can be expressed as

$$D = \frac{2\sum_{i=1}^{n} y_i \hat{y}_i}{\sum_{i=1}^{n} y_i^2 + \sum_{i=1}^{n} \hat{y}_i^2}. \tag{5.12}$$

While Dice coefficient is commonly used as the loss function in state-of-the-art medical image segmentation tasks [143], it can have convergence (instability) issues in the case of small or rare classes. To take advantage of the stability of the cross-entropy loss function (Equation (5.10)) and the ability of the Dice factor (Equation (5.12)) in dealing with imbalance issues, this research defines the loss function by combining these two losses as:

$$L = H - \log D. \tag{5.13}$$

Minimizing the loss L simultaneously maximizes the probability of correctly classified pixels and the similarity between labeled masks and predictions.

It is worth mentioning that the U-Net model used in this work has been designed with some modifications compared to the original U-Net architecture proposed in [135]. For example, instead of one input image, a set of four multi-light images are given to the network. Also, the size of input images in this work

are 320×320 while the original U-Net model [135] used input images of size 572×572. Zero-padding has been used to avoid shrinking in both down- and up-sampling convolutional layers. Moreover, the Adam optimizer is employed instead of Stochastic Gradient Descent to benefit from its faster convergence rate. Finally, to reduce overfitting, the original U-Net architecture has been modified by adding spatial dropout layers after each max-pooling layer in the down-sampling path, and also before convolutional layers in up-sampling steps.

5.4.2 Training of U-Net for hole inspection

To generate the training dataset, 100 holes with a diameter of 6 mm have been drilled in a unidirectional CFRP plate with 4 mm thickness and $[0/45/90/45]_s$ layup. Different levels of drilling damage have been created by varying the feed rate and using tools with different amounts of wear. The drilling induced damage at the hole entry, i.e., peel-up delamination, has been considered for testing the inspection system. The multi-light end-effector in Figure 5.2 has been used to capture a set of four images for each hole subject to lights 1 to 4. These images are then fused and processed through the steps described in Sections 2 and 3 to generate the output masks for training the U-Net model. Although the output masks are generated automatically, mislabeling in segmentation of cracks was unavoidable in some cases due to drilling dust and uncut fibers. Therefore, about 10% of the crack segmentations have been manually edited to reduce the possible suboptimal performance of trained model in the test phase. In total, 400 images

(100 sets of 4) were captured from 100 holes. 70% of the image sets are used for training, 20% for validation, and the remaining 10% are used for testing.

Data augmentation techniques have been applied on the training images to increase the amount and diversity of the data. This helps prevent overfitting and increases the robustness of the trained model. For data augmentation, the captured images and their corresponding output masks are first cropped with varying frames in terms of positioning and size while ensuring that cropped images always include the damage regions. The images are then rotated, flipped, and resized to 320×320 pixel dimensions. The resulting training dataset after augmentation includes 3,960 sets of 4 images with their corresponding output masks.

The U-Net model has been implemented in Python using Keras neural network library with TensorFlow backend. The training process runs on an RTX 2060 (6GB) GPU. The model is trained with a batch size of 5. The rate of 0.2 is used in all dropouts. The learning rate in Adam optimizer is set to 0.0001. The model is trained for 50 epochs where each epoch goes through the whole training dataset once. Figure 5.11 shows the training and validation losses (Equation (5.13)) and F1 scores (Dice, Equation (5.12)) for each epoch. The latest best model with the lowest validation loss (0.0737) has been saved in the 29th epoch, when the corresponding F1 score of 95.86% is achieved. Figure 5.12 shows the change in the accuracy of the damage segmentation and detected cracks during the training progress for a sample validation hole. As can be seen, in

epochs 1 and 5 the cracks are not visible. By minimizing the loss function, gradually more details become segmented, and by epoch 29, the detected cracks greatly resemble the ground truth segmentations.

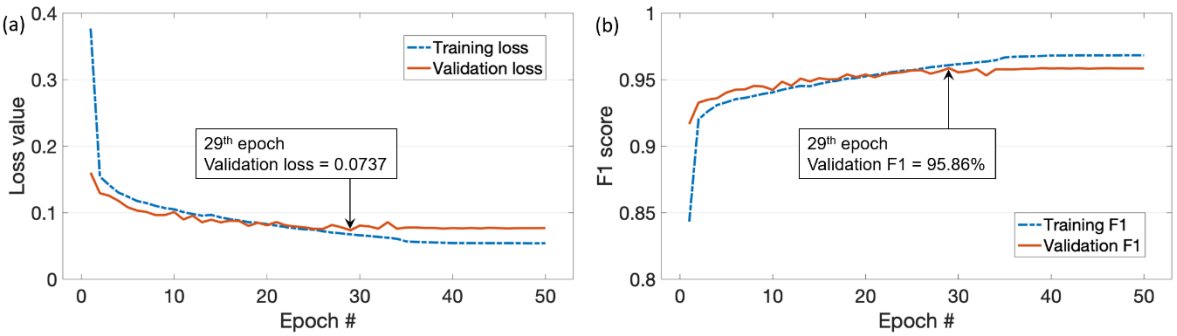

Figure 5.11: Training and validation (a) losses and (b) F1 scores (Dice) for each epoch.

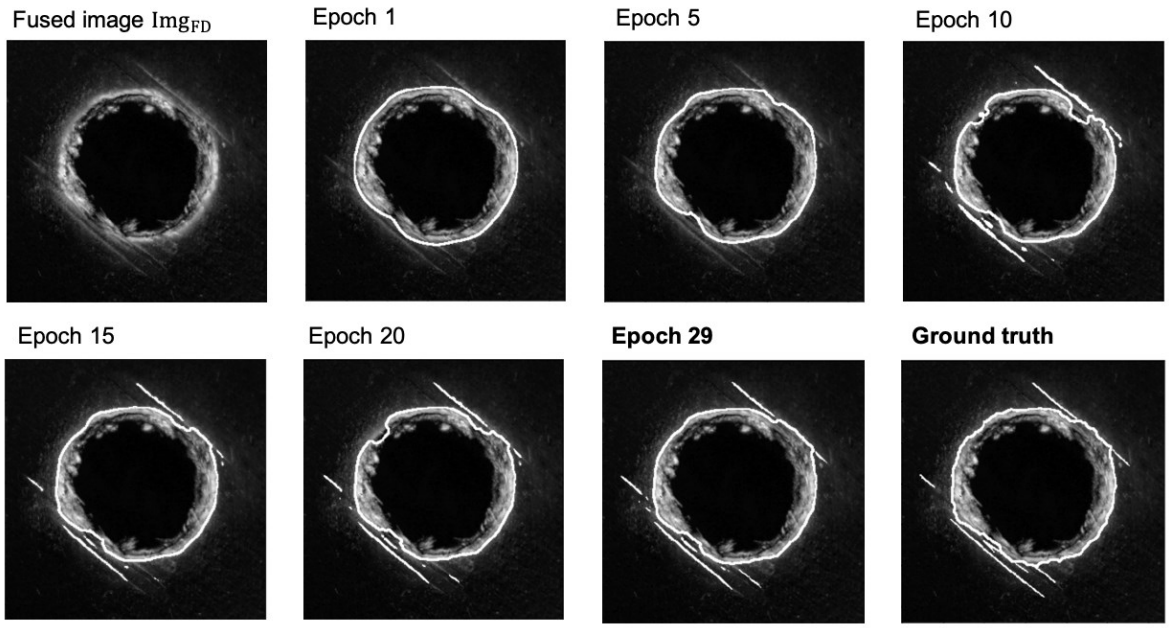

Figure 5.12: Results for the segmentation of a sample validation hole at different epochs.

