An Intelligent Framework for Autonomous Robot-based Machine-tending Applications

by

Feiyu Jia

A thesis submitted in partial fulfillment of the requirements for the degree of

Doctor of Philosophy

Department of Mechanical Engineering

University of Alberta

Abstract

The manufacturing sector stands as a fundamental pillar of worldwide economies, contributing markedly to global economic expansion. Over the past five years, the implementation of automated machine-tending systems has widely extended from simulation or laboratory environments to real-world scenarios in manufacturing workshops, as robotics and artificial intelligence develop rapidly. Machine tending is a critical part of the manufacturing process through interaction with the machine and surrounding environments. Currently, most of the machine tending tasks are still carried out manually or via collaborative robots by cooperating with humans on site. However, with the development of AI-enabled robots, intelligent manufacturing has been moving from mass production to mass customization and uses robots and artificial intelligence techniques to minimize human interventions in manufacturing activities. Inspection of the machine' working status is critical in manufacturing processes, ensuring that machines work correctly without any interruptions, e.g., in lights-out manufacturing. In addition, autonomous robot-based machine tending applications are necessary for smart factories with the increasing demand for full autonomy. So far, there is no attempt has been made toward the framework for fully autonomous robot-based machine-tending applications. Consequently, this research aims to develop an intelligent framework to attend CNC machines by integrating autonomous mobile manipulation systems, scene text recognition, real-time object detection, position estimation and path planning techniques to achieve fully autonomous operations of the mobile manipulator in manufacturing environments, which consist of four main stages: 1) a path planning, and docking to charging station method for autonomous mobile manipulator system to enable system move between different workstations and autonomous charging for continuously and smoothly working; 2) an automatic object detection method for the machine-tending system to identify the target

machine in the workspace; 3) an intelligent inspection of machine's working status through command recognition; 4) a button detection and localization method to support the manipulation by moving toward the control buttons to execute machine instructions.

In the first stage, an autonomous mobile manipulation system (AMM) is proposed for part handling, loading and unloading, and some other auxiliary tasks in machine-tending. In addition, an improved path-planning algorithm based on Rapidly-exploring Random Tree (RRT) and the quintic B-spline curve technique is proposed for robotic machine-tending systems to move between the workstation and the charging dock. Furthermore, an autonomous docking and charging method is developed for machine-tending systems to work continuously in manufacturing environments. This method requires two steps: i) detecting the charging station, typically in an unstructured environment, and ii) autonomously docking to the charging station. For charging station detection, a YOLOv7-based method is developed to quickly and accurately recognize the charging station.

The second stage of the proposed framework is the identification of the target CNC machine. It is important to note that there is often more than one CNC machine working in a representative manufacturing environment. Therefore, a deep learning-based machine detection method, called SiameseRPN, is developed to recognize the specific machine from a group of machines in the workspace. This method combines the Siamese neural network and region proposal network.

In the third part, a command recognition method by integrating the text region proposal network, recurrent neural network and connectionist temporal classification was proposed to read and understand the CNC machine instruction for further operation. To improve the accuracy of recognition performance, a dictionary-guided procedure is also proposed.

In the last stage, a benchmark dataset for five different types of control buttons on the Haas CNC machine is created and the YOLOv7-based benchmark button detection method is developed to identify and localize the target buttons recognized in the machine commands to assist in the instructions executions of robotic machine-tending systems.

Preface

This thesis is an original work by Feiyu Jia, and the research is performed under the supervision of Dr. Rafiq Ahmad and Dr. Yongsheng Ma. Some parts of this thesis have been submitted or published and are listed below. As such, the thesis is organized in paper format by following the paper-based thesis guidelines.

1. F. Jia, Z. Samadikhoshkho, Muhammad Tufail, Y. Ma, and R. Ahmad (2023) “An intelligent operation framework for autonomous robot-based machine-tending applications”, Applied Intelligence. (Under Revision). (Chapter 6). (I was responsible for conceptualization, literature review, coding, model construction, and writing of the original manuscript; R. Ahmad is the main supervisor and principal investigator of the project, and Y. Ma was the co-supervisor on this research).

2. F. Jia, Y. Ma, and R. Ahmad (2023) “Review of current vision-based robotic machine-tending applications”, The International Journal of Advanced Manufacturing. (Chapter 2) (I was responsible for the conceptualization, literature review, and writing of the original manuscript; R. Ahmad is the main supervisor and principal investigator of the project, and Y. Ma was the co-supervisor on this research).

3. F. Jia, M. Afaq, B. Ripka, Q. Huda, and R. Ahmad (2023) “Vision and Lidar-based autonomous docking and recharging of a mobile robot for machine tending in autonomous manufacturing environments”, Applied Science, vol. 13, no. 19, p.10675. (Chapter 4) (I was responsible for the conceptualization, literature review, data collection, coding and model construction, and writing of the original manuscript; R. Ahmad is the main supervisor and principal investigator of the project, and Y. Ma was the co-supervisor on this research).

4. F. Jia, A. Jebelli, Y. Ma, and R. Ahmad (2022) "An Intelligent Manufacturing Approach Based on a Novel Deep Learning Method for Automatic Machine and Working Status Recognition", Applied Science, vol. 12, no. 11, p.5697. (Chapter 5) (I was responsible for the conceptualization, literature review, coding, model construction, and writing of the original manuscript; R. Ahmad is the main supervisor and principal investigator of the project, and Y. Ma was the co-supervisor on this research).
5. F. Jia, Y. Ma, and R. Ahmad (2021) "Vision-Based Associative Robotic Recognition of Working Status in Autonomous Manufacturing Environment", *Procedia CIRP*, vol. 104, pp. 1535–1540. (Chapter 5) (I was responsible for the conceptualization, literature review, coding, model construction, and writing of the original manuscript; R. Ahmad is the main supervisor and principal investigator of the project, and Y. Ma was the co-supervisor on this research).
6. F. Jia, J. Tzintzun, and R. Ahmad (2020) "An Improved Robot Path Planning Algorithm for a Novel Self-adapting Intelligent Machine Tending Robotic System", *in Mechanisms and Machine Science,* vol. 86, Springer International Publishing, 2020, pp. 53–64. (Chapter 3) (I was responsible for the conceptualization, literature review, coding, model construction, and writing of the original manuscript; R. Ahmad is the main supervisor and principal investigator of this research)

Acknowledgments

First and foremost, I would sincerely thank my supervisors, Dr. Rafiq Ahmad and Dr. Yongsheng Ma, for their kind guidance, constant support, care, and encouragement. Without their help, this work would not be possible. Also, I would like to thank my supervisory committee member, Dr. Martin Barczyk, who has provided invaluable suggestions to improve my research work.

I would like to thank all my colleagues from Dr. Rafiq Ahmad's group and the SMART lab, who contributed to a friendly research environment and helped me a lot in research. They are more than colleagues; they are friends and families.

Last but not least, I appreciate the support from my beloved parents. Without their support and encouragement, I would not have completed this journey.

Thank You.

Table of Contents

List of Tables

List of Figures

List of Abbreviations

SMEs	Small and Medium-sized Enterprises
CNC	Computer Numerical Control
HMI	Human-Machine Interface
MT	Machine Tending
RRT	Rapidly-exploring Random Trees
CV	Computer Vision
AI	Artificial Intelligence
STR	Scene Text Recognition
Cobots	Collaborative Robots
AMRs	Autonomous Mobile Robots
AMMs	Autonomous Mobile Manipulators
OCR	Optical Character Recognition
ML	Machine Learning
DL	Deep Learning
SVM	Support Vector Machine
NNs	Neural Networks
ANNs	Artificial Neural Networks

RL	Reinforcement Learning
CNNs	Convolutional Neural Networks
GANs	Generative Adversarial Networks
LSTM	Long Short-Term Memory
RNNs	Recurrent Neural Networks
YOLO	You Only Look Once
R-CNNs	Region-based CNNs
SSD	Single Shot Detector
RPN	Region Proposal Network
HOG	Histogram of Oriented Gradients
SNNs	Siamese Neural Networks
SiameseRPN	Siamese Region Proposal Network
CRNN	Convolutional Recurrent Neural Network
SGD	Stochastic Gradient Descent
IoU	Intersection over Union
FFT	Fast Fourier Transform
mAP	Mean Average Precision

Chapter 1 : Introduction

1.1 Background

1.1.1 Manufacturing Industry

The manufacturing industry is a vital component of global economies, which holds immense importance and contributes significantly to overall economic growth. Recent global manufacturing industry reports indicate that the manufacturing sector constitutes an estimated 16% of the global GDP, and the global general manufacturing market is expected to increase from US$649.8 billion in 2020 to US$732.2 billion by 2027 [1]. It is also an essential source of employment, employing millions of workers worldwide and accounting for approximately 14% of total employment [1]. The significance of the manufacturing industry extends beyond its substantial GDP contribution and the provision of employment. It serves as a critical catalyst for innovation and technological progression.

Furthermore, small and medium-sized enterprises (SMEs) in the manufacturing industry constitute a significant and vibrant portion of the global economy. These entities are instrumental in profit generation and economic development. Recent studies reveal that SMEs represent an estimated 45% of the total added value in the manufacturing sector while employing over 70 million individuals worldwide, a significant proportion of total manufacturing employment [2].

Despite its achievements, the global manufacturing industry is facing several challenges. A persistent labor shortage is a prominent issue. As the manufacturing industry continues to evolve and integrate new technologies, skilled and experienced workers are becoming scarce, resulting in difficulties in recruiting and retaining qualified personnel [3]. Additionally, the imperative of ensuring worker safety continues to be a major concern as manufacturers aim to construct safe working environments and minimize accident risks [4].

To increase productivity and gain a competitive edge over other manufacturing factories, the concept of lights-out manufacturing was proposed. It advocates for machinery to operate round-the-clock, without human intervention or with minimal human involvement during the second and third shifts [5]. Typically, extending operational hours demands extensive planning and scheduling, and often leads to increased labor expenses through either overtime or hiring more staff. However, with the advancement of Industry 4.0, The transformation of machine-tending systems into full robotic automation by integrating robotics, smart sensors, artificial intelligence and the Internet of Things, makes it feasible to achieve uninterrupted nighttime production without needing human oversight. Thus, in lights-out factories, all manufacturing tasks are carried out entirely by autonomous robotic systems, eliminating the necessity for labor force, light and windows. This pattern improves efficiency by prolonging operational hours and reducing defects in products. Due to energy savings, high efficiency, and the considerable reduction in labor costs, the lights-out factory has been widespread and preferred by most enterprises [6]. Moreover, some unexpected situations, such as the COVID-19 pandemic, have considerably impacted the global manufacturing industry, accelerating the implementation of advanced technologies to facilitate remote or autonomous operations [7]. This crisis underscored the importance of digital transformation and Industry 4.0 in establishing more flexible and resilient autonomous systems [8].

1.1.2 Machine Tending (MT)

Machine tending refers to the operation of industrial machine tools in a manufacturing plant performed by human operators or automation systems [4], which is the significant process of monitoring and supervising machines on the shop floor., Machine tending activities, such as loading and unloading materials or workpieces from machines or production lines, monitoring machines through initiating, stopping machine operations, and adjusting machine settings, and conducting quality inspections of outputs to ensure the seamless and uninterrupted operation of the machinery [9], are shown in Figure 1.1. Consequently, machine-tending tasks can often be repetitive, monotonous, physically demanding, and time-intensive.

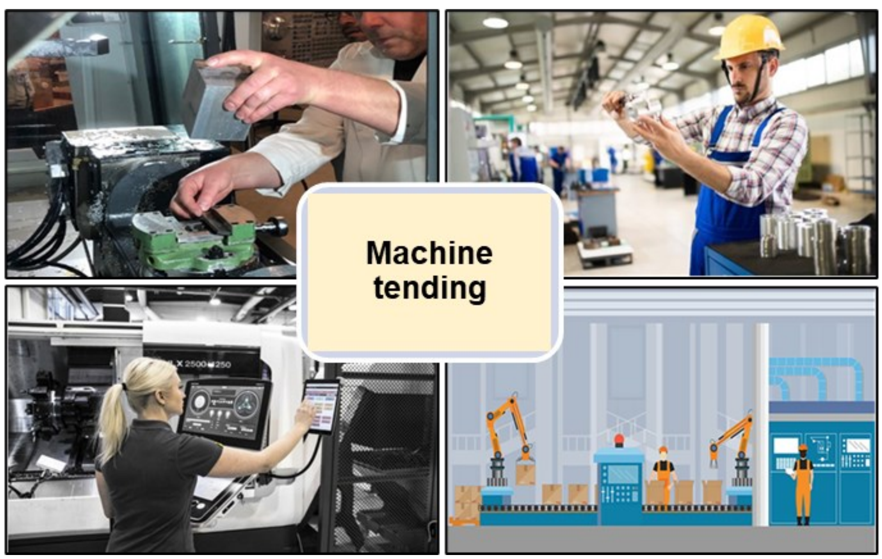

Figure 1.1: Example of machine-tending activities.

Nonetheless, recent progress in robotics and sensor technology is increasingly instrumental in addressing these problems within the manufacturing sector. By incorporating these advanced technologies, many automated robotic machine-tending systems [10], [11], [11], [12], [12] have been developed and deployed to improve productivity, which are shown in Figure 1.2, minimize manual labor, enhance consistency, and increase overall operational efficiency. Thus, human operators are liberated to concentrate on more intricate tasks that necessitate critical thinking and decision-making capabilities.