5.5 Calculation of the damage factor

In the context of drilling composites, using a damage factor serves a crucial role. The primary objective is to systematically measure and portray the extent of damage that happened during the drilling process. These factors provide a quantitative representation, assigning numerical values that help assess the severity or level of induced damages. In the case of vision-based inspection, the damage factor can be used as a method for identifying holes with poor quality that may need to be inspected more thoroughly using more complex inspection methods, such as X-Ray imaging.

To date, several methods have been proposed for quantifying damages in drilling composites. In general, these methods calculate a damage factor which is a number representing the level of induced damages. The most conventionally used delamination factor was proposed by Chen as [144]:

$$F_d = \frac{D_{max}}{D_{nom}}, \tag{5.14}$$

where D_{max} and D_{nom} are the maximum diameter of the delamination zone and the nominal diameter of drilled hole, respectively, as illustrated in Figure 13.a. While the conventional F_d shows satisfactory results for fiberglass composites, accurate assessment of non-uniform damages in drilled CFRPs requires explicit consideration of the area of damages as well. Davim et al. [145] proposed an adjusted delamination factor (F_{da}) that considers both the diameter and area of the delamination zone. F_{da} can be calculated as:

$$F_{da} = F_d + \frac{A_d(F_d^2 - F_d)}{A_{max} - A_{nom}}, \tag{5.15}$$

where A_{max} and A_{nom} are the areas of the circles corresponding to D_{max} and D_{nom}, respectively, and A_d is the area of the damaged zone. Figure 5.13 (a) illustrates the parameters to be measured for the calculation of adjusted delamination factor. Although F_{da} is a more accurate representation of damages compared to the one-dimensional delamination factor, F_d, it can show inaccuracies when dealing with fine cracks [14]. For example, in the two cases shown in Figure 5.13 (a) and (b), F_{da} gives almost the same value since fine cracks have a negligible area. However, the case shown in Figure 5.13 (b) will be more prone to failure due to the cracks, and thus should receive a worse delamination value.

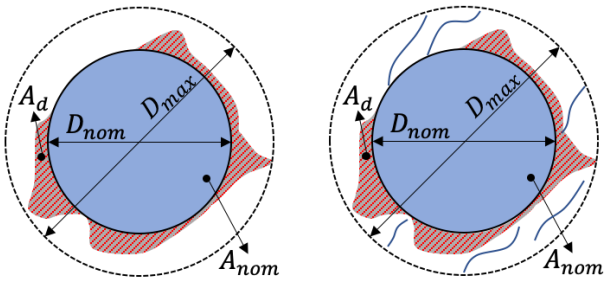

Figure 5.13: Schematic of drilling induced damages: (a) uniform damage area, (b) damage area with fine cracks.

To address the shortcomings of F_{da} in dealing with fine cracks, a damage factor (FEED[1]) was proposed by Babu et al. [146]. This method gives equal weight to the length of cracks and the damage area. Let A_e denote the combined area of the hole, damage zone, and cracks; P_e also represents the total perimeter of these combined areas including individual cracks. FEED can then be calculated as:

$$FEED = \frac{D_{ea} + D_{ep}}{2D_{nom}}, \tag{5.16}$$

where D_{ea} is the equivalent diameter of a circle with the area equal to A_e, and D_{ep} is the equivalent diameter of a circle with the perimeter equal to P_e, i.e.,

$$D_{ea} = \sqrt{4A_e/\pi} \quad , \quad D_{ep} = P_e/\pi. \tag{5.17}$$

Therefore, D_{ea} reflects the area of the damage zone, and D_{ep} accounts for the effect of cracks.

In this research, the FEED factor is used for the evaluation of drilling damages. From the output masks generated by the neural network (Figure 5.10), the hole mask is used for extracting D_{nom}, the damage area and crack masks are used for calculating P_e, and A_e is found from the area of the damage zone envelop. It should be noted that the area of crack lines can practically be ignored compared

[1] In the original article [146] named effective equivalent delamination factor (FEED).

to the area of the damage envelop, but the length of cracks must be considered in calculating P_e.

5.6 Test results

Table 5.2 compares the FEED factors calculated from the trained neural network (labeled with subscript UNet) and image processing results (labeled with subscript IP) for 10 test drilled holes which were not seen by the model during the training process. As can be seen, the results obtained from the trained model show very good agreement with those of image processing with the maximum discrepancy of 5.4%. Figure 5.14 shows the comparison between the identified damage segmentations and parameters D_{ea}, D_{ep} and D_{nom} used in the calculation of FEED factor in four of the test holes (2, 4, 7 and 9). Minor discrepancies between the obtained FEED factors is due to the difference between the calculated D_{ep}, which is very sensitive to the detected length of cracks. Nevertheless, it can be seen that the trained U-Net can provide segmented damage envelops and crack lines very similar to those obtained from image processing. In terms of computation time, the results from the U-Net model were obtained on average in 0.08 seconds using the RTX 2060 GPU. Similar results obtained through the image processing steps take around 0.90 seconds using an i5-8500B CPU. It should be mentioned again that the trained U-Net model is able to segment damages and cracks directly from the four 'raw' multi-light images without any image processing required. In other words, the U-Net model has practically learned to perform image fusion,

processing, and segmentation automatically by going through the training dataset which was generated originally by an extensive image processing routine. The U-Net model will significantly enhance the computational efficiency of the proposed inspection system and will allow rapid real-time detection of damages on production lines.

Table 5.2 Comparison between delamination factors from image processing (IP) and semantic segmentation (U-Net).

Hole #	$FEED_{IP}$	$FEED_{UNet}$	% err.
1	1.45	1.52	4.79
2	1.57	1.61	2.29
3	1.74	1.72	1.05
4	1.91	2.00	4.60
5	2.07	2.03	1.97
6	2.24	2.19	1.93
7	2.24	2.33	3.84
8	2.29	2.30	0.36
9	2.25	2.37	5.42
10	2.35	2.29	2.47

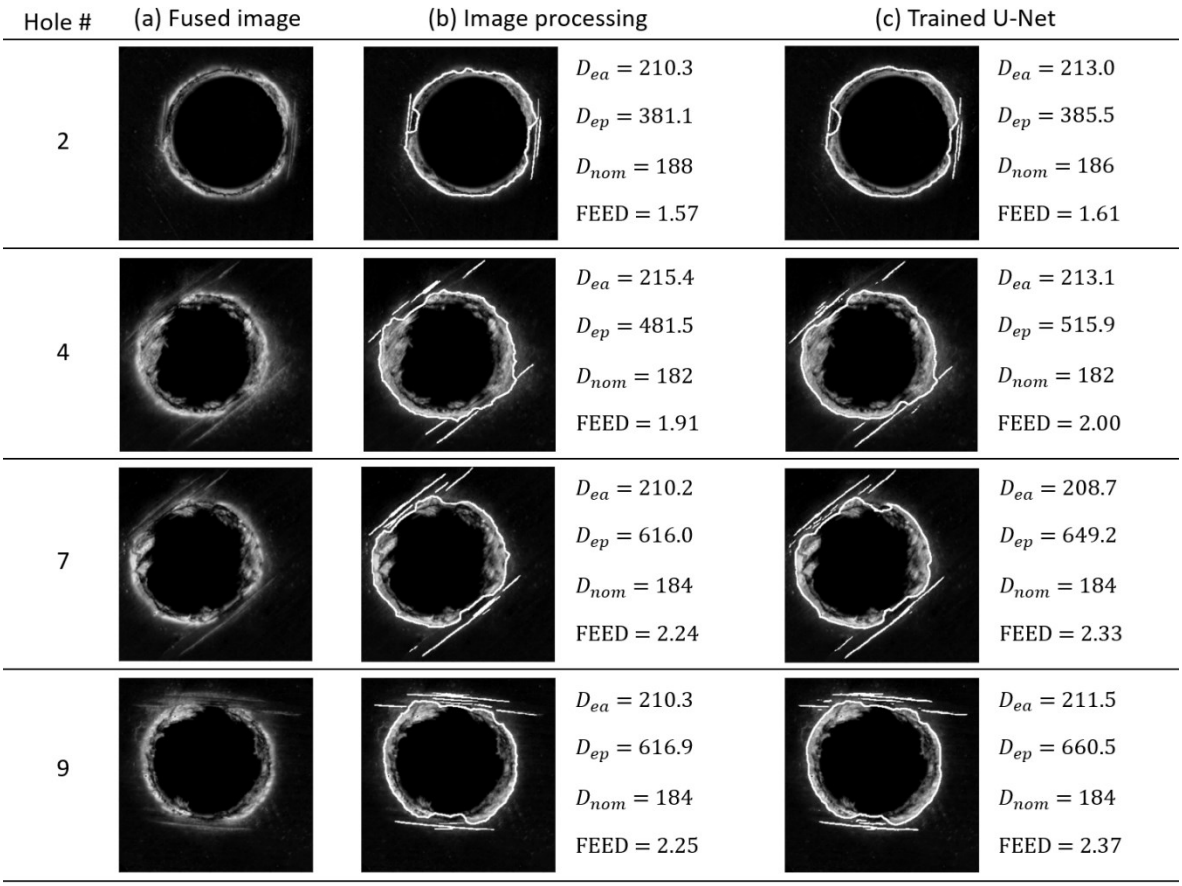

Figure 5.14: (a)Fused images of test holes, (b) segmented damages obtained from image processing, and (c) fused image with prediction outlines. All dimensions are in pixels.

5.7 Summary

This research presents a vision-based system for autonomous inspection of drilled holes in CFRP composites. A multi-light robotic end-effector is designed to sequentially illuminate the inspected hole from four different directions to ensure uniform extraction of defects around the hole. A pixel-level image fusion process is proposed to suppress the background texture of the material and highlight the damages. An extensive image processing routine has been developed to reliably

segment the hole profile, damage area, and crack lines. The segmented images serve as labeled output (masks) for training a deep learning model. A fully convolutional neural network with the U-Net architecture has been developed to learn the image processing and fusion procedures. The trained U-Net model can detect the hole profile, damages, and cracks directly from four 'raw' multi-light images without the need for any post-processing of captured images. Segmented damages and cracks are used to calculate the delamination factor for inspected holes. Delamination factors obtained from the U-Net model match very well with those obtained from the image processing routine, showing a maximum discrepancy of 5.4%. The proposed multi-light system along with the trained deep learning model can be used for rapid inspection of drilled holes in CFRPs directly on aerospace production lines.

Chapter 6

Conclusion

This thesis focuses on the development and application of vision-based techniques to enhance the accuracy and autonomy of robotic drilling and inspection processes in aerospace composite manufacturing. The motivation arises from the prevalent practices of manual labor and intricate robot programming associated with conventional methodologies in aerospace drilling. The ultimate objective is to transition towards fully autonomous robotic drilling systems empowered by sensory capabilities, promising increased efficiency, reliability, and precision.

6.1 Summary of completed work

Two fundamental objectives were pursued in this research:

1. Accuracy Enhancement of Robotic Drilling:

Addressing the inadequacies of off-the-shelf industrial robots in meeting aerospace drilling accuracy requirements, a vision-based measurement solution is proposed. The system's core aim is to accurately measure the spatial coordinates of the robot's drilling spindle. Calibration of the vision system is executed within the robot's workspace using a laser tracker. By affixing circular targets to the drilling spindle, they are tracked through the calibrated vision system. Various target localization methods are thoroughly evaluated, focusing on their performance in robot motion tracking. The assessment includes both distance and pose measurements, which are subsequently compared to the measurements of a laser tracker. The integration of online robot error correction, using the relative robot pose with respect to the part fixtures, is implemented and analyzed. The research extends to the correction of relative errors in dual robot systems employed in manufacturing of large aerospace structures.

2. Autonomous Inspection of Drilled Holes:

Vision-based inspection systems stand as a promising solution for efficient evaluation of drilled holes in aerospace composites. However, the intricate nature of the task, worsened by the characteristics of CFRPs, demands innovative approaches. An autonomous system is proposed for the detection and segmentation of damage and cracks around drilled holes in CFRP composites. The solution is composed of three modules: an automated multi-light imaging end-effector to illuminate inspected holes, image processing steps for automated segmentation of hole features, and a deep FCN based on the U-Net architecture

for pixel-wise semantic segmentation of hole images. The proposed system is tested, showcasing its ability to accurately detect hole profiles, damage areas, and cracks.

These contributions hold substantial implications for the field of robotic manufacturing, bridging gaps and mitigating limitations present in the current aerospace industry practices. By enhancing drilling accuracy and enabling autonomous inspection, the presented vision-based frameworks improve the efficiency and safety of aerospace composite production. Through careful investigations, innovative methodologies, and the integration of novel solutions, this research advances the accuracy and autonomy of robotic drilling processes within aerospace manufacturing.

6.2 Future research directions

Building upon the advancements achieved in this thesis, there are several promising avenues for future research that could further enhance the field of vision-based techniques for autonomous robotic drilling and inspection in aerospace manufacturing.

1. Integration of Multi-Sensory Feedback Mechanisms:

While the developed vision-based solution demonstrates impressive accuracy improvements, incorporating multi-sensory feedback mechanisms could lead to even more adaptive and responsive systems. Investigating the integration of sensor fusion techniques, such as incorporating force or torque feedback, can

enhance the system's ability to dynamically adjust drilling parameters based on the material's characteristics, leading to optimized drilling processes.

2. *Human-Robot Collaboration:*

As the field moves towards more collaborative human-robot workspaces, research could focus on developing hybrid systems where human operators and robots work together seamlessly. Designing vision-based interfaces that allow operators to provide high-level guidance to robots during complex tasks, while the robots use vision systems for precise execution, could lead to enhanced productivity and safety.

3. *Enhanced Deep Learning Architectures:*

While the proposed U-Net architecture is effective in the detection of drilling-induced damage, exploring more advanced and novel deep learning models and architectures could lead to even higher accuracy and robustness. Investigating the incorporation of attention mechanisms, generative adversarsial networks (GANs), or transfer learning approaches tailored to specific aerospace composite characteristics could yield further improvements in defect detection.

Investigating the interpretability of neural networks represents a potential avenue for advancing this research by enhancing our capacity to comprehend the decision-making mechanisms of trained models in a human-understandable manner. Despite neural networks often being labeled as "black boxes" due to their complex nature, there exist various elements contributing to their interpretability, making them not entirely inscrutable. Layer-wise Relevance Propagation (LRP)

stands out as a technique specifically crafted to attribute the neural network's predictions back to its input features. This method can be valuable in making engineering design decisions based on the significance of inputs to the model. Additionally, the incorporation of attention mechanisms explicitly highlights the segments of the input sequence that receive focus during predictions. This imparts a degree of interpretability by revealing where the model directs its attention, facilitating a more nuanced understanding of the model's inner workings.

By pursuing these directions, the research community can further refine and expand the capabilities of vision-based techniques, ensuring their relevance and impact in the ever-evolving landscape of aerospace manufacturing automation.