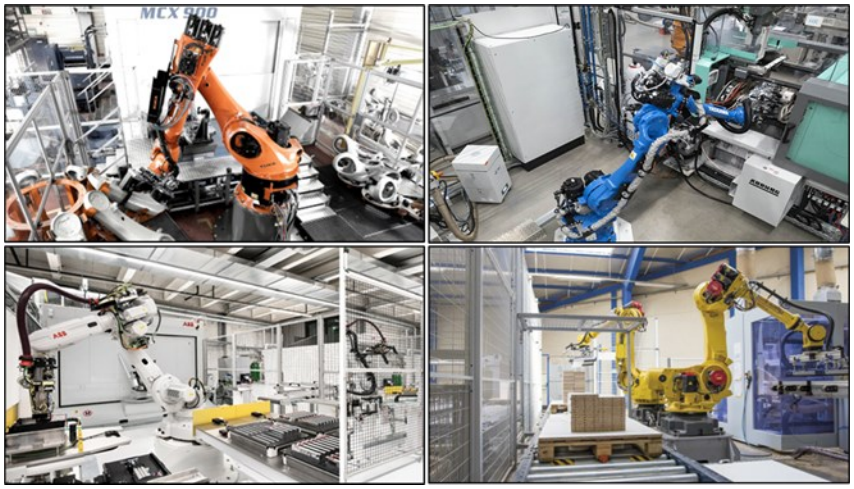

Figure 1.2: Examples of machine tending by conventional industrial robot systems.

Machine-tending applications are the oldest of applications for traditional industrial robots and are even more economically beneficial today than they were when the first robot was installed on a die-casting machine [4]. Nowadays, those benefits are more pronounced than before, with the cost of industrial robots in real terms has declined while labor rates have soared. At the same time, robots are helping manufacturers address many of the key challenges they face, including tight labor pools, global market competitiveness, and safety.

However, while conventional industrial robots have proven essential in automating machine tending tasks, they are limited by their capacities. They are typically designed for specific tasks, with a limited capacity to adapt to changes or to handle new tasks without substantial reprogramming or physical adjustments [13]. Moreover, conventional industrial robots typically necessitate operation within confined areas or cages to ensure worker safety [14]. They lack the requisite intelligence capabilities to operate collaboratively with humans without posing a safety risk, which restricts their use in cooperative scenarios [15]. Furthermore, they function optimally in highly controlled environments with precisely defined tasks. Any alteration in these conditions or deviation from the expected can precipitate substantial errors or even a complete cessation of operations [16]. From the perspective of most SMEs, traditional industrial robots represent a considerable financial outlay both in terms of initial acquisition and ongoing maintenance costs [1]. These robotic machine-tending systems demand significant space, limiting their suitability for constrained working environments and small manufacturing facilities.

1.2 Research Motivation

Numerous research and developments have been carried out to overcome these limitations and contributed to the application of robot machine-tending systems. Autonomous and self-adaptive robot systems can be a solution to achieve fully automated machine tending, which can adapt to changing environments, learn to make decisions on their own based on an assortment of integrated digital technologies and enterprise-wide data, and evolve as the environment around them changes [17], [18], [19]. Autonomous mobile manipulator (AMM) systems can be used to realize fully-automated machine tending, which expands the flexibilities and capabilities of machine-tending systems by combining locomotion with manipulation abilities. Figure 1.3 shows an AMM used in this study in machine tending.

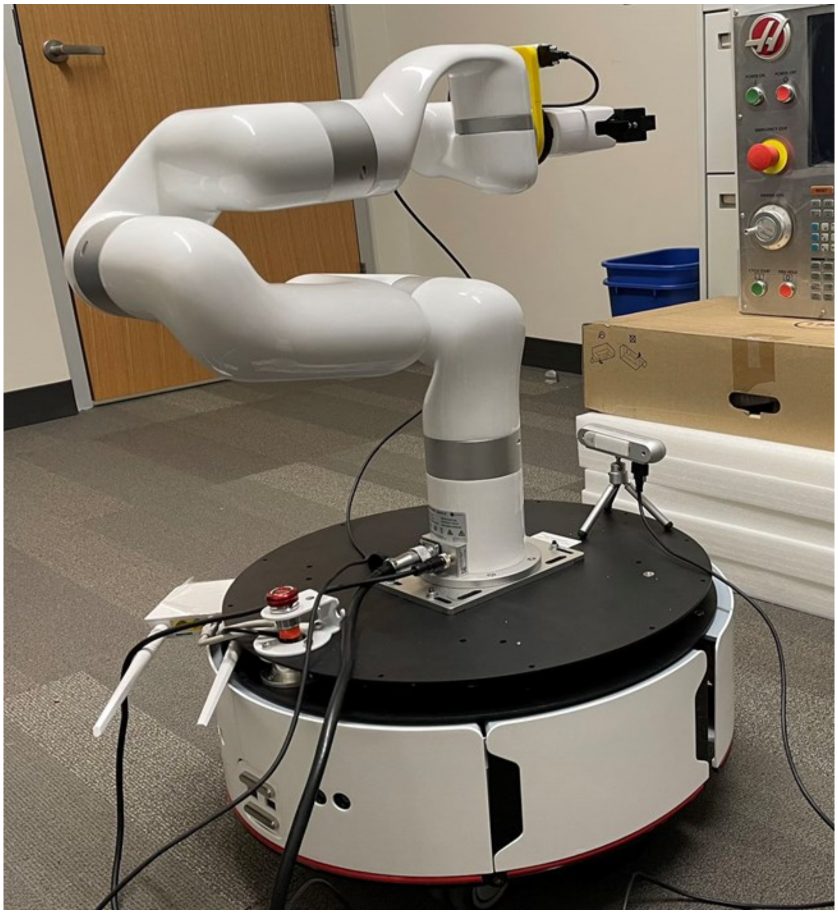

Figure 1.3: The AMM system used in this study.

Despite all the advantages offered by current machine-tending applications, they are still in the very early stage of development and face numerous challenges to achieve fully autonomous machine tending with minimal or without human intervention. So far, machine-tending tasks are still mainly performed based on extensive human intervention, and there is no attempt has been made toward the intelligent framework to achieve fully autonomous robot-based machine tending [13]. To minimize or completely replace the human workers in the machine-tending tasks, the machine-tending system needs to learn and imitate human behavior, called the "eye-brain-hand" process, which is necessary to achieve autonomous machine tending [16]. This process can be divided into four steps. Firstly, identify the target machine from a group of machines. Secondly, read and understand the commands on the machine's display to execute the instructions and handle the emergency. Then, detect and localize the related target buttons in real-time once the commands are recognized, and instructions are associated with specific buttons. Finally, the robotic machine-tending system executes the commands. Although many object detection methods have been developed, these methods mainly focus on classifying and detecting different object categories. Therefore, these methods are not suitable for detecting a target machine from many similar CNC machines in the manufacturing environment. Another issue is that a benchmark dataset and a button detection method are lacking for machine-tending tasks. In addition, a proper command recognition method is missing for current autonomous machine-tending systems. Finally, the docking and autonomous charging method for the mobile platform is also necessary to enable the autonomous machine-tending systems to work continuously and smoothly between different workstations with surrounding environments.

Therefore, to address these problems, this thesis investigates and explores an intelligent framework for fully autonomous robot-based machine-tending applications by integrating robotics, vision sensors, computer vision, and deep learning technologies. The system envisioned shall be completely independent and work independently from the CNC machine, similar to a machine-tending operator.

1.3 Research Objectives

The primary purpose of this thesis is to widely facilitate and accelerate fully autonomous machine tending by integrating emerging technologies into machine-tending robotic systems, as it provides the potential to overcome the challenges and issues in the manufacturing sector. Working toward the inclusion of full automation and advanced technologies and techniques can assist in reducing the costs of the initial installation costs and human labor usage, improve production accuracy and efficiency, and improve safety issues. For this purpose, this thesis aims to research emerging technologies in the machine-tending domain, and the main objective of the thesis is to ***investigate and develop an intelligent framework for autonomous robot-based machine-tending applications***.

The specific objectives of developing this framework are outlined as follows:

- **O1:** Review the machine-tending applications in terms of their current situation, associated emerging technologies, challenges, and prospective research trends (literature analysis).
- **O2:** Design an autonomous recharging strategy for machine-tending systems to enable them to work continuously and smoothly during machine-tending applications (mobile system path planning and control).
- **O3:** Develop an intelligent machine detection method to identify the specific CNC machine from a group of machines in the workspace (machine detection, identification, and localization).
- **O4:** Develop an autonomous command recognition method for the inspection of the CNC machine's working status (automated text detection, recognition, and interpretation).
- **O5:** Develop a real-time button detection and localization method for CNC machine control keys and a control scheme to assist in the further operations of robotic machine-tending systems (automated task execution and control).

1.4 Research Methodology

The proposed methodology aims to achieve a fully autonomous machine-tending process. A preliminary investigation was conducted, and a systematic literature review was completed to explore state-of-the-art machine-tending applications in current situations, emerging technologies, main challenges, and future directions. The current machine-tending application can be improved by integrating path planning, real-time object detection and localization, scene text recognition, and intelligent decision-making. An overview of the methodology is presented in Figure 1.4.

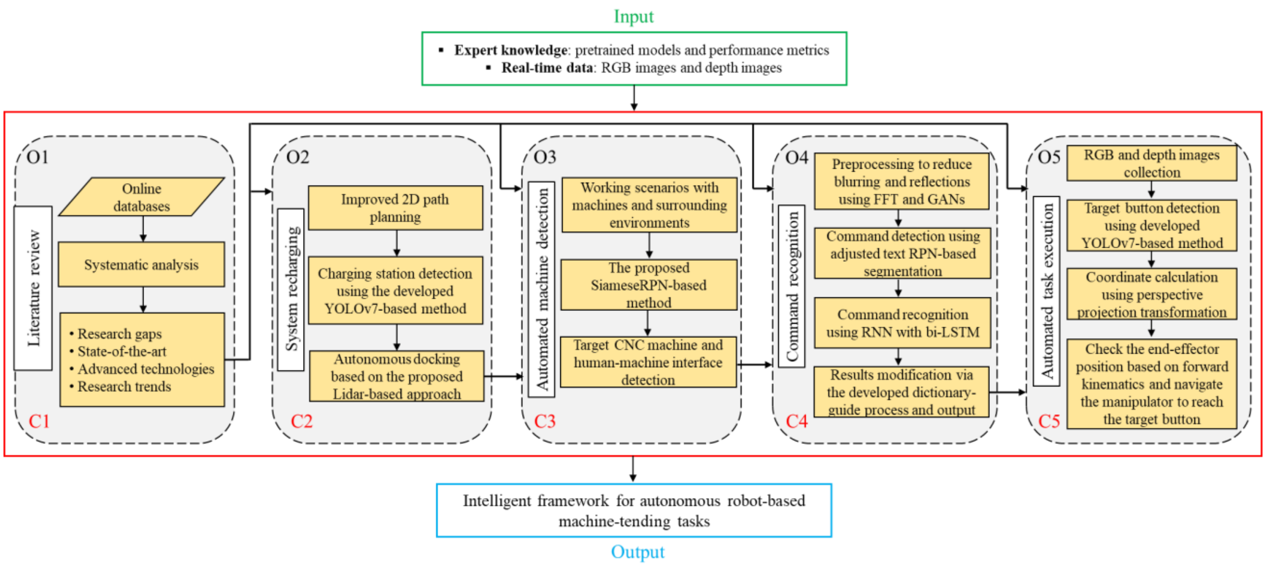

Figure 1.4: Overview of the methodology.

With the input of real-time data obtained from the practical manufacturing environment, the four main objectives of this thesis are addressed in the main process. The proposed methodology is developed based on five research gaps with proposed solutions: preliminary investigation, autonomous charging, intelligent machine detection, working status inspection, and real-time button detection and localization. The details of each step are explained as follows:

- For preliminary investigation, a systematic literature review based on the Protocol of Preferred Reporting Items for Systematic Review and Meta-Analyses to analyze the 50 scientific literatures related to machine tending in the last five years is conducted. It contributes to the evolution of machine-tending applications by investigating the impacts of emerging trends of advanced technologies, such as autonomous mobile robots, computer vision, machine learning, and deep learning.

- For autonomous charging for the machine-tending system, there are three main stages. Firstly, an improved intelligent robot path planning method was proposed to enable machine-tending systems to move between workstations and charge stations. Once the system is close to the charging station, a YOLOv7-based method is applied to recognize and localize the charger. Finally, a vision and lidar-based docking strategy is developed to achieve quick and accurate docking of the machine-tending system to the target charger.
- Intelligent machine detection is performed using the proposed deep learning-based method: SiameseRPN. It consists of two subnetworks: the region proposal network and the Siamese neural networks. It is observed that it performs better in distinguishing one specific CNC machine from a group of similar machines compared with other object detection methods.
- In working status inspection, a deep learning-based command recognition method is developed by combining the text detection and recognition branches. Its architecture includes three parts: the adjusted text region proposal network, the recurrent neural network, and the connectionist temporal classification. Following the dictionary-guided procedure, it can achieve an accuracy of 100% in recognizing machine instructions.
- In the last stage, to achieve the basic operation of machine-tending systems, a benchmark dataset about machine control keys is created, and a benchmark YOLOv7-based button detection and localization method is proposed and evaluated for future utilization in machine-tending tasks.

1.5 Thesis Outline

This thesis consists of seven chapters.

Chapter 1 provides the background of machine tending in the manufacturing industry. The motivations of this research are summarized by the challenges and issues of the implementation of emerging technologies. A brief statement on the objectives of this research and the main methodologies used are also presented at the end of this chapter.

Chapter 2 presents the first research contribution, “Review of current vision-based robotic machine-tending applications”. It investigates the recent research in the machine-tending domain and summarizes state-of-the-art topics covered in this thesis, including robotics, target machine detection in environments, working status recognition, and abnormal handling.

Chapter 3 presents an improved path-planning algorithm based on an RRT and quintic B-spline curve to effectively generate smooth and collision-free paths for our designed Novel Self-adapting Intelligent Robotic Machine-Tending Systems. This method will be further integrated with the docking method proposed in Chapter 4 for autonomous charging for the machine-tending systems in manufacturing environments.