Bibliography

1. Maghami, S.A., Rezaeepazhand, J., Yousefsani, S.A.: Effect of corner bluntness on energy absorbing capability of non-circular metallic tubes subjected to axial impact. Int. J. Eng. Trans. B Appl. 26, 875–884 (2013). https://doi.org/10.5829/idosi.ije.2013.26.08b.09
2. Yang, Y., Boom, R., Irion, B., van Heerden, D.J., Kuiper, P., de Wit, H.: Recycling of composite materials. Chem. Eng. Process. Process Intensif. 51, 53–68 (2012). https://doi.org/10.1016/j.cep.2011.09.007
3. Minnetyan, L., Chamis, C.C., Gotsis, P.K.: Damage progression in bolted composites. J. Thermoplast. Compos. Mater. 11, 231–248 (1998). https://doi.org/10.1177/089270579801100303
4. Maghami, S.A., Tahani, M.: Thermal bending analysis of doubly curved composite laminated shell panels with general boundary conditions and laminations. J. Therm. Stress. 38, 250–270 (2015). https://doi.org/10.1080/01495739.2014.985568
5. Türk, D.A., Kussmaul, R., Zogg, M., Klahn, C., Leutenecker-Twelsiek, B., Meboldt, M.: Composites part production with additive manufacturing technologies. Procedia CIRP. 66, 306–311 (2017). https://doi.org/10.1016/j.procir.2017.03.359
6. Grutta, J.T., Miskioglu, I., Charoenphan, S., Vable, M.: Strength of bolted joints in composites under concentrated moment. J. Compos. Mater. 34, 1242–1262 (2000). https://doi.org/10.1106/L5A3-WFQ3-N7B0-EQET
7. Devlieg, R., Szallay, T.: Applied accurate robotic drilling for aircraft fuselage. In: SAE 2010 Aerospace Manufacturing and Automated Fastening Conference & Exhibition. pp. 180–186. SAE International (2010)
8. Hefele, J., Brenner, C.: Robot pose correction using photogrammetric tracking. In: Proc. SPIE 4189, Machine Vision and Three-Dimensional Imaging Systems for Inspection and Metrology. pp. 170–178 (2001)
9. Pérez, L., Rodríguez, Í., Rodríguez, N., Usamentiaga, R., García, D.F.: Robot guidance using machine vision techniques in industrial environments: A comparative review. Sensors. 16, (2016). https://doi.org/10.3390/s16030335

10. Newman, M., Monsarrat, B., Fortin, Y., Côté, G., Hajzargarbashi, A., Khoshdarregi, M.: Towards a new real-time metrology guidance method for robotized machining of aerostructures robust against cutting fluids and debris. Int. J. Adv. Manuf. Technol. 126, 2647–2663 (2023). https://doi.org/10.1007/s00170-023-11242-2

11. Clarke, T., Wang, X.: The control of a robot end-effector using photogrammetry. Int. Arch. Photogramm. Remote Sens. XXXIII, 137–142 (2000)

12. Van Duin, S., Crothers, P.J., Steele, P., Newberry, J.: A comparison between indoor GPS versus laser tracking metrology for robotic drilling. SAE Tech. Pap. 115, 1016–1022 (2006). https://doi.org/10.4271/2006-01-3177

13. Mineo, C., Pierce, S.G., Wright, B., Cooper, I., Nicholson, P.I.: PAUT inspection of complex-shaped composite materials through six DOFs robotic manipulators. Insight - Non-Destructive Test. Cond. Monit. 57, 161–166 (2015). https://doi.org/10.1784/insi.2014.57.3.161

14. Geng, D., Liu, Y., Shao, Z., Lu, Z., Cai, J., Li, X., Jiang, X., Zhang, D.: Delamination formation, evaluation and suppression during drilling of composite laminates: A review. Compos. Struct. 216, 168–186 (2019). https://doi.org/10.1016/j.compstruct.2019.02.099

15. Labudzki, R., Legutko, S., Raos, P.: The essence and applications of machine vision. Tech. Gaz. 21, 903–909 (2014)

16. Maghami, A., Khoshdarregi, M.: Target Localization Approaches for High-Precision Optical Tracking in Robotic Applications. J. Intell. Robot. Syst.

17. Maghami, A., Khoshdarregi, M.: Online Robot Pose Correction Using an Optical Tracking Solution. Int. J. Robot. Autom. (2023)

18. Maghami, A., Imbert, A., Côté, G., Monsarrat, B., Birglen, L., Khoshdarregi, M.: Calibration of Multi-Robot Cooperative Systems Using Deep Neural Networks. J. Intell. Robot. Syst. 107, 55 (2023). https://doi.org/10.1007/s10846-023-01867-6

19. Hernandez, A., Maghami, A., Khoshdarregi, M.: A Machine Vision Framework for Autonomous Inspection of Drilled Holes in CFRP Panels. 2020 6th Int. Conf. Control. Autom. Robot. ICCAR 2020. 669–675 (2020). https://doi.org/10.1109/ICCAR49639.2020.9108000

20. Maghami, A., Salehi, M., Khoshdarregi, M.: Automated vision-based inspection of drilled CFRP composites using multi-light imaging and deep learning. CIRP J. Manuf. Sci. Technol. 35, 441–453 (2021).

https://doi.org/10.1016/j.cirpj.2021.07.015

21. Boochs, F., Schutze, R., Simon, C., Marzani, F., Wirth, H., Meier, J.: Increasing the accuracy of untaught robot positions by means of a multi-camera system. In: 2010 International Conference on Indoor Positioning and Indoor Navigation. pp. 1–9. IEEE (2010)
22. Barnfather, J.D., Goodfellow, M.J., Abram, T.: Development and testing of an error compensation algorithm for photogrammetry assisted robotic machining. Measurement. 94, 561–577 (2016). https://doi.org/10.1016/j.measurement.2016.08.032
23. Liu, B., Zhang, F., Qu, X.: A Method for Improving the Pose Accuracy of a Robot Manipulator Based on Multi-Sensor Combined Measurement and Data Fusion. Sensors. 15, 7933–7952 (2015). https://doi.org/10.3390/s150407933
24. Gharaaty, S., Shu, T., Joubair, A., Xie, W.F., Bonev, I.A.: Online pose correction of an industrial robot using an optical coordinate measure machine system. Int. J. Adv. Robot. Syst. 15, 1–16 (2018). https://doi.org/10.1177/1729881418787915
25. Li, J., Berglund, J., Auris, F., Hanna, A., Vallhagen, J., Akesson, K.: Evaluation of Photogrammetry for Use in Industrial Production Systems. In: 2018 IEEE 14th International Conference on Automation Science and Engineering (CASE). pp. 414–420. IEEE (2018)
26. Shen, Y., Zhang, X., Cheng, W., Zhu, L.: Quasi-eccentricity error modeling and compensation in vision metrology. Meas. Sci. Technol. 29, 045006 (2018). https://doi.org/10.1088/1361-6501/aaa5e7
27. Wang, L., Zhang, L., Yu, Z., Chen, F., Si, X., He, D.: Precision circular target location in vision coordinate measurement system. In: Wang, A., Zhang, Y., and Ishii, Y. (eds.) Advanced Materials and Devices for Sensing and Imaging III. pp. 90–97 (2007)
28. Schütze, R., Raab, C., Boochs, F., Wirth, H., Meier, J.: OPTOPOSE - A multi-camera system for fast and precise determination of position and orientation for moving effector. In: 9th Conference on Optical 3D Measurement Techniques. pp. 115–124. , Vienna, Austria (2009)
29. Huang, G., Huang, G., Zhou, Y., Dou, S., Wang, W., Hu, X.: Retro-reflective target with high accuracy for industrial photogrammetry. Meas. Sci. Technol. 34, 015001 (2022). https://doi.org/10.1088/1361-6501/ac9921
30. Nubiola, A., Slamani, M., Joubair, A., Bonev, I.A.: Comparison of two calibration methods for a small industrial robot based on an optical CMM

and a laser tracker. Robotica. 32, 447–466 (2014). https://doi.org/10.1017/S0263574713000714