Chapter 4 proposes an autonomous docking method based on computer vision and lidar sensors for a mobile robot operating in a manufacturing environment. The proposed method is based on a lidar-based approach and Yolov7 models that can quickly and accurately recognize the charging station.

Chapter 5 proposes a deep learning-based method for the CNC machine detection and working status recognition through an independent robot system without human intervention. Given that there is often more than one machine working in a representative industrial environment, the SiameseRPN method is developed to recognize and locate a specific machine from a group of machines on the shop floor. A deep learning-based text recognition method is designed to identify the working status from the human–machine interface (HMI) display, and it is evaluated in a simulation environment.

Chapter 6 proposes three parts of an intelligent operation framework for autonomous robot-based machine tending applications. First, a command recognition method by using Fast Fourier Transform (FFT) and Generative Adversarial Networks (GANs) to reduce blurring and reflections. A dictionary-guided modification is developed to correct the output recognized results. This method is validated and achieves an accuracy of 100%. Second, a YOLOv7-based button detection method is developed. In this step, a benchmark dataset for five different types of control buttons on the Haas CNC machine is created, and the proposed button detection method is evaluated to observe its performance in real-world scenarios. This model can achieve an overall mAP_0.5 of 98.8%. Finally, according to the results of case studies, the mobile manipulator can successfully reach the targeted button through the control scheme in the proposed intelligent framework.

Finally, the conclusion of this thesis, summary of research contributions, limitations of this study, and future works are discussed in Chapter 7.

Chapter 2 : State of the Art

2.1 Chapter Overview

The manufacturing sector stands as a fundamental pillar of worldwide economies, contributing markedly to global economic growth [2]. However, the manufacturing industry is persistently confronted with issues impeding its development and expansion, such as manpower shortages, safety concerns, the high initial investment for installation, and long return on investment [13]. Within this context, machine tending has become a crucial component of the manufacturing process and potentially serves as a viable solution to the aforementioned predicaments. Over the past five years, the implementation of automated machine-tending systems has widely extended from simulation or laboratory environments to practical application in manufacturing workshops, as robotics and artificial intelligence develop rapidly. To fully benefit from the potential of machine-tending applications, it is necessary to comprehend and tackle the challenges associated with it. Therefore, this chapter aims to contribute to the evolution of machine-tending applications by investigating the impacts of emerging trends of advanced technologies, such as autonomous mobile robots, computer vision, machine learning and deep learning. This chapter, hence, offers a systematic literature review based on the Protocol of Preferred Reporting Items for Systematic Review and Meta-Analyses to analyze the 50 scientific literatures related to machine tending in the last five years. The findings of this review elucidate the prevailing trends in emerging technologies that are advancing the autonomy of machine tending. A noteworthy observation is that most of the research and applications are currently in their prototypical stage. Additionally, this chapter deliberates some challenges and potential future perspectives for achieving fully autonomous machine tending. In conclusion, this review provides valuable insights into the current situation, identified bottlenecks and prospective future directions within the domain of machine-tending applications.

2.1.1 Research Motivation and Objective

Despite the enumerated limitations, Ongoing advancements in recent years in robotics and artificial intelligence are gradually overcoming these challenges, paving the way for the evolution of more flexible and adaptable intelligent robotic machine-tending systems [20]. However, the absence of a comprehensive review of the machine tending in terms of its current situation, associated emerging technologies, challenges, and prospective research trends, motivates me to undertake this study. The primary aim of this review is to elucidate the machine-tending applications that have emerged within the past five years, particularly in the context of the increasing implementation of advanced digital technologies in machine-tending applications.

2.1.2 Chapter Organization

This chapter presents a detailed analysis of 50 studies from the last five years, highlighting present-day machine-tending applications and advanced technologies. The chapter is structured as follows. Section 2.2 presents a methodology used to obtain the relevant literature. Section 2.3 discusses the statistical results observed after a general analysis of the selected research studies and provides a detailed overview of the advanced technologies used in machine-tending applications. Section 2.4 discusses the challenges, open issues, and future perspectives of machine tending. Finally, Section 2.5 concludes the review.

2.2 Literature Review Method

This chapter conducts a systematic literature review to investigate the current machine-tending applications in the manufacturing industry. Cases are searched where the term “machine tending" appeared in the title, abstract, or keywords. In addition, a review protocol is also defined, and an evaluation process is applied to ensure a high-quality search process and make a systematic literature review. These steps are explained in the sequential subsections to follow.

2.2.1 Review Protocol

A review protocol can provide an efficient strategy for literature review. This chapter follows the adjusted Preferred Reporting Items for Systematic Reviews and Meta-Analyses (PRISMA) [21] framework which aims to identify, evaluate and interpret the relevant literature to answer certain research questions. After establishing the research questions, a search strategy is defined to identify the relevant literature. Based on the research objective, four online publication repositories are used: Scopus, ScienceDirect, IEEE Xplore, and Web of Science. Finally, an approach is developed to identify pertinent publications on predefined inclusion and exclusion criteria, presented in Table 2.1.

Table 2.1: Study selection criteria.

Inclusion criteria	• Peer-reviewed journal and conference papers • Studies published between 2019 and 2023 • Articles should provide answers to the research questions • Papers must include the title, year, source, abstract, and DOI
Exclusion criteria	• Summaries of events and seminars, book reviews, and editorial • Papers published before 2019 • The publication is not available in full text • The publication is not in English

2.2.1.1 Research Questions

An explicit protocol organizes this systematic review to answer a series of research questions. These research questions aim to gain insights into advanced digital technologies in machine-tending applications ranging across different dimensions. Three research questions addressed in this literature review are listed below:

- RQ1: What emerging technologies have been applied for machine-tending applications?
- RQ2: How have these technologies impacted machine tending?
- RQ3: What are the main challenges and future directions to achieve fully autonomous machine tending by integrating these technologies?

2.2.1.2 Literature Search and Selection Criteria

After formulating the research questions, a search strategy is defined to identify the relevant literature. In alignment with the research objective, four digital databases are employed for the literature search: Scopus, ScienceDirect, IEEE Xplore, and Web of Science. Finally, a methodology is developed to identify the pertinent publications on certain predefined inclusion and exclusion criteria. A preliminary search equation consisting of ideas and concepts directly associated with machine tending is outlined below.

Preliminary search equation: (“manufacturing”) AND (“machine tending”), OR (“machine vision”, “computer vision” OR “artificial intelligence” OR “machine learning” OR “deep learning” OR “robot”). Based on this search equation, a total of 1303 articles were selected (Scopus – 363, IEEE Xplore – 414, ScienceDirect – 31, and Web of Science - 495). The distribution of the number of publications obtained through the preliminary search equation is illustrated in Figure 2.1.

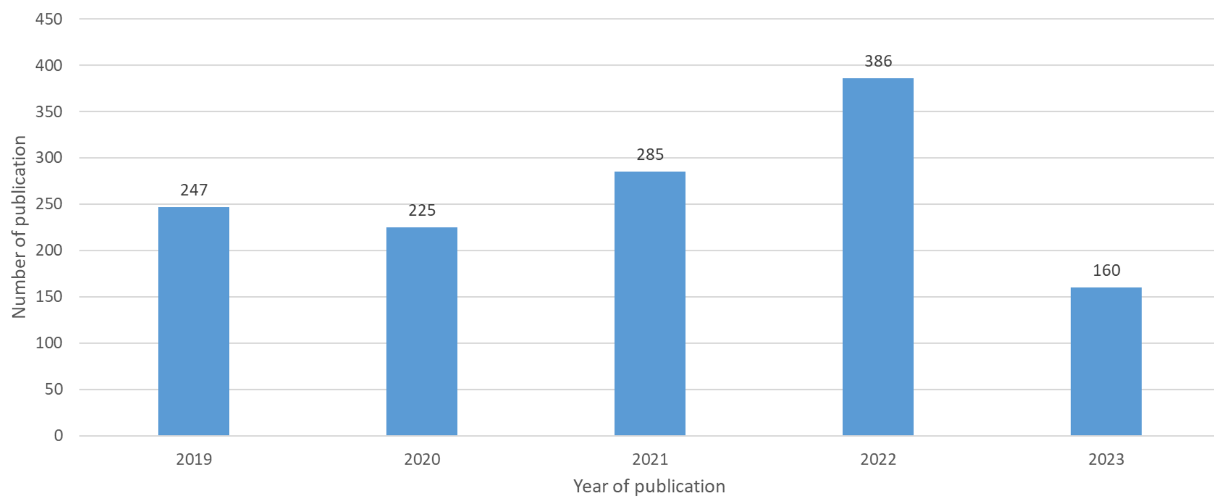

Figure 2.1: Research work obtained from the initial search equation by publication year.

2.2.2 Evaluation Process

Upon establishing the review protocol, the succeeding phase in the systematic analysis is the evaluation process. PRISMA (Preferred Reporting Items for Systematic Reviews and Meta-Analysis), an evidence-based strategy [21], is utilized for this study, as depicted in Figure 2.2. This procedure delineates the number of records identified, screened, assessed for eligibility, and eventually included in the systematic review, as explicated below:

- Identification: This literature review explores four online databases, leading to the manifestation of duplicate records. These duplicates need to be identified and consequently removed from the systematic review. Following the initial search, a total of 1019 records were identified.

- Screening: Once removing duplicated records, the title and abstract of the papers are analyzed before reading the full article. In this step, a total of 722 records were excluded based on titles and abstracts.

- Eligibility: In this stage, a thorough reading of the full text is performed to assess the eligibility of a total of 297 records are assessed for eligibility concerning the objective of this study, as they either failed to answer the research questions, or the full text was not available. Out of these, a total of 252 papers were excluded.

- Inclusion: After identifying a total of 45 eligible articles to be relevant for this literature review from the previous steps, an additional 5 papers are added to the study through a cross-reference approach, adding up the total number of publications for meta-analysis to 50.

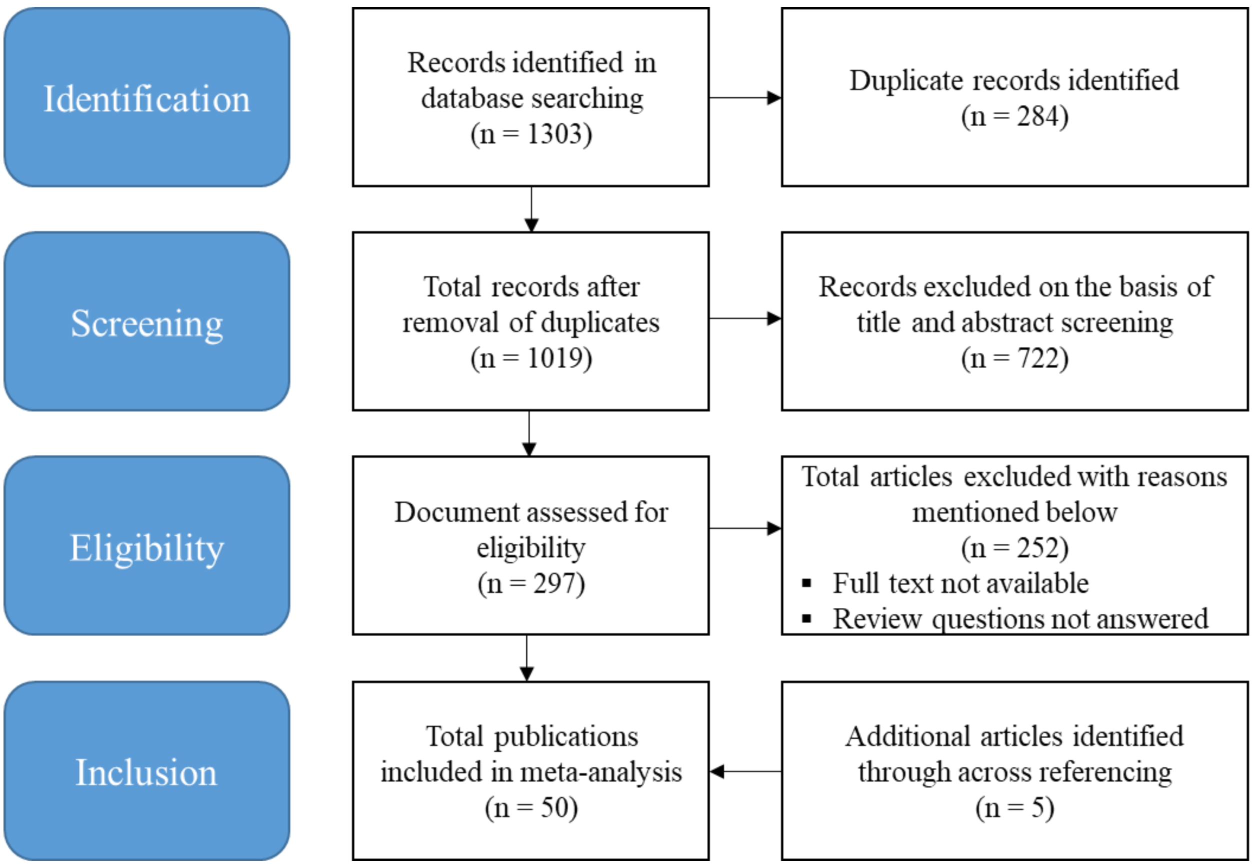

Figure 2.2: Adapted PRISMA approach for systematic literature review.