31. Roig, A.B., Espinosa, J., Perez, J., Ferrer, B., Mas, D.: Three-dimensional planar object tracking with sub-pixel accuracy. Optik (Stuttg). 126, 2684–2689 (2015). https://doi.org/10.1016/j.ijleo.2015.06.070
32. Canny, J.: A Computational Approach to Edge Detection. IEEE Trans. Pattern Anal. Mach. Intell. PAMI-8, 679–698 (1986). https://doi.org/10.1109/TPAMI.1986.4767851
33. Satriya, T., Wibirama, S., Ardiyanto, I.: Robust pupil tracking algorithm based on ellipse fitting. In: 2016 International Symposium on Electronics and Smart Devices (ISESD). pp. 253–257. IEEE (2016)
34. Mai, F., Hung, Y.S., Zhong, H., Sze, W.F.: A hierarchical approach for fast and robust ellipse extraction. Pattern Recognit. 41, 2512–2524 (2008). https://doi.org/10.1016/j.patcog.2008.01.027
35. Rao, B., Zhou, Y., Nie, Y.: Detection and Tracking of Weak Exoatmospheric Target with Elliptical Hough Transform. Remote Sens. 14, 491 (2022). https://doi.org/10.3390/rs14030491
36. Lu, W., Tan, J.: Detection of incomplete ellipse in images with strong noise by iterative randomized Hough transform (IRHT). Pattern Recognit. 41, 1268–1279 (2008). https://doi.org/10.1016/j.patcog.2007.09.006
37. Ahn, S.J., Rauh, W., Warnecke, H.-J.: Least-squares orthogonal distances fitting of circle, sphere, ellipse, hyperbola, and parabola. Pattern Recognit. 34, 2283–2303 (2001). https://doi.org/10.1016/S0031-3203(00)00152-7
38. Sturm, P., Gargallo, P.: Conic fitting using the geometric distance. In: Computer Vision – ACCV 2007. pp. 784–795. Springer Berlin Heidelberg, Berlin, Heidelberg (2007)
39. Fitzgibbon, A., Pilu, M., Fisher, R.B.: Direct least square fitting of ellipses. IEEE Trans. Pattern Anal. Mach. Intell. 21, 476–480 (1999). https://doi.org/10.1109/34.765658
40. Heikkila, J.: Moment and curvature preserving technique for accurate ellipse boundary detection. In: Proceedings. Fourteenth International Conference on Pattern Recognition (Cat. No.98EX170). pp. 734–737. IEEE Comput. Soc (2002)
41. Förstner, W., Gülch, E.: A Fast Operator for Detection and Precise Location of Distinct Points, Corners and Centres of Circular Features,

(1987)

42. Ouellet, J.-N., Hébert, P.: Precise ellipse estimation without contour point extraction. Mach. Vis. Appl. 21, 59–67 (2009). https://doi.org/10.1007/s00138-008-0141-3

43. Pătrăucean, V., Gurdjos, P., von Gioi, R.G.: A Parameterless Line Segment and Elliptical Arc Detector with Enhanced Ellipse Fitting. In: Eccv 2012. pp. 572–585 (2012)

44. Datta, A., Kim, J.-S., Kanade, T.: Accurate camera calibration using iterative refinement of control points. In: 2009 IEEE 12th International Conference on Computer Vision Workshops, ICCV Workshops. pp. 1201–1208. IEEE (2009)

45. Park, F.C., Martin, B.J.: Robot sensor calibration: solving AX=XB on the Euclidean group. IEEE Trans. Robot. Autom. 10, 717–721 (1994). https://doi.org/10.1109/70.326576

46. Arthur, J., Khoshdarregi, M.: Pose optimization and path improvement in robotic drilling through minimization of joint reversals. Adv. Robot. 36, 1076–1086 (2022). https://doi.org/10.1080/01691864.2022.2125828

47. Saha, P.K.: Aerospace Manufacturing Processes. CRC Press (2016)

48. Gao, Y., Wu, D., Nan, C., Chen, K.: Normal direction measurement in robotic drilling and precision calculation. Int. J. Adv. Manuf. Technol. 76, 1311–1318 (2015). https://doi.org/10.1007/s00170-014-6320-7

49. Gong, M., Yuan, P., Wang, T., Yu, L., Xing, H., Huang, W.: A novel method of surface-normal measurement in robotic drilling for aircraft fuselage using three laser range sensors. In: 2012 IEEE/ASME International Conference on Advanced Intelligent Mechatronics (AIM). pp. 450–455. IEEE (2012)

50. Yuan, P., Wang, Q., Shi, Z., Wang, T., Wang, C., Chen, D., Shen, L.: A micro-adjusting attitude mechanism for autonomous drilling robot end-effector. Sci. China Inf. Sci. 57, 1–12 (2014). https://doi.org/10.1007/s11432-014-5190-9

51. Chen, D., Yuan, P., Wang, T., Wang, Q., Wang, C., Wang, F.: Intelligent surface normal measurement method of end effector for the aeronautical drilling robot. In: 2014 International Conference on Multisensor Fusion and Information Integration for Intelligent Systems (MFI). pp. 1–6. IEEE (2014)

52. Wang, Z., Qin, X., Bai, J., Tan, X., Li, J.: Design and Implementation of Multifunctional Automatic Drilling End Effector. IOP Conf. Ser. Mater.

Sci. Eng. 187, 012032 (2017). https://doi.org/10.1088/1757-899X/187/1/012032

53. Chen, D., Yuan, P., Wang, T., Cai, Y., Tang, H.: A Normal Sensor Calibration Method Based on an Extended Kalman Filter for Robotic Drilling. Sensors. 18, 3485 (2018). https://doi.org/10.3390/s18103485

54. Han, L., Yu, L., Zhu, X.: A Novel Method for Pose and Position Calibration of Laser Displacement Sensors. Sensors. 23, 1762 (2023). https://doi.org/10.3390/s23041762

55. Hexagon Metrology: Leica Absolute Tracker AT960 Brochure

56. Wang, Z., Keogh, P.: Experimental Evaluation of Leica T-Mac Angular Accuracy. In: The Journal of the CMSC. pp. 28–32. CMSC 2018 (2018)

57. Liu, S., Liu, C., Liu, Z., Xie, Y., Xu, J., Chen, K.: Laser tracker-based control for peg-in-hole assembly robot. In: The 4th Annual IEEE International Conference on Cyber Technology in Automation, Control and Intelligent. pp. 569–573. IEEE (2014)

58. Diaz Posada, J.R., Schneider, U., Pidan, S., Geravand, M., Stelzer, P., Verl, A.: High accurate robotic drilling with external sensor and compliance model-based compensation. In: 2016 IEEE International Conference on Robotics and Automation (ICRA). pp. 3901–3907. IEEE (2016)

59. Droll, S.: Real Time Path Correction of Industrial Robots with Direct End-Effector Feedback from a Laser Tracker. SAE Int. J. Aerosp. 7, 222–228 (2014). https://doi.org/10.4271/2014-01-2248

60. Moeller, C., Schmidt, H.C., Koch, P., Boehlmann, C., Kothe, S., Wollnack, J., Hintze, W.: Real Time Pose Control of an Industrial Robotic System for Machining of Large Scale Components in Aerospace Industry Using Laser Tracker System. SAE Int. J. Aerosp. 10, 100–108 (2017). https://doi.org/10.4271/2017-01-2165

61. Chen, D., Yuan, P., Wang, T., Cai, Y., Xue, L.: A Compensation Method for Enhancing Aviation Drilling Robot Accuracy Based on Co-Kriging. Int. J. Precis. Eng. Manuf. 19, 1133–1142 (2018). https://doi.org/10.1007/s12541-018-0134-8

62. Fernandez, S.R., Olabi, A., Gibaru, O.: On-line Accurate 3D Positioning Solution for Robotic Large-Scale Assembly Using a Vision System and a 6Dof Tracking Unit. In: 2018 IEEE 3rd Advanced Information Technology, Electronic and Automation Control Conference (IAEAC). pp. 682–688. IEEE (2018)

63. Wang, Z., Zhang, R., Keogh, P.: Real-Time Laser Tracker Compensation of Robotic Drilling and Machining. J. Manuf. Mater. Process. 4, 79 (2020). https://doi.org/10.3390/jmmp4030079

64. Cvitanic, T., Melkote, S., Balakirsky, S.: Improved state estimation of a robot end-effector using laser tracker and inertial sensor fusion. CIRP J. Manuf. Sci. Technol. 38, 51–61 (2022). https://doi.org/10.1016/j.cirpj.2022.03.011

65. Schneider, U., Posada, D.J.R., Drust, M., Verl, A.: Position control of an industrial robot using an optical measurement system for machining purposes. In: 11th International Conference on Manufacturing Resarch (ICMR2013). pp. 307–312. , Cranfield University, United Kingdom (2013)

66. Chen, J., Yang, C., Hofschulte, J., Jiang, W., Zhang, C.: A robust optical/inertial data fusion system for motion tracking of the robot manipulator. J. Zhejiang Univ. Sci. C. 15, 574–583 (2014). https://doi.org/10.1631/jzus.C1300302

67. Gharaaty, S., Shu, T., Joubair, A., Xie, W.F., Bonev, I.A.: Online pose correction of an industrial robot using an optical coordinate measure machine system. Int. J. Adv. Robot. Syst. 15, 1–16 (2018). https://doi.org/10.1177/1729881418787915

68. Zhu, W., Mei, B., Yan, G., Ke, Y.: Measurement error analysis and accuracy enhancement of 2D vision system for robotic drilling. Robot. Comput. Integr. Manuf. 30, 160–171 (2014). https://doi.org/10.1016/j.rcim.2013.09.014

69. Frommknecht, A., Kuehnle, J., Effenberger, I., Pidan, S.: Multi-sensor measurement system for robotic drilling. Robot. Comput. Integr. Manuf. 47, 4–10 (2017). https://doi.org/10.1016/j.rcim.2017.01.002

70. Mei, B., Zhu, W.: Accurate positioning of a drilling and riveting cell for aircraft assembly. Robot. Comput. Integr. Manuf. 69, 102112 (2021). https://doi.org/10.1016/j.rcim.2020.102112

71. Yuanwei Liu, Zhenyun Shi, Yuan, P., Chen, D., Minqing Lin, Zhijun Li: A visual positioning and measurement system for robotic drilling. In: 2016 IEEE 14th International Workshop on Advanced Motion Control (AMC). pp. 461–466. IEEE (2016)

72. Zhan, Q., Wang, X.: Hand–eye calibration and positioning for a robot drilling system. Int. J. Adv. Manuf. Technol. 61, 691–701 (2012). https://doi.org/10.1007/s00170-011-3741-4

73. Filonov, I.V., Zmeu, K.V., Notkin, B.S., Baranchugov, I.A.: Laser Measuring System for Tool and Surface Relative Positioning When

Robotic Processing. In: 2020 International Multi-Conference on Industrial Engineering and Modern Technologies (FarEastCon). pp. 1–5. IEEE (2020)

74. Mei, B., Zhu, W., Yan, G., Ke, Y.: A new elliptic contour extraction method for reference hole detection in robotic drilling. Pattern Anal. Appl. 18, 695–712 (2015). https://doi.org/10.1007/s10044-014-0394-6

75. Lu, C., Xia, S., Shao, M., Fu, Y.: Arc-Support Line Segments Revisited: An Efficient High-Quality Ellipse Detection. IEEE Trans. Image Process. 29, 768–781 (2020). https://doi.org/10.1109/TIP.2019.2934352

76. Lou, H., Lu, M., Cui, H., Jiang, T., Tian, W., Huang, Y.: High-precision location for occluded reference hole based on robust extraction algorithm. Meas. Sci. Technol. 32, (2021). https://doi.org/10.1088/1361-6501/abb165

77. Yu, L., Bi, Q., Ji, Y., Fan, Y., Huang, N., Wang, Y.: Vision based in-process inspection for countersink in automated drilling and riveting. Precis. Eng. 58, 35–46 (2019). https://doi.org/10.1016/j.precisioneng.2019.05.002

78. Yang, H., Jiang, P., Wang, F.: Multi-View-Based Pose Estimation and Its Applications on Intelligent Manufacturing. Sensors. 20, 1–17 (2020). https://doi.org/10.3390/s20185072

79. Jiang, T., Cui, H., Cheng, X., Tian, W.: A Measurement Method for Robot Peg-in-Hole Prealignment Based on Combined Two-Level Visual Sensors. IEEE Trans. Instrum. Meas. 70, 1–12 (2021). https://doi.org/10.1109/TIM.2020.3026802

80. Ayyad, A., Halwani, M., Swart, D., Muthusamy, R., Almaskari, F., Zweiri, Y.: Neuromorphic vision based control for the precise positioning of robotic drilling systems. Robot. Comput. Integr. Manuf. 79, 102419 (2023). https://doi.org/10.1016/j.rcim.2022.102419

81. Arun, K.S., Huang, T.S., Blostein, S.D.: Least-Squares Fitting of Two 3-D Point Sets. IEEE Trans. Pattern Anal. Mach. Intell. PAMI-9, 698–700 (1987). https://doi.org/10.1109/TPAMI.1987.4767965

82. Shah, M.: Solving the Robot-World/Hand-Eye Calibration Problem Using the Kronecker Product. J. Mech. Robot. 5, (2013). https://doi.org/10.1115/1.4024473

83. Chou, J.C.K., Kamel, M.: Finding the Position and Orientation of a Sensor on a Robot Manipulator Using Quaternions. Int. J. Rob. Res. 10, 240–254 (1991). https://doi.org/10.1177/027836499101000305

84. Chen, H.H.: A screw motion approach to uniqueness analysis of head-eye

geometry. In: Proceedings. 1991 IEEE Computer Society Conference on Computer Vision and Pattern Recognition. pp. 145–151. IEEE Comput. Sco. Press (1991)

85. Daniilidis, K., Bayro-Corrochano, E.: The dual quaternion approach to hand-eye calibration. In: Proceedings of 13th International Conference on Pattern Recognition. pp. 318–322. IEEE (1996)

86. Ying-Cherng Lu, Chou, J.C.K.: Eight-space quaternion approach for robotic hand-eye calibration. In: 1995 IEEE International Conference on Systems, Man and Cybernetics. Intelligent Systems for the 21st Century. pp. 3316–3321. IEEE (1995)

87. Li, A., Wang, L., Wu, D.: Simultaneous robot-world and hand-eye calibration using dual-quaternions and kronecker product. Int. J. Phys. Sci. 5, 1530–1536 (2010)

88. Wang, W., Slotine, J.-J.E.: A Theoretical Study of Different Leader Roles in Networks. IEEE Trans. Automat. Contr. 51, 1156–1161 (2006). https://doi.org/10.1109/TAC.2006.878754

89. Cicek, E., Dasdemir, J., Zergeroglu, E.: Coordinated synchronization of multiple robot manipulators with dynamical uncertainty. Trans. Inst. Meas. Control. 37, 672–683 (2015). https://doi.org/10.1177/0142331214550520

90. KUKA Deutschland GmbH: KUKA.RoboTeam 3.0 – For KUKA System Software 8.5. , Augsburg, Germany (2018)

91. Norman, A.R., Schönberg, A., Gorlach, I.A., Schmitt, R.: Validation of iGPS as an external measurement system for cooperative robot positioning. Int. J. Adv. Manuf. Technol. 64, 427–446 (2013). https://doi.org/10.1007/s00170-012-4004-8