2.3 Results of Literature Review

This section presents the analysis results of implementing emerging technologies in various machine-tending tasks and provides some critical insights, contributing to answering RQ1 and RQ2. As shown in Figure 2. 3, the majority of the included studies engage with four different types of machine-tending tasks:

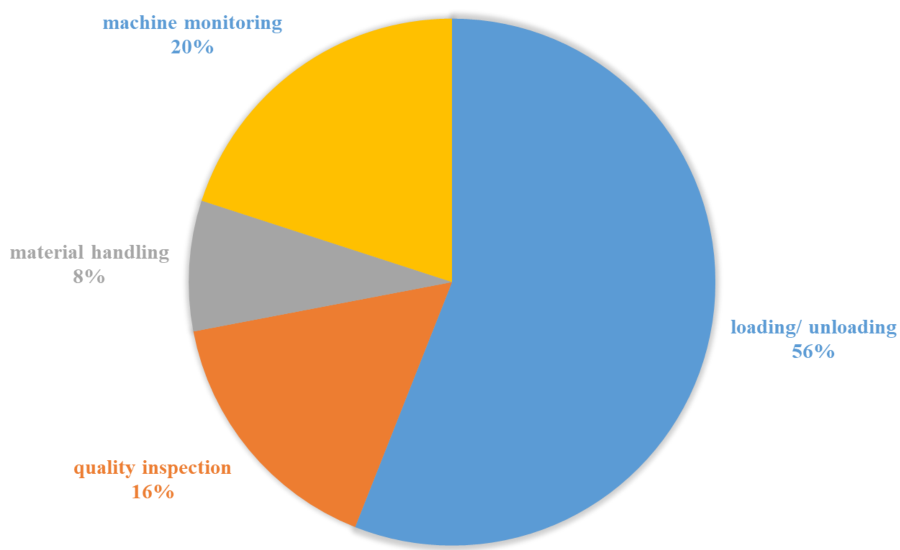

Figure 2.3: Distribution of selected literature by different types of machine-tending tasks.

i) loading and unloading, also named pick-and-place or part feeding. In [22], a UR5 collaborative robot is used to manipulate manufactured parts for loading and unloading in production factories. In the system, two RGB-D cameras are integrated, where one is mounted on the end-effector of the robot arm for objection detection, and the other is used for pose estimation of detected parts. The example is shown in Figure 2.4.

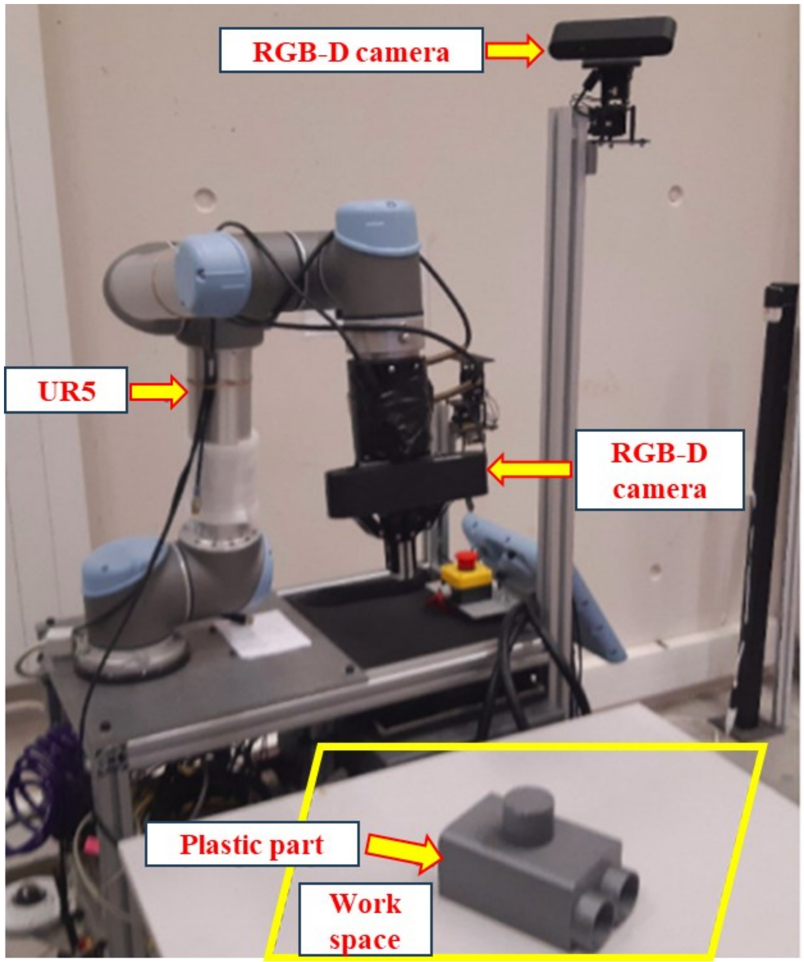

Figure 2.4: The example of part pick and place, adapted from [22].

ii) quality inspection. In [23], a UR5 collaborative robot with a multi-functional gripper machine-tending system is designed to pick the manufactured part, and check the quality of the products through the implemented vision system based on the proposed convolutional neural networks. This example is presented in Figure 2.5.

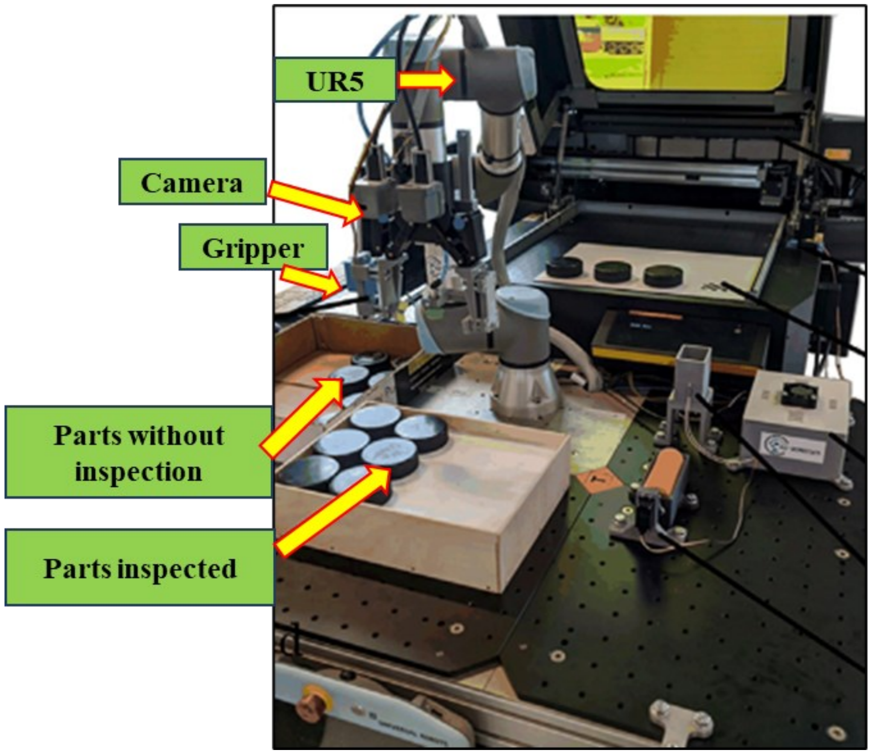

Figure 2.5: The example of part quality inspection, adapted from [23].

iii) machine monitoring. For example, in [24], a KUKA robot arm with a smart vision system is used for machine monitoring. The system configuration is shown in Figure 2.6, where an RGB-D camera is installed on the robot arm to capture real-time photos inside the CNC lathe machine and monitor machine tool condition through the proposed anomaly detection method.

Figure 2.6: The example of machine monitoring, adapted form [24].

iv) material handling or parts transportation between stations. For instance, in [25], a KUKA autonomous mobile robot with a gripper and vision system attached to the manipulator end-effector is utilized to grasp the raw material from the local storage and transport it to the machine before loading operations.

According to the literature analysis, it is observed that a fully autonomous machine tending to replace a human worker to monitor the screen for emergency handling and instruction execution, such as cycle start and stop, is the area not covered by the current literature.

The major trends apparent following the systematic literature review, presented in Table 2.2, on emerging technologies in machine-tending applications are observed in topics such as robotics, artificial intelligence, and computer vision. Each of these technologies has been explored in the subsequent sections of this paper.

Table 2.2: Review of machine-tending systems with emerging technologies (integration of Robotics, CV and AI techniques).

Types of tasks	Robotics	CV application	Algorithms	Maturity level	Reference
Pick-and-place	Cobots	Object detection	SIFT and SVM	Prototype	[26]
Pick-and-place	RAS	Object detection	CNNs	Prototype	[27]
Quality inspection	RAS	Edge detection	Binarization and Perspective Projection	Prototype	[28]
Machine monitoring	RAS	Edge detection	Canny	Prototype	[24]
Quality inspection	AMR	Edge detection	Canny	Prototype	[24]

Loading and unloading	AMM	Object detection	3D point cloud	Prototype	[29]
Material handling	AMM	-	-	Conceptual	[15]
Machine monitoring	Cobots	Object detection and OCR	SiameseRPN, HOG, and RNN	Prototype	[30]
Pick-and-place	Cobots	Object detection and localization	YOLOv3	Prototype	[22]
Pick-and-place	RAM	Object detection and localization	YOLOv5	Prototype	[31]
Loading and unloading	Cobots and AMR	Object detection and localization	CNNs	Prototype	[32]
Loading and unloading	AMM	-	-	Prototype	[33]
Pick-and-place	RAM	Object detection and localization	YOLOv3	Prototype	[34]
Pick-and-place	AMM	Object detection	SSD	Prototype	[35]
Machine monitoring	Cobots	Tool wear detection	YOLOv4	Conceptual	[36]
Pick and place	Cobots	Object detection and localization	YOLOv5, CNN, and SVM	Prototype	[23]
Quality inspection	Cobots	Object detection	CNNs and SVM	Prototype	[23]
Autonomous charging	AMM	Object detection and localization	-	Conceptual	[37]
Pick-and-place	Cobots	Object detection and localization	-	Prototype	[38]
Pick-and-place	Cobots	Object detection	YOLOv5	Prototype	[39]

Pick-and-place	Cobots	Object detection	YOLOv4	Conceptual	[40]
Machine monitoring	-	Tool wear detection	CNNs	Conceptual	[41]
Loading and unloading	Cobots	-	-	Conceptual	[42]
Quality inspection	-	Object detection	Gaussian and Gabor Filter	Prototype	[43]
Pick-and-place	RAM	Edge detection/object detection	Gaussian, Canny, and CNNs	Prototype	[44]
Pick-and-place	RAM	Object detection	CNNs	Prototype	[45]
Material handling	AMM	Barcode recognition	Morphological, Gaussian, and Harris	Conceptual	[46]
Pick-an-place	AMM	Object detection	-	Prototype	[47]
Pick-and-place	RAM	Object detection	Binarization, and HOG	Conceptual	[48]
Pick-and-place	Cobots	Object detection	CNNs	Prototype	[49]
Quality inspection	-	Object detection	SVM	Prototype	[50]
Machine monitoring	RAM	Object detection and Speech recognition	SSD, HMM, and LSTM	Prototype	[51]
Material handling	AMR	Object detection	CNNs	Conceptual	[52]
Pick-and-place	Cobots	Object detection	SSD	Conceptual	[53]
Pick-and-place	RAM	Segmentation and Classification	FT, and BPNN	Prototype	[54]
Machine monitoring	RAM	Tool wear detection	Sobel filter	Prototype	[55]

Pick-and-place	AMM	Object detection and localization	R-CNN	Prototype	[56]
Pick-and-place	Cobots	Task planning	Q-Learning	Prototype	[57]
Material handling	AMM	Marker recognition	-	Prototype	[25]
Pick-and-place	Cobots	Object detection	YOLOv4	Prototype	[58]
Pick-and-place	RAM	Object detection	YOLOv3	Prototype	[59]
Loading and unloading	AMM	Object detection	-	Prototype	[13]
Machine monitoring	Cobots	OCR	CRNN	Prototype	[60]
Machine monitoring	-	Tool damage detection	CNNs	Conceptual	[61]

2.3.1 Robotics in Machine Tending

Robotics has been identified as an essential driving force in the evolution of Industry 4.0, contributing to the digital transformation of manufacturing processes and enabling the establishment of smart factories [62]. As sensor technologies and artificial intelligence evolve, their integration with robotics will continue to play a vital role in shaping the future of manufacturing and addressing the challenges faced by the manufacturing sector. Recently, robots equipped with advanced perception and control capabilities have been used for autonomous machine tending [63]. These robotic systems are competent in various tasks, including loading and unloading workpieces, changing tools or fixtures, monitoring machines, recognizing the operational status of machines, and performing quality inspections [64]. A synthesis of the selected literature reveals that robotic systems applied to machine tending can be broadly categorized into three types: robotic arm systems, collaborative robots, and autonomous mobile manipulators. A detailed explanation of these systems will be provided in the following sections.

2.3.1.1 Robotic Arm Systems (RAS)

Robotic arm systems, also called industrial robots or manipulators, have been a mainstay in the manufacturing sector for several decades, integral components in tasks ranging from assembly lines to welding, and machine tending [65]. One of the primary advantages of these robotic arm systems is their precision and consistency, which allows them to repeatedly perform tasks to a degree of accuracy that far surpasses human capabilities. Further, they can operate tirelessly over extended periods, leading to increased productivity and efficiency, particularly for repetitive tasks. They are also able to handle hazardous materials and work in environments that may be unsafe or unsuitable for human workers, thereby enhancing worker safety [66]. This feature is particularly beneficial to free up operators from low-skilled machine-tending tasks.

However, conventional robotic arm systems do present some limitations. For instance, they often require a dedicated, enclosed workspace due to safety concerns, consuming considerable floor space. They are also typically designed for specific tasks and lack the flexibility to adapt to changing production requirements without considerable reprogramming or physical modifications [67]. Additionally, traditional robotic arm systems often necessitate a high initial investment for their acquisition and setup. They require regular maintenance and may entail significant costs if any major repair or overhaul is needed. The programming and operation of these systems also typically require specialized technical skills, potentially adding to the operational costs [68].