92. Shen, N., Yuan, H., Li, J., Wang, Z., Geng, L., Shi, H., Lu, N.: Efficient Model-Free Calibration of a 5-Degree of Freedom Hybrid Robot. J. Mech. Robot. 14, 1–13 (2022). https://doi.org/10.1115/1.4053824

93. Messay, T., Ordóñez, R., Marcil, E.: Computationally efficient and robust kinematic calibration methodologies and their application to industrial robots. Robot. Comput. Integr. Manuf. 37, 33–48 (2016). https://doi.org/10.1016/j.rcim.2015.06.003

94. Chen, X., Zhang, Q., Sun, Y.: Non-kinematic calibration of industrial robots using a rigid–flexible coupling error model and a full pose measurement method. Robot. Comput. Integr. Manuf. 57, 46–58 (2019). https://doi.org/10.1016/j.rcim.2018.07.002

95. Theissen, N.A., Gonzalez, M.K., Barrios, A., Archenti, A.: Quasi-Static Compliance Calibration of Serial Articulated Industrial Manipulators. Int. J. Autom. Technol. 15, 590–598 (2021). https://doi.org/10.20965/ijat.2021.p0590

96. Bai, Y.: On the comparison of model-based and modeless robotic calibration based on a fuzzy interpolation method. Int. J. Adv. Manuf. Technol. 31, 1243–1250 (2007). https://doi.org/10.1007/s00170-005-0278-4

97. Zhao, G., Zhang, P., Ma, G., Xiao, W.: System identification of the nonlinear residual errors of an industrial robot using massive measurements. Robot. Comput. Integr. Manuf. 59, 104–114 (2019). https://doi.org/10.1016/j.rcim.2019.03.007

98. Aoyagi, S., Kohama, A., Nakata, Y., Hayano, Y., Suzuki, M.: Improvement of robot accuracy by calibrating kinematic model using a laser tracking system-compensation of non-geometric errors using neural networks and selection of optimal measuring points using genetic algorithm-. In: 2010 IEEE/RSJ International Conference on Intelligent Robots and Systems. pp. 5660–5665. IEEE (2010)

99. Meggiolaro, M.A., Dubowsky, S., Mavroidis, C.: Geometric and elastic error calibration of a high accuracy patient positioning system. Mech. Mach. Theory. 40, 415–427 (2005). https://doi.org/10.1016/j.mechmachtheory.2004.07.013

100. Nguyen, H.-N., Le, P.-N., Kang, H.-J.: A new calibration method for enhancing robot position accuracy by combining a robot model–based identification approach and an artificial neural network–based error compensation technique. Adv. Mech. Eng. 11, 168781401882293 (2019). https://doi.org/10.1177/1687814018822935

101. Bai, Y., Wang, D.: Using Shallow Neural Network Fitting Technique to Improve Calibration Accuracy of Modeless Robots. In: MacIntyre, J., Maglogiannis, I., Iliadis, L., and Pimenidis, E. (eds.) 15th IFIP International Conference on Artificial Intelligence Applications and Innovations (AIAI). pp. 623–631. Springer International Publishing, Cham (2019)

102. Wu, H., Tizzano, W., Andersen, T.T., Andersen, N.A., Ravn, O.: Hand-Eye Calibration and Inverse Kinematics of Robot Arm Using Neural Network. In: Kim, J.-H., Matson, E.T., Myung, H., Xu, P., and Karray, F. (eds.) Robot Intelligence Technology and Applications 2: Results from the 2nd International Conference on Robot Intelligence Technology and Applications. pp. 581–591. Springer International Publishing, Cham (2014)

103. Nguyen, H.-N., Zhou, J., Kang, H.-J.: A calibration method for enhancing robot accuracy through integration of an extended Kalman filter algorithm and an artificial neural network. Neurocomputing. 151, 996–1005 (2015). https://doi.org/10.1016/j.neucom.2014.03.085

104. Xu, W., Dongsheng, L., Mingming, W.: Complete calibration of industrial robot with limited parameters and neural network. In: 2016 IEEE International Symposium on Robotics and Intelligent Sensors (IRIS). pp. 103–108. IEEE (2016)

105. LI, B., TIAN, W., ZHANG, C., HUA, F., CUI, G., LI, Y.: Positioning error compensation of an industrial robot using neural networks and experimental study. Chinese J. Aeronaut. 35, 346–360 (2022). https://doi.org/10.1016/j.cja.2021.03.027

106. Su, H., Yang, C., Mdeihly, H., Rizzo, A., Ferrigno, G., De Momi, E.: Neural Network Enhanced Robot Tool Identification and Calibration for Bilateral Teleoperation. IEEE Access. 7, 122041–122051 (2019). https://doi.org/10.1109/ACCESS.2019.2936334

107. Ruan, C., Gu, X., Li, Y., Zhang, G., Wang, W., Hou, Z.: Base frame calibration for multi-robot cooperative grinding station by binocular vision. In: 2017 2nd International Conference on Robotics and Automation Engineering (ICRAE). pp. 115–120. IEEE (2017)

108. Gan, Y., Dai, X.: Base frame calibration for coordinated industrial robots. Rob. Auton. Syst. 59, 563–570 (2011). https://doi.org/10.1016/j.robot.2011.04.003

109. Santolaria, J., Ginés, M.: Uncertainty estimation in robot kinematic calibration. Robot. Comput. Integr. Manuf. 29, 370–384 (2013). https://doi.org/10.1016/j.rcim.2012.09.007

110. Bisong, E.: Building Machine Learning and Deep Learning Models on Google Cloud Platform. Apress, Berkeley, CA (2019)

111. Masters, D., Luschi, C.: Revisiting Small Batch Training for Deep Neural Networks. http://arxiv.org/abs/1804.07612. (2018)

112. Fleischer, J., Teti, R., Lanza, G., Mativenga, P., Möhring, H.C., Caggiano, A.: Composite materials parts manufacturing. CIRP Ann. 67, 603–626 (2018). https://doi.org/10.1016/j.cirp.2018.05.005

113. Sugita, N., Shu, L., Kimura, K., Arai, G., Arai, K.: Dedicated drill design for reduction in burr and delamination during the drilling of composite materials. CIRP Ann. 68, 89–92 (2019). https://doi.org/10.1016/j.cirp.2019.04.094

114. Tsao, C.C., Hocheng, H.: Computerized tomography and C-Scan for measuring delamination in the drilling of composite materials using various drills. Int. J. Mach. Tools Manuf. 45, 1282–1287 (2005). https://doi.org/10.1016/j.ijmachtools.2005.01.009

115. Davim, J.P., Reis, P.: Drilling carbon fiber reinforced plastics manufactured by autoclave—experimental and statistical study. Mater. Des. 24, 315–324 (2003). https://doi.org/10.1016/S0261-3069(03)00062-1

116. Campos Rubio, J., Abrao, A.M., Faria, P.E., Correia, A.E., Davim, J.P.: Effects of high speed in the drilling of glass fibre reinforced plastic: Evaluation of the delamination factor. Int. J. Mach. Tools Manuf. 48, 715–720 (2008). https://doi.org/10.1016/j.ijmachtools.2007.10.015

117. Gaugel, S., Sripathy, P., Haeger, A., Meinhard, D., Bernthaler, T., Lissek, F., Kaufeld, M., Knoblauch, V., Schneider, G.: A comparative study on tool wear and laminate damage in drilling of carbon-fiber reinforced polymers (CFRP). Compos. Struct. 155, 173–183 (2016). https://doi.org/10.1016/j.compstruct.2016.08.004