2.3.1.2 Collaborative Robots (Cobots)

Cobots are designed to work alongside humans in a shared workspace. They can be programmed to perform machine tending tasks autonomously or under the guidance of human operators [69]. Cobots often incorporate safety features such as force-sensing technology or vision systems to ensure safe human-robot interaction. Compared with traditional robotic arm systems, Cobots can be seamlessly integrated into existing workflows due to their compact size and inherent safety features, which makes them highly adaptable and versatile in varying manufacturing environments by integrating with their abilities to be quickly reprogrammed and redeployed for different tasks. This feature enables Cobots to increasingly take over low-skilled manufacturing tasks and manufacturing facilities like machine tending, pick and place, and quality inspection [12]. Moreover, Cobots differ from their traditional industrial robots in their programming and operational simplicity. They often come with intuitive interfaces and programming methods, such as hand-guiding or visual programming, which enable non-expert users to set up and operate them without requiring extensive robotic training [11].

While Cobots have made substantial strides in increasing safety and interaction with human workers, mobility limitations exist. Cobots are usually designed to be stationary and are installed at a specific location within a production line [70]. They perform tasks within a defined working envelope and are generally not equipped to move around the workspace and work between workstations. This constraint limits their adaptability and versatility in dynamically changing manufacturing environments. It restricts them from performing tasks that require a high degree of spatial mobility or tasks that need to be done in various locations within the facility [71]. Although Cobots are designed for easy repositioning and can be manually moved between different workstations, this process requires time and human labor, affecting the efficiency of the production line [72].

2.3.1.3 Autonomous Mobile Manipulators (AMMs)

To overcome the limitation of Cobots, an emerging trend is the integration of Cobots with mobile robot platforms, named Autonomous Mobile Manipulators (AMMs) [73]. AMMs combine the benefits of Cobots, such as safe human interaction and versatile manipulation capabilities, with the autonomous mobility provided by Autonomous Mobile Robots (AMRs) [36]. The prototype of the AMM, used in our lab, is shown in Figure 2.7.

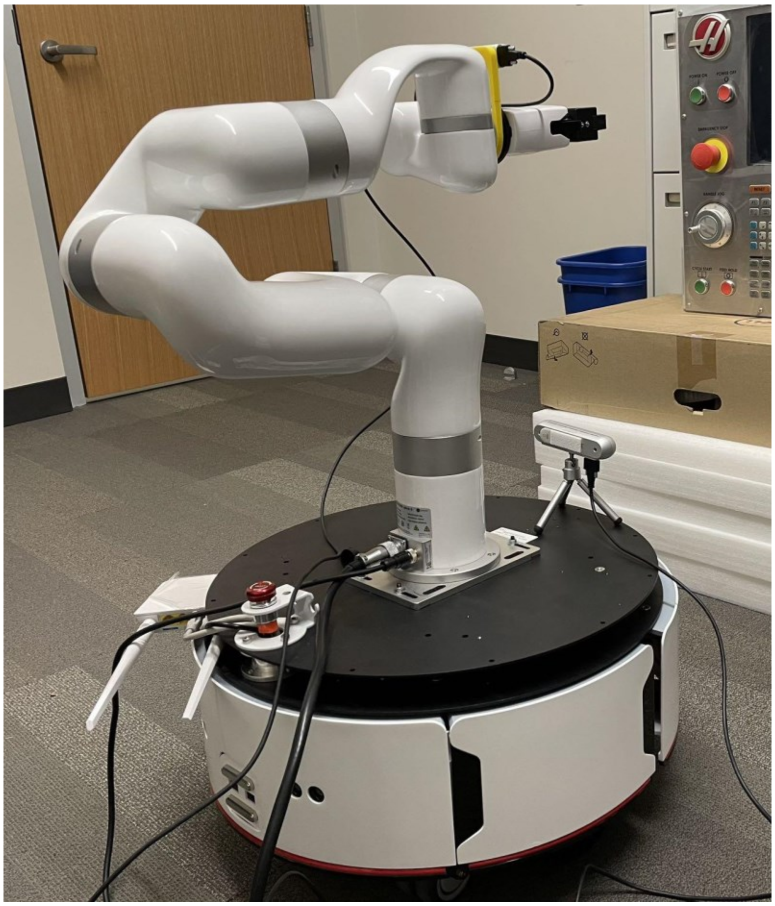

Figure 2.7: The prototype of the AMM.

The mobile platform is usually equipped with various sensors, such as LIDAR or cameras, to perceive the environment and avoid obstacles while moving within a workspace [74]. The manipulator mounted on the mobile platform is also equipped with different sensors, such as an end-effector camera, to facilitate and perform more complex tasks, such as object pick-and-place and transporting workpieces between different machines or workstations [15]. The main advantage of AMMs is their ability to bring manipulation capabilities to different locations within a workspace, expanding the scope of automation and providing the flexibility that stationary robotic systems cannot offer [30]. Therefore, they can autonomously move from one workstation to another, performing machine-tending tasks such as loading and unloading, picking and placing, and inspection. With the integration of advanced artificial intelligence (AI) techniques, AMMs are becoming increasingly capable of adapting to dynamic environments and performing complex machine-tending tasks with less human intervention or even further without human intervention. They can learn from the surrounding environment, make decisions based on real-time data, and improve performance over time [30].

However, it is noticed that AMMs are limited by their battery and operating time [75]. According to the analysis results of the literature, these systems are still in their early stages of development and adoption, and thus further research and development are needed to address the challenges associated with autonomous charging, dynamic task planning, and seamless interaction with tended machines in unstructured environments.

As technology advances, the adoption of AMMs is expected to grow, further transforming the manufacturing landscape. AMMs are also expected to play a crucial role in the next generation of manufacturing, driving the transition toward full automation and unmanned smart factories [72].

2.3.2 Computer Vision in Machine Tending

Computer vision (CV), which refers to machine vision in industrial applications, is a significant component of Industry 4.0 like other digital technologies [76]. It enables the CV system to sense and see the world through capturing real-time images or videos, image processing, and analysis of visual information [77]. Given its abilities and benefits, CV technologies can be applied to facilitate autonomous machine tending by increasing efficiency, reducing errors, gathering data, and robotic guidance [78], [79]. In addition, it may substitute for an absence of experienced workers to release them from some low-skilled machine-tend tasks such as quality inspection, processes monitoring, and part feeding. In machine-tending applications, the framework of a typical CV system is shown in Figure 2.8.

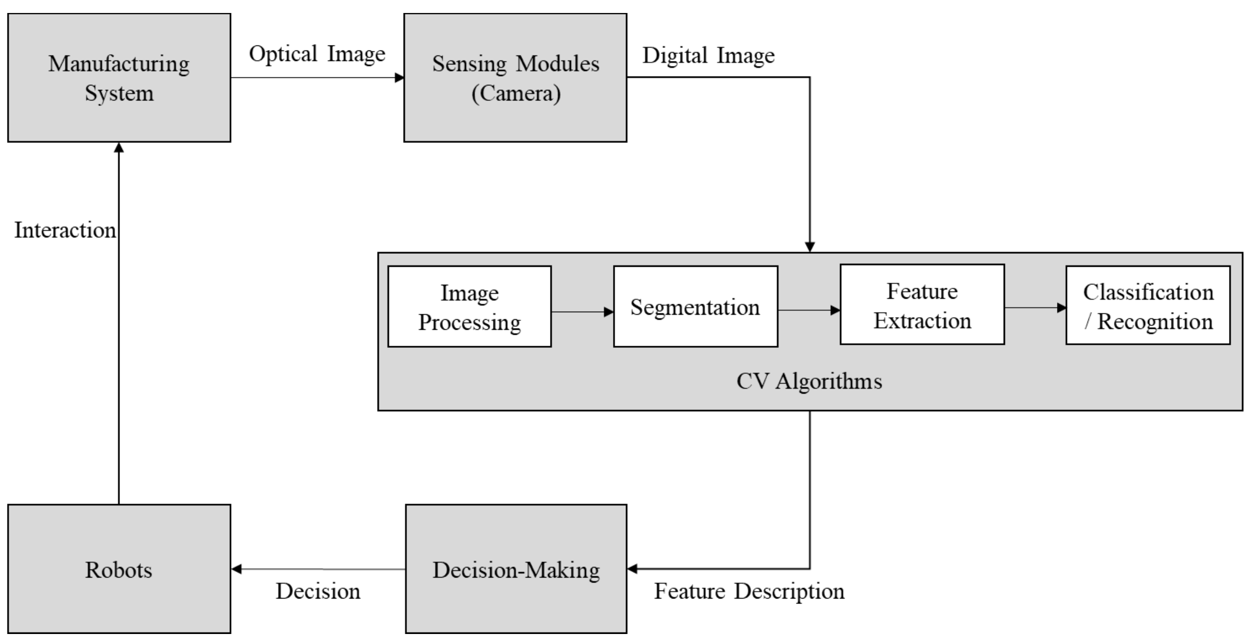

Figure 2.8: The framework of the manufacturing-oriented CV system.

2.3.2.1 Low-Level Image Processing

Image processing, also known as pre-processing, refers to the initial phase of image processing. It is the technology to transform a digital image using a series of mathematical operations, and the output can be either a processed image or a collection of characteristics or parameters related to the original image [80]. The primary objective of low-level image processing is to improve the quality of an image to facilitate the success of further procedures. This process is performed to enhance image quality, improve feature visibility, and reduce noise, which enables further accurate object recognition in machine-tending tasks such as loading and unloading, part identification, quality control, and machine monitoring. For image enhancement, some techniques such as Histogram, and Gamma Correction, can be used to enrich the visibility of features in an image [48]. In real-world applications, image blurring often occurs because of camera jitter. Therefore, image restoration is an important process to recover a captured original image. Median filtering, Gaussian smoothing, and Laplacian sharpening are used to highlight particular features or remove noise by emphasizing or de-emphasizing certain frequency ranges in an image [25], [43], [44]. The thresholding techniques convert grayscale images to binary images, making it easier to separate objects of interest from the background [28]. In addition, edge detection algorithms such as Sobel [55], and Canny [24] are applied to highlight the object boundaries within an image. Some morphological operations like dilation and erosion [60] can be beneficial to removing small imperfections or separating objects in an image.

However, these algorithms can be compromised in uncontrolled environments due to the variability in classifications, changing lighting conditions, and occlusion, making implementing low-level computer vision algorithms in practical machine-tending applications challenging. In summary, these low-level image processing techniques are extensively adopted in the preprocessing phase, serving as the foundation for more complex tasks. Improving image quality and extracting basic image features establishes the groundwork for high-level techniques to perform tasks such as object detection and localization in machine-tending applications.

2.3.2.2 Object Detection and Localization

Object detection and localization are key techniques in computer vision that have widespread applications in the realm of machine tending in the manufacturing sector. The main goal of object detection algorithms is to recognize the target object through real-time video frames through vision sensors built into the machine-tending systems, generating a bounding box and a class label for each detected object [81]. Object detection identifies the presence and location of multiple objects in 2D image space, while object localization takes object detection a step further localization by focusing on identifying the location of a single object in real-world spatial space [76].

This technology allows machine-tending systems to identify and classify different workpieces or tools based on their features. It is crucial for tasks such as accurate loading and unloading, sorting or packaging parts [27]. In addition, using stereo vision, structured light, or time-of-flight cameras to capture depth information to estimate the pose and position of objects in practical manufacturing environments [31]. This allows machines to handle parts of varying shapes and sizes and perform tasks in three dimensions, such as defect detection [43] in real manufacturing processes, with high efficiency and accuracy.

Although the applications using object detection and localization technology in machine tending are integral to enhancing automation and efficiency in the manufacturing sector, the success of these techniques heavily relies on the large amounts of high-quality data, which is adopted for training and fine-tuning of algorithms to accommodate specific tasks within dynamic, and real-world conditions.

2.3.2.3 *Optical Character Recognition (OCR)*

OCR is a computer vision technology that converts text in images, collected through cameras, into machine-readable and editable data. It enables machines to read and understand characters, numbers, or labels presented on workpieces or displays [82]. Therefore, OCR techniques can be utilized to accurately identify part numbers, serial numbers, or other important information, ensuring accurate detection and recognition in the machine-tending process. There are some applications for OCR in machine tending [83]. For instance, part identification can be used to read and identify characters or numbers engraved or printed on parts [84]. This can help in sorting, tracking, and managing parts during manufacturing [85].

Moreover, in quality control, OCR can be applied to verify printed labels, barcodes, or texts on manufactured products to identify misprints or errors, ensuring that only qualified products pass through the manufacturing line [86]. In addition, OCR could be utilized to monitor the work status of CNC machines by reading and interpreting commands on displays, which can improve productivity and enhance overall safety in autonomous manufacturing environments [60]. In conclusion, OCR technology has the potential to significantly improve the efficiency and accuracy of machine tending applications within the manufacturing sector.

Despite its benefits, it is noted that the implementation of OCR can be influenced by many factors in real-world scenarios, such as the quality of captured images, the instability of the vision system, and lighting conditions in the workspace [30]. Consequently, continuous advancements in OCR are essential to overcome these challenges and expand its implementation in practical machine-tending applications.

2.3.3 Artificial Intelligence in Machine Tending

Artificial intelligence (AI) involves the formulation of theory and the architecture of computer systems capable of learning and behaving as humans and performing tasks traditionally requiring human intelligence, such as sensory perception and decision-making [87]. AI, specifically in the context of machine learning (ML) and deep learning (DL), is regarded as one of the key drivers behind the digital transformation of the manufacturing sector [88]. These technologies have the potential to enhance manufacturing production, enhance real-time monitoring, optimize loading and unloading procedures, and improve quality inspection. Several intelligent machine-tending systems [13], [30], [64] have been developed by integrating ML and DL algorithms, and these techniques are explored in the next two subsections, providing a comprehensive understanding of their applications in machine tending.