118. Hrechuk, A., Bushlya, V., Ståhl, J.-E.: Hole-quality evaluation in drilling fiber-reinforced composites. Compos. Struct. 204, 378–387 (2018). https://doi.org/10.1016/j.compstruct.2018.07.105

119. Voß, R., Henerichs, M., Rupp, S., Kuster, F., Wegener, K.: Evaluation of bore exit quality for fibre reinforced plastics including delamination and uncut fibres. CIRP J. Manuf. Sci. Technol. 12, 56–66 (2016). https://doi.org/10.1016/j.cirpj.2015.09.003

120. Caprino, G., Tagliaferri, V.: Damage development in drilling glass fibre reinforced plastics. Int. J. Mach. Tools Manuf. 35, 817–829 (1995). https://doi.org/10.1016/0890-6955(94)00055-O

121. Stone, R., Krishnamurthy, K.: A neural network thrust force controller to minimize delamination during drilling of graphite-epoxy laminates. Int. J. Mach. Tools Manuf. 36, 985–1003 (1996)

122. Khashaba, U.A.: Delamination in drilling GFR-thermoset composites. Compos. Struct. 63, 313–327 (2004). https://doi.org/10.1016/S0263-8223(03)00180-6

123. Seif, M.A., Khashaba, U.A., Rojas-Oviedo, R.: Measuring delamination in carbon/epoxy composites using a shadow moiré laser based imaging technique. Compos. Struct. 79, 113–118 (2007). https://doi.org/10.1016/j.compstruct.2005.11.039

124. Faraz, A., Biermann, D., Weinert, K.: Cutting edge rounding: An innovative tool wear criterion in drilling CFRP composite laminates. Int.

J. Mach. Tools Manuf. 49, 1185–1196 (2009). https://doi.org/10.1016/j.ijmachtools.2009.08.002

125. Faraz, A., Biermann, D.: Subsequent drilling on pilot holes in woven carbon fibre reinforced plastic epoxy laminates: the effect of drill chisel edge on delamination. Proc. Inst. Mech. Eng. Part B J. Eng. Manuf. 225, 1493–1504 (2011). https://doi.org/10.1177/0954405411413335

126. Sause, M.G.R.: In Situ Monitoring of Fiber-Reinforced Composites. (2016)

127. Caggiano, A., Angelone, R., Teti, R.: Image Analysis for CFRP Drilled Hole Quality Assessment. Procedia CIRP. 62, 440–445 (2017). https://doi.org/10.1016/j.procir.2017.03.045

128. Shi, Y., Zhang, Y., Harik, R.: Manufacturing feature recognition with a 2D convolutional neural network. CIRP J. Manuf. Sci. Technol. 30, 36–57 (2020). https://doi.org/10.1016/j.cirpj.2020.04.001

129. Zouhri, W., Dantan, J.Y., Häfner, B., Eschner, N., Homri, L., Lanza, G., Theile, O., Schäfer, M.: Optical process monitoring for Laser-Powder Bed Fusion (L-PBF). CIRP J. Manuf. Sci. Technol. 31, 607–617 (2020). https://doi.org/10.1016/j.cirpj.2020.09.001

130. Fujishima, M., Narimatsu, K., Irino, N., Mori, M., Ibaraki, S.: Adaptive thermal displacement compensation method based on deep learning. CIRP J. Manuf. Sci. Technol. 25, 22–25 (2019). https://doi.org/10.1016/j.cirpj.2019.04.002

131. Fujishima, M., Narimatsu, K., Irino, N., Ido, Y.: Thermal displacement reduction and compensation of a turning center. CIRP J. Manuf. Sci. Technol. 22, 111–115 (2018). https://doi.org/10.1016/j.cirpj.2018.04.003

132. Ye, Y., Ma, K., Zhou, H., Arola, D., Zhang, D.: An automated shearography system for cylindrical surface inspection. Measurement. 135, 400–405 (2019). https://doi.org/10.1016/j.measurement.2018.11.085

133. Cha, Y.-J., Choi, W., Büyüköztürk, O.: Deep Learning-Based Crack Damage Detection Using Convolutional Neural Networks. Comput. Civ. Infrastruct. Eng. 32, 361–378 (2017). https://doi.org/10.1111/mice.12263

134. Long, J., Shelhamer, E., Darrell, T.: Fully Convolutional Networks for Semantic Segmentation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR). pp. 3431–3440 (2015)

135. Ronneberger, O., Fischer, P., Brox, T.: U-Net: Convolutional Networks for Biomedical Image Segmentation. In: IEEE Access. pp. 234–241 (2015)

136. Liu, Z., Cao, Y., Wang, Y., Wang, W.: Computer vision-based concrete crack detection using U-net fully convolutional networks. Autom. Constr.

104, 129–139 (2019). https://doi.org/10.1016/j.autcon.2019.04.005

137. Pintus, R., Dulecha, T.G., Ciortan, I., Gobbetti, E., Giachetti, A.: State-of-the-art in Multi-Light Image Collections for Surface Visualization and Analysis. Comput. Graph. Forum. 38, 909–934 (2019). https://doi.org/10.1111/cgf.13732

138. Louverdis, G., Vardavoulia, M.I., Andreadis, I., Tsalides, P.: A new approach to morphological color image processing. Pattern Recognit. 35, 1733–1741 (2002). https://doi.org/10.1016/S0031-3203(01)00166-2

139. Gonzalez, C.I., Melin, P., Castro, J.R., Mendoza, O., Castillo, O.: An improved sobel edge detection method based on generalized type-2 fuzzy logic. Soft Comput. 20, 773–784 (2016). https://doi.org/10.1007/s00500-014-1541-0

140. Nair, V., Hinton, G.E.: Rectified linear units improve restricted boltzmann machines. In: Proceedings of the 27th international conference on machine learning (ICML-10) (2010)

141. Scherer, D., Müller, A., Behnke, S.: Evaluation of Pooling Operations in Convolutional Architectures for Object Recognition. In: BMC Bioinformatics. pp. 92–101 (2010)

142. Ni, Z.-L., Bian, G.-B., Zhou, X.-H., Hou, Z.-G., Xie, X.-L., Wang, C., Zhou, Y.-J., Li, R.-Q., Li, Z.: RAUNet: Residual Attention U-Net for Semantic Segmentation of Cataract Surgical Instruments. In: Gedeon, T., Wong, K.W., and Lee, M. (eds.) Neural Information Processing. pp. 139–149. Springer International Publishing, Cham (2019)

143. Milletari, F., Navab, N., Ahmadi, S.A.: V-Net: Fully convolutional neural networks for volumetric medical image segmentation. Proc. - 2016 4th Int. Conf. 3D Vision, 3DV 2016. 565–571 (2016). https://doi.org/10.1109/3DV.2016.79

144. Chen, W.-C.: Some experimental investigations in the drilling of carbon fiber-reinforced plastic (CFRP) composite laminates. Int. J. Mach. Tools Manuf. 37, 1097–1108 (1997). https://doi.org/10.1016/S0890-6955(96)00095-8

145. DAVIM, J., RUBIO, J., ABRAO, A.: A novel approach based on digital image analysis to evaluate the delamination factor after drilling composite laminates. Compos. Sci. Technol. 67, 1939–1945 (2007). https://doi.org/10.1016/j.compscitech.2006.10.009

146. Babu, J., Paul Alex, N., Abraham, S.P., Philip, J., Anoop, B., Davim, J.P.: Development of a comprehensive delamination assessment factor and its evaluation with high-speed drilling of composite laminates using a twist

drill. Proc. Inst. Mech. Eng. Part B J. Eng. Manuf. 232, 2109–2121 (2018). https://doi.org/10.1177/0954405417690552