2.3.3.1 Machine Learning (ML) in Machine Tending

Machine learning (ML) is a subfield of AI that allows systems to learn and improve from experience without being explicitly programmed automatically. The process of ML begins with observations or data, inclusive of examples, direct experiences, or instructions, to discern patterns and contribute to decision-making processes [88]. Machine learning (ML) techniques are broadly classified into three categories: 1) supervised learning which encompasses linear regression [89], Bayesian linear regression [90], and support vector regression (SVM) [26]; 2) unsupervised learning, including K-means clustering [91], and neural networks (NN) [61]; and 3) reinforcement learning such as Q learning [57]. In terms of machine tending, NN is commonly employed to identify defects in parts or to differentiate between diverse types of workpieces based on specific features, and Figure 2.9 shows the preliminary architecture of an artificial neural network (ANN). K-means clustering is often deployed to detect anomalies or machine faults. Reinforcement learning techniques such as Q learning [57] could be used in machine-tending instances where the robotic arm system needs to learn optimal strategies for parts transportation.

However, in many machine-tending scenarios, high-quality data can be scarce or expensive to acquire. As a result, transfer learning [61], a method involving the use of a pre-trained model as a starting point for a new machine learning task, can assist in time-saving and resource conservation when formulating machine learning applications for machine tending.

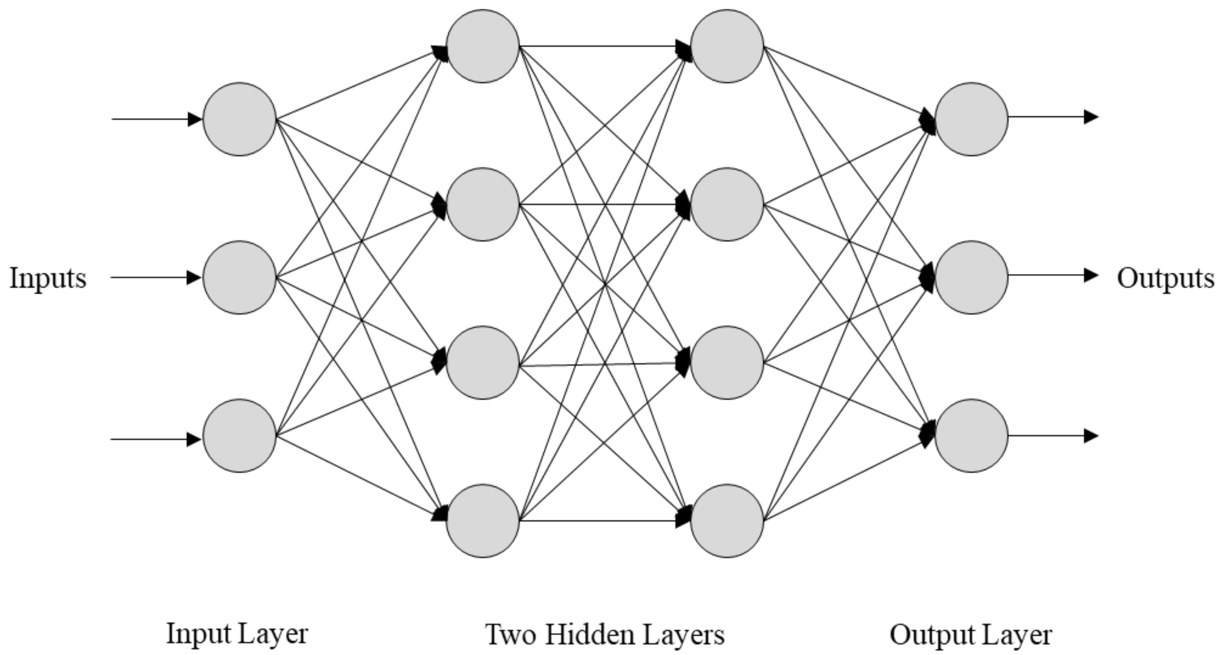

Figure 2.9: The architecture of the neural networks (NN).

2.3.3.2 Deep Learning (DL) in Machine Tending

Deep learning (DL) is the extension of classical ML, consisting of a deeper number of processing layers. The primary advantage of DL is featuring learning which involves the automatic extraction of high-level features from large datasets to assist in solving more complex problems efficiently and accurately [92]. Different DL algorithms have been used in machine-tending tasks such as convolutional neural networks (CNNs) [53], long short-term memory (LSTM) networks

[51], recurrent neural (RNN) networks [60], and generative adversarial networks (GANs) [93]. Figure 2. 10 presents an example of the architecture of CNNs. Given their proficiency in handling multi-dimensional and unstructured data, various CNNs such as region-based CNNs (RCNN), Faster-RCNN, Mask-RCNN, SSD, and YOLO have been employed in numerous machine-tending tasks. Especially, real-time object detection capability is very important for real-world machine-tending applications, and one-stage CNN-based methods such as SSD [35] and YOLO series [71], [72], are more popular to tackle these problems in terms of both efficiency and accuracy. Currently, the state-of-the-art real-time object detectors are mainly based on YOLO-based methods. RNNs are used when there is a need for the model to remember the context or sequence of events, therefore in machine tending, RNNs can be utilized for tasks such as predicting the tool lifespan or machine monitoring when a machine needs an emergency stop or other operations based on a sequence of sensor readings [30][60].

Given that data is scarce or expensive to obtain in many machine-tending scenarios, GANs [94] can be used to generate synthetic training data, which can be useful to get the pre-train model with good performance. In addition, the concept of deep reinforcement learning (combining neural networks with reinforcement learning) has gained prominence. It involves training a model to make a sequence of decisions to maximize a reward through trial and error. In machine tending, this can be used to optimize the path that a robot arm might take to pick and place objects.

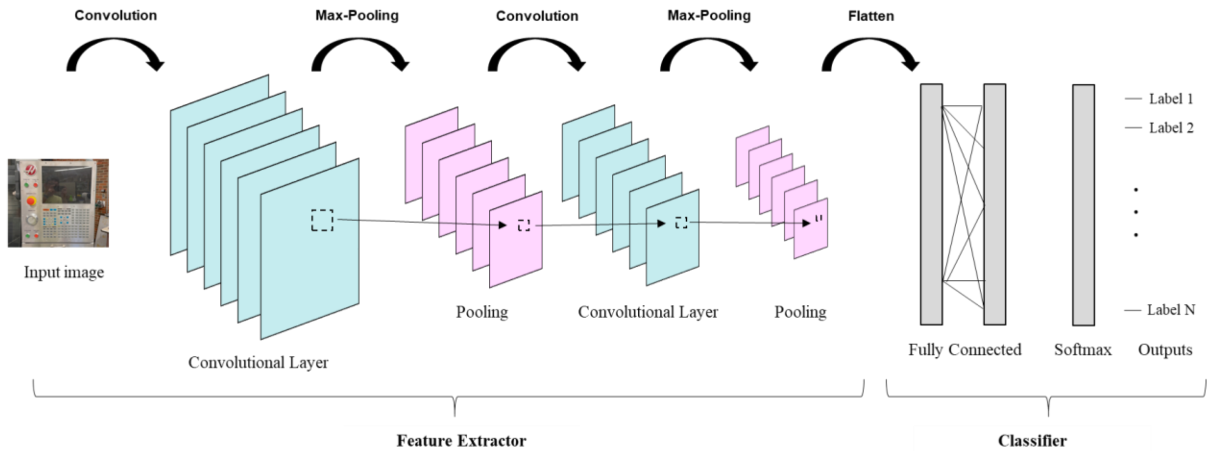

Figure 2.10: The architecture of a convolutional neural network (CNN).

2.4 Discussions and Future Trends

This section analyzes the selected literature and provides some critical insights to answer the RQ3 in terms of some current challenges and future trends to achieve fully autonomous machine tending.

2.4.1 Challenges

- *Robust perception systems:* To achieve efficient and error-free machine tending, a perception system involving object detection, recognition, localization and tracking capabilities is essential. Developing robust and reliable perception systems for recognizing and handling various objects, especially in dynamic and uncontrolled environments, remains a challenge [95].
- *Dynamic environment awareness:* Adapting to dynamic and changing environments, such as the presence of moving objects, shifting obstacles, or varying lighting conditions, poses challenges for machine tending systems. Therefore, developing algorithms and strategies

that can handle dynamic environments and make real-time adjustments is an ongoing research area [96].

- *Decision-making abilities:* To substitute human workers in a workspace, the ability to make task planning and generate execution strategies is important to the machine-tending systems because there are various uncertainties in real manufacturing environments, such as machine breakdowns. A big challenge, thus, is how to make decisions dynamically according to real-time visual information [97].
- *Runtime and autonomous charging:* The ability of autonomous charging for machine-tending systems is essential to achieve seamless manufacturing processes. The challenge for achieving autonomous charging is there is a lack of a robust and adapted navigation module to guide the machine-tending systems to the charging station in dynamic manufacturing environments with varying and complex illumination conditions [37].
- *Implementation:* Currently, computer vision and deep-learning algorithms utilized in machine tending are still relatively classic algorithms. The latest achievements of deep learning methods have not been quickly applied to the machine-tending applications facing real-world manufacturing environments. In addition, the majority of machine-tending applications in the literature are still in the prototype phase based on ideal problems and scenarios with a large amount of labelled data, which is very different from real-world issues. This also leads to the difficulty of implementing the latest research achievements into real machine-tending systems [97].
- *Data collection:* Data is one of the most important parts of machine learning and deep learning-based methods. Although more data can be obtained with the development of sensor technologies, collecting proper and high-quality data in some complex manufacturing

environments with poor lighting conditions is still a big challenge. The illumination condition is one of the biggest challenges for real-world computer vision systems [98]. Moreover, the reflections on the surface of the object are a significant challenge affecting the quality of images [99]. Additionally, camera jitter and shadows, while the mobile robot system is moving, are also big concerns [30].

- *Data labelling:* Although data can be obtained from real-world complicated manufacturing environments, the original data usually lacks the necessary labels which are important to machine learning and deep learning algorithms. Therefore, humans need to manually label large amounts of raw data, which is very time-consuming. The current challenge is a lack of effective algorithms for handling nonlabelled data, as well as lacking methods to automatically label original data [30]. To apply deep-learning-based methods to many different manufacturing scenarios for different machine-tending tasks, more efforts need to be made in this regard.
- *Benchmark datasets:* Although there are already some benchmarks for different real-world applications in the computer vision domain, such as COCO, MNIST, and ICDAR, it is difficult to directly apply these datasets to specific manufacturing cases because these benchmarks are mostly created for particular tasks such as vehicles detection, pedestrians' detection, car plate recognition and traffic sign detection [81]. Therefore, more benchmarks for various machine-tending tasks are needed to be created for CV and DL algorithms to be further continuously utilized in machine-tending applications.

2.4.2 Future Directions

Although current research observed in this chapter, presented in Table 2.2, seems to focus on all major topics in machine-tending applications, such as advanced robotics, smart sensors, computer vision techniques, and artificial intelligence, rising topics within the machine-tending field and potential collaborations between different research directions can be identified. Consequently, based on the research literature, and the above discussion of current challenges of machine-tending applications, this section proposes some potential future directions to facilitate and accelerate the implementation of advanced technologies in machine tending and broaden the research associated with machine tending in manufacturing.

2.4.2.1 Cognitive robotic machine-tending systems

By integrating artificial intelligence into conventional robotic machine-tending systems and learning a large set of manufacturing scenarios, cognitive robotic machine-tending systems can safely handle unusual situations without prior occurrences [100]. Cognitive abilities enable machine-tending systems to perceive, comprehend, and interact with working scenarios and the surrounding environment, which can bring advanced automation to numerous machine-tending tasks, including pick-and-place, quality inspection, and machine monitoring. Recent work can be found for different machine-tending tasks. For instance, an augmented reality-assisted gesture-based robotic system is proposed in [101] to assist workers with no coding skills to tackle pick-and-place tasks in which different-sized products are randomly positioned in an unstructured environment. In [13], a semi-autonomous mobile manipulator machine-tending system is developed to transport parts between different workstations through remote teleoperation. An intelligent robotic sorting system [31] is proposed to pick and sort the various industrial parts in an uncontrolled environment. In [16], a fully teleoperated mobile manipulator system is developed to execute both material handling and loading and unloading tasks. However, the integration of vision systems and robotics is still in the early stages for robotic machine-tending systems. With the continuous development and evolution of smart perception sensors and computer vision technology, researchers will need to focus on designing cognitive robotic machine-tending systems by implementing evolving technologies.

2.4.2.2 Real-time quality inspection and monitoring

Currently, prevalent methods for assessing product quality and overseeing machine conditions remain manual inspection, but it can be time-consuming, labor-intensive, high-cost, and occasionally, associated risks. However, computer vision, characterized by its low cost, reasonable accuracy, and high robustness, is witnessing a surge in its application within the machine-tending field, particularly for quality inspection and condition monitoring. Noteworthy achievements have been observed in recent literature over the past half-decade. For example, [23], [24], [28] developed vision-based methods for product quality inspection and quality control, and [30], [41], [60] proposed deep learning-based methods for machine tools detection or working status monitoring. A notable challenge, however, persists with deep learning-based architectures like CNNs, which present weak performance in detecting and recognizing tiny objects [81]. It is also observed that certain tasks may not be optimally addressed by deep learning-based methods. Traditional image processing techniques retain a significant role and can be combined with machine learning or deep learning techniques to substantially enhance overall performance [30]. As manufacturing tends to deploy vision systems and vision-based algorithms to substitute manual inspection, more studies need to be done by researchers to address such challenges, thereby benefiting the machine-tending process through the maturation and integration of computer vision techniques.

2.4.2.3 Imitation learning methods for robotic machine-tending systems

One of the primary future desires in machine tending is to mimic humans' behavior in given tasks and provide machine-tending systems cognitive competence to minimize or even completely replace human workers on shop floors. As such, imitation learning, which enables robotic machine-tending systems to acquire knowledge via perception, observation and capture of human demonstrative motions, is a significant research topic. In recent years, deep RL stands out as the predominant technique enabling robotic systems to comprehend the working scenario and has been widely applied to deal with intricate tasks [96]. For example, Chen et al. [57] proposed a meta-RL framework to improve the adaptability of robotic systems to new tasks by transferring the policies from previously learned task modules. In [102], a Relative Entropy Q-Learning (REQ) is proposed to execute a complex pick-and-place manipulation task. In [103], the authors present a multi-view unified RL framework to perform complex tasks. However, most of the studies are still in the conceptual stages and validated in robotic simulation environments, and thus, more robust imitation learning methods for robotic machine-tending systems to perform complex and long-horizon tasks in practical applications are required.

2.4.2.4 Task-oriented models for machine-tending tasks

Over the last five years, the interest in adopting and developing standard ML and DL-based models for specific machine-tending tasks has been steadily growing. Related research contributions can be found in industrial parts' detection, localization, pick-and-place, and transportation [22], [23], [31], [39], [58], [59]. Nevertheless, these frameworks require extensive training computations and fine-tuning of parameters to obtain desirable outcomes for particular tasks. Once optimal performance is achieved, employing these pre-trained models and their corresponding parameters in analogous scenarios through transfer learning becomes feasible. However, most current research is limited by two concerns: benchmark datasets for specific tasks and data labelling. Building benchmark datasets is one of the promising trends for developing task-oriented models, considering that public datasets are not always accessible [104]. In a relevant study presented by [94], GANs were employed to construct a synthetic defect dataset. Data collected via sensing devices in real manufacturing environments is unannotated raw data. The quality of such data can be compromised due to poor illumination conditions or issues related to reflections. Therefore, it cannot be used directly for further procedures. Addressing this problem necessitates the establishment of effective preprocessing mechanisms for large amounts of raw data and original images, including data cleaning and image preprocessing [77]. Moreover, the exploration of either automatic or semiautomatic data annotation methods is promising to reduce the time costs associated with building trusted labelled datasets [105].

2.5 Conclusions

Driving by the rapid development of advanced techniques and digital technologies in Industry 4.0, such as robotics, computer vision, and artificial intelligence, machine tending has developed towards digital transformation and full automation. To obtain a comprehensive understanding of the current situation, bottlenecks and perspectives of machine-tending applications in the present-day manufacturing sector, a systematic literature review is presented in this study. Based on the adapted PRISMA approach and review protocol, a total of 50 papers from the last five years were identified and analyzed. According to an extensive analysis of selected articles, three main prevailing technologies including robotics, computer vision and artificial intelligence were discussed. In addition, critical analysis is performed and several conclusions are drawn such as the majority of applications are still in the conceptual or prototype phases, and a framework to achieve fully autonomous machine tending by integrating emerging technologies lacks more research work and validation. Consequently, eight challenges and four potential future research directions are summarized to achieve further or complete autonomy in machine tending.

Chapter 3 : An Improved 2D Path Planning Algorithm for a Novel Self-adapting Intelligent Robotic Machine-Tending System

3.1 Chapter Overview

Recently, robots have been widely used in industry to perform repetitive, dumb, dangerous, dull, and dirty tasks. Therefore, industry robotics forms an essential part of the manufacturing backbone [106]. Task flexibility and robotic mobility are two main advantages that mobile manipulators bring to industrial applications. Furthermore, a mobile robot shall adapt to changing environments and perform a variety of tasks. It is essential to combine locomotion capabilities with manipulation abilities to extend the capabilities of future robotics. The concept of mobile manipulation refers to a robotic system that integrates a mobile platform, a robot arm, supported by a vision system. "Mobile manipulation" was introduced by Schuler in 1984 [107], where a robot arm is mounted on a mobile platform to execute tool handling and delivering duty in a workshop environment.

This chapter focuses exclusively on the machine-tending domain by developing an improved path-planning algorithm for our designed robotic system. Path planning is a purely geometric process that is only concerned with finding a collision-free path regardless of the feasibility of the path [108], [109]. Recently, sampling-based path planning (SBP) algorithms have received considerable attention due to their capability in complex and time-critical real-world planning problems [110]. SBP is unique in the fact that planning occurs by sampling the configuration space (C-space). In a sense, SBP attempts to capture the connectivity of the C-space by sampling it. This randomized approach has its advantages in terms of providing fast solutions for severe problems.

Furthermore, the computational capacity of the robot controller unit is limited in my applications; SBP is a better choice because of its lower computational cost.

Sampling-based planning was proposed to overcome the complexity of deterministic robot planning algorithms for a robot with six degrees of freedom [111]. The use of random computations to solve otherwise rather tricky problems has been immensely successful [112]. Both sampling-based planners and the success of random computations inspired the development of the Randomized Potential Planner (RPP) [113]. RPP used random walks to escape the local minima of the potential field planner. Later on, a planner based entirely on random walks, with adaptive parameters, was proposed [114].

The work of Barraquand and Latombe [115] paved the way for a new generation of path-planning algorithms that employ randomization. Arguably, the most commonly used SBP algorithms are the Probabilistic Roadmap Method (PRM) [116], [117] and Rapidly-exploring Random Trees (RRT) [118]. Several other algorithms were developed at the same time that outperformed RPP. The intuitive implementation of both RRT and PRM and the quality of the solutions lead to their widespread adoption in robotics and many other fields. Although the idea of connecting points sampled randomly is fundamental in both approaches, these two methods are different in the manner that they construct a graph connecting the points [119]. A comprehensive survey of work in SBP is presented [120]. The PRM algorithm has been recorded to be implemented well in high-dimensional state spaces. The PRM is created by curves or straight lines that enable the robot to go anywhere in its free space. The two well-known methods, named visibility graph (VG) and Voronoi diagram (VD) have achieved very good results for very different types of roads. However, the path generated by VG touches obstacles at the vertices or edges and thus is dangerous for the robot. Contrary, the solution paths based on VD are not optimal because they tend to maximize the distance between the robot and the obstacles.

RRT has received a considerable amount of attention, because of its computational efficiency and effectiveness and its ability to find a feasible motion plan relatively quickly, even in high-dimensional space [121]. In [122], RRT is implemented and combined with a slightly modified dynamic window method. RRT explores a robot working area by incrementally building a tree, creating new branches by generating points randomly and linking them to the closest location for which an obstacle-free path is obtained. A problem in RRT is that it produces paths with many branches in the workspace by using the randomized technique. To overcome this problem, an improved robot path-planning algorithm based on the RRT and quintic B-spline curves is proposed to generate collision-free and smooth paths for a novel self-adapting intelligent machine-tending robotic system.

The chapter is organized as follows: In Section 3.2, a novel self-adapting intelligent machine-tending robotic system is introduced. The principle of the algorithm is demonstrated in Section 3.3. In Section 3.4, the results and discussion are presented. Finally, Section 3.5 concludes the paper.

3.2 Self-adapting Intelligent Machine-Tending Robotic system

In this study, two self-adapting intelligent mobile robot systems, named the Husky mobile robot and the Ufactory mobile robot, are used to perform the developed 2D path planning algorithm in this chapter. The structure of these two mobile robot systems is shown in Figure 3.1 and Figure 3.2, respectively. .

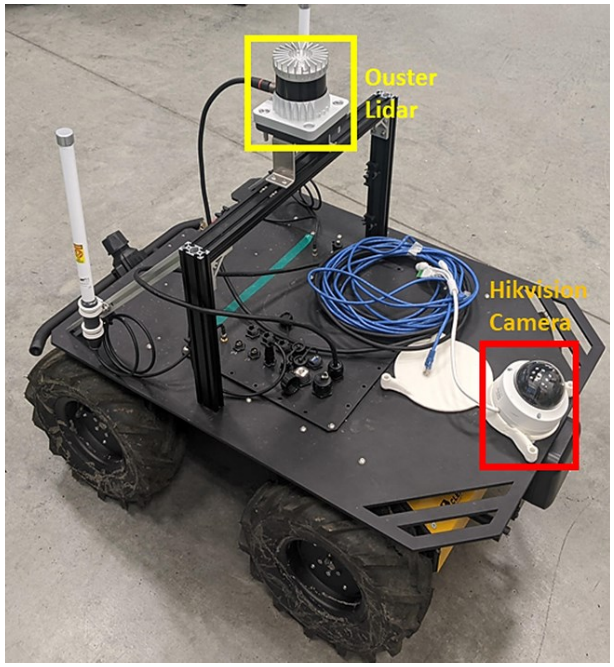

Figure 3.1: The structure of the Husky mobile robot system used in the study.

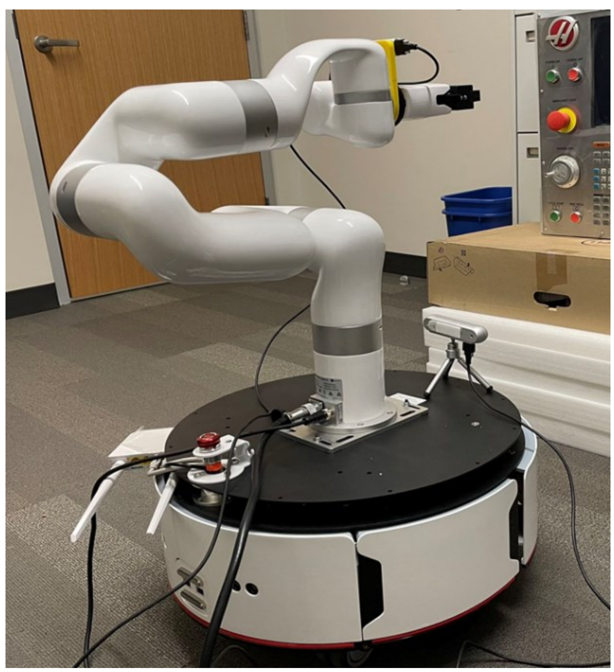

Figure 3.2: The structure of the Ufactory mobile robot system used in the study.

3.3 The Proposed 2D Path Planning Method

3.3.1 Principle of the RRT algorithm

To introduce the proposed improved 2D path planning algorithm, the necessary procedures of RRT must be defined. They are demonstrated as follows: (1) sampling: this procedure is used to select a configuration randomly and add it to the tree. The samples can be either in the free, or obstacle configuration space. This is the core of the RRT; (2) metric function: this procedure returns a value, or cost that indicates the effort required to reach C1 from C2, given two configurations C1 and C2 in the workspace; (3) nearest neighbors: this procedure uses a search algorithm which returns that closest node to the new sample. The value is based on the predefined metric function; (4) select parent: this procedure selects the nearest existing node to connect to the newly sampled node. That current node is considered parent; (5) local planning: given two configurations C1 and C2, this procedure attempts to establish a connection between them; (6) collision checking: it is mostly a Boolean function that returns success or failure when connecting two configurations C1 and C2. A connection is successful if it does not intersect an obstacle area [119].

The ability of RRT to explore free space in a workspace with obstacles is illustrated in Figure 3.7, where the vertical and horizontal axis represent the height and width of the 2D configuration space in meters. Two red crosses indicate the start and goal points, respectively. Black circles represent the obstacles in the space. The green straight line is the tree or path generated by the RRT algorithm. By updating new nodes and connecting new nodes by checking collision-free edges based on RRT, the shortest path from the start point to the target point can be calculated.

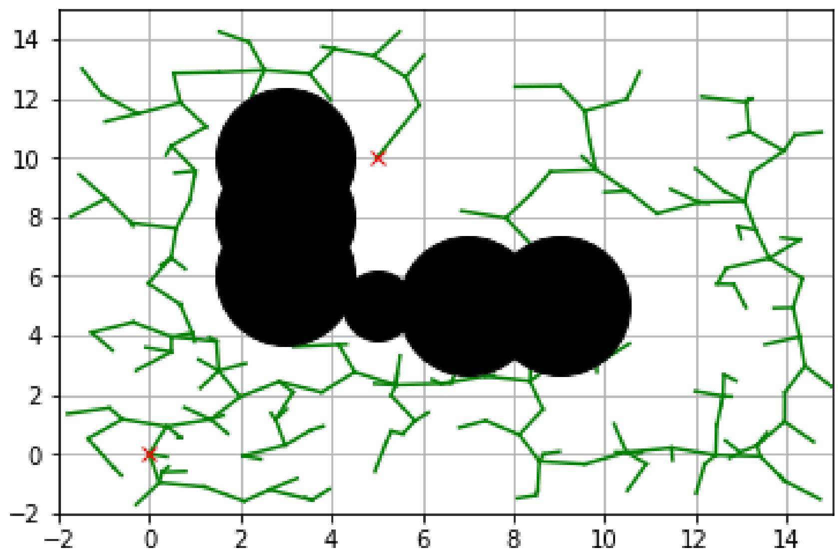

Figure 3.3: The example of RRT.

It is known that the RRT algorithm is intuitive to employ straight-line paths, and the tree produced contains too many branches. However, for most robotic systems this is not a feasible plan due to kinematic or dynamic constraints. Thus, we integrate the quintic B-spline curves to help the robotic system generate a curvature continuous path in a practical workspace.

3.3.2 Path Generation using Quintic B-spline Curves

In this part, we use a quintic B-spline curve [123] to interpolate points Q_k, k = 0…n. The quintic B-spline curve is presented as follows:

$$S(t) = \sum_{i=0}^{m} P_i b_{i,5}(t) \tag{3.1}$$

Where P_i is the control point, m = n + 4, $b_{i,5}(t)$, i = 0, …, m, is a quintic B-spline basis function. According to the chord length method, if an interpolating curve follows very closely to the data polygon, the length of the curve segment between two adjacent data points would be very close to the length of the chord of these two data points. The length of the interpolating curve would also be very close to the total length of the data polygon. Then, we assign a parameter value τ_i to each data point Q_i. It is shown as follows:

$$\tau_0 = 0 \tag{3.2}$$

$$\tau_i = \tau_{i-1} + \frac{|Q_i - Q_{i-1}|}{L}, i = 1, \ldots, n \tag{3.3}$$

where L is the sum of the lengths of these chords:

$$L = \sum_{i=1}^{n} |Q_i - Q_{i-1}| \tag{3.4}$$

Therefore, the ratio of the chord length from the data point Q_0 to data point Q_k, denoted as L_k, over the length of the data polygon is

$$L_k = \frac{\sum_{i=1}^{k} |Q_i - Q_{i-1}|}{L} \tag{3.5}$$

For a clamped quintic B-spline curve, a knot vector for interpolation can be calculated without end conditions [124]:

$$t_0 = \cdots = t_5 = 0, t_{n+1} = \cdots = t_{n+6} \tag{3.6}$$

$$t_{j+5} = \frac{1}{5} \sum_{i=j}^{j+6} \tau_i, j = 1, \ldots, n-5 \tag{3.7}$$

The performance of the quintic B-spline curve to smooth a straight-line path is shown in Figure 3.8.

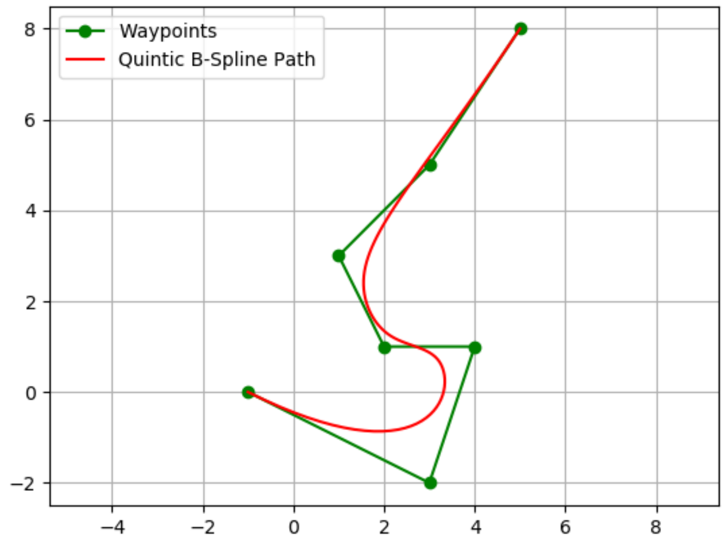

Figure 3.4: The example of the Quintic B-Spline Curve.

3.4 Results and Discussion

In this paper, five different scenarios (the goal point in each scenario in a different position) are executed to evaluate the developed path-planning algorithm. In each figure, the black circle areas represent the obstacles in the workspace. The green line represents the generated trees; the yellow line is the original path generated by the RRT algorithm; the blue line is the trimmed path, and the red is the improved smooth, continuous curve path produced by the developed 2D path-planning algorithm. The results are presented below:

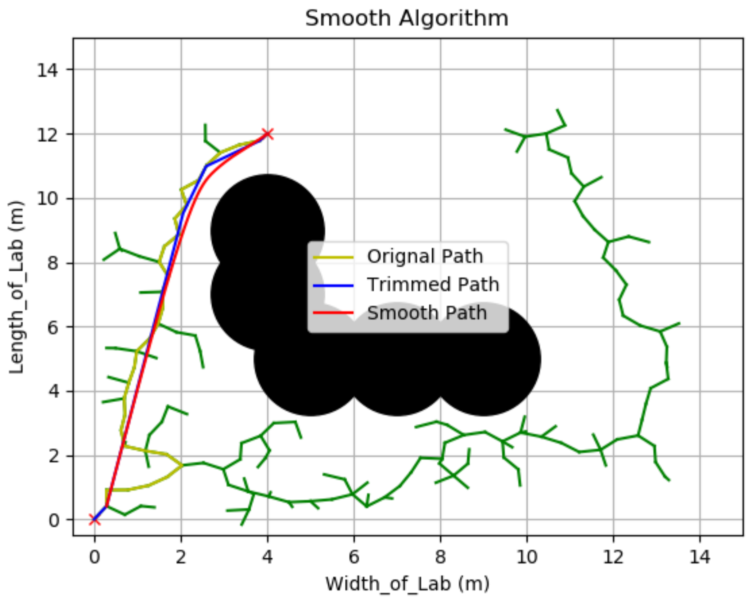

Figure 3.5: The example of scenario 1.

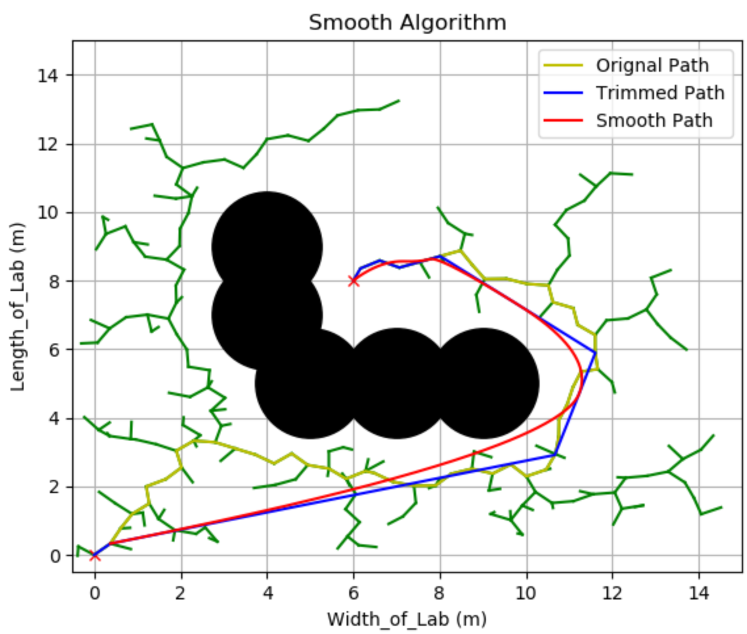

Figure 3.6: The example of scenario 2.

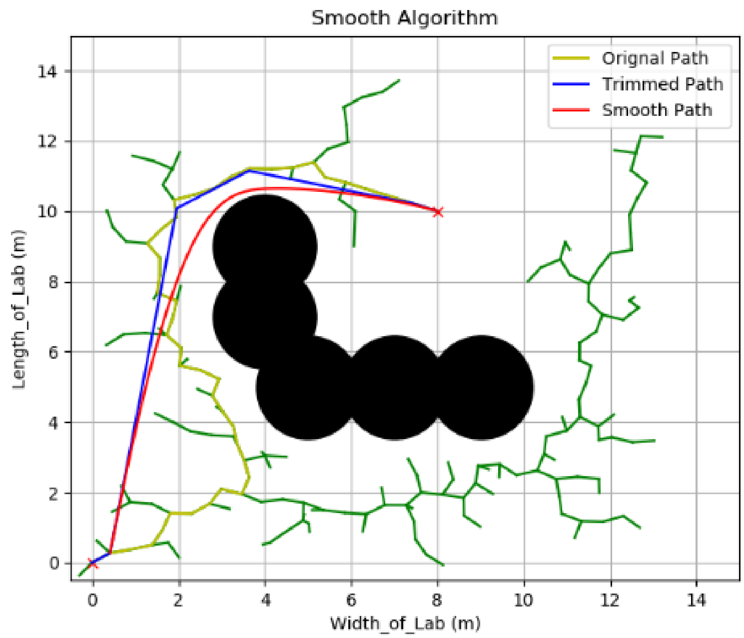

Figure 3.7: The example of scenario 3.

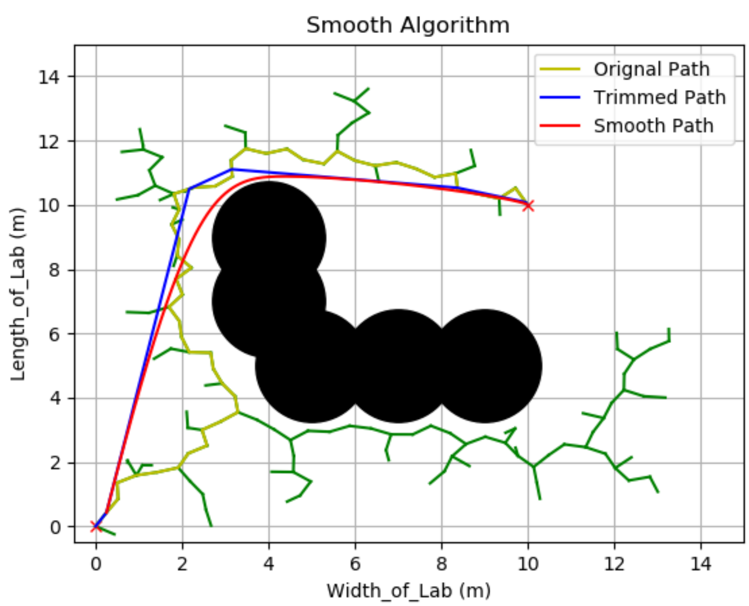

Figure 3.8: The example of scenario 4.

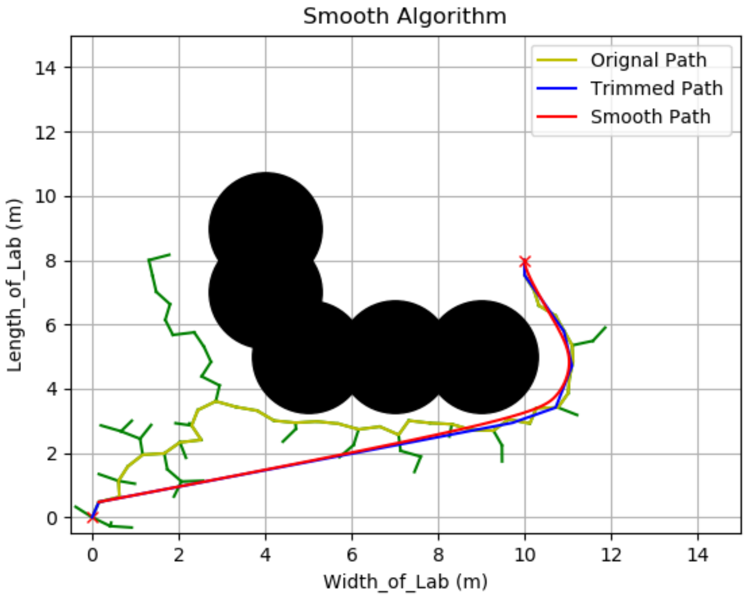

Figure 3.9: The example of scenario 5.

The path generated based on the proposed improved 2D path planning method can find a shorter and smoother path, which makes the path feasible for vehicle motion according to the kinematic and dynamic constraints. However, there still exists a chance that the vehicle might collide with obstacles. Therefore, continuous collision checking must be executed. A rectangular is placed in the simulation environment as a mobile robotic system, and detection of the collision between the four edges and the boundary of the obstacles is performed. When a path cannot be generated, the error message will be reported.

If there is no collision, the path will be generated successfully. Once the path is generated, the mobile robot system can move along the path at a constant rate by coinciding its orientation with the tangent of the curve. The algorithm is applied to three cases to evaluate the performance. The visualizations of these case studies are shown as follows:

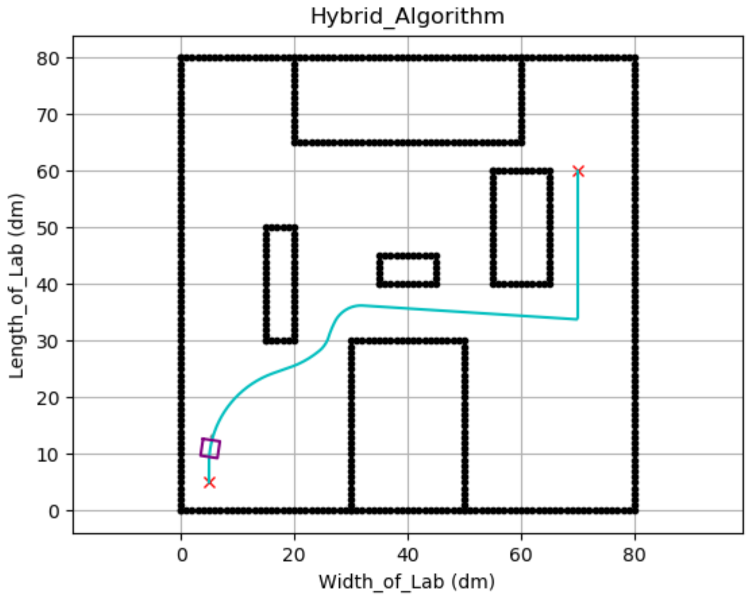

Figure 3.10: The example of case study 1.

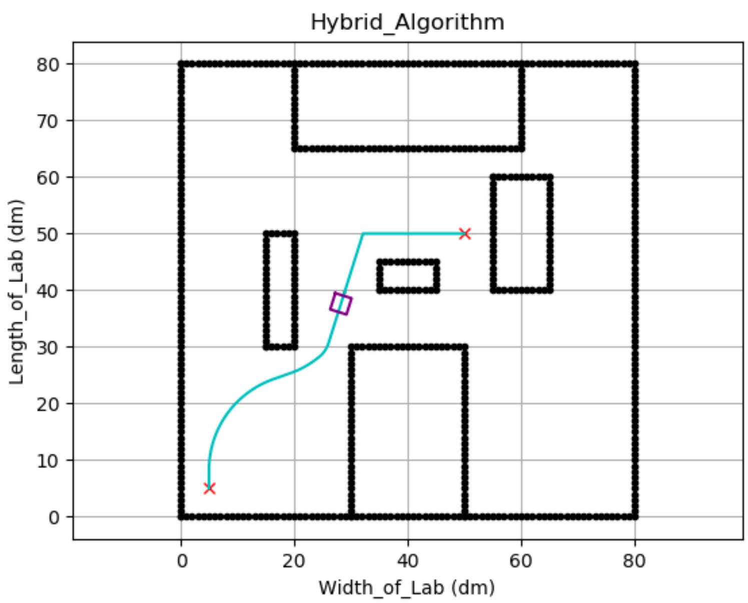

Figure 3.11: The example of case study 2.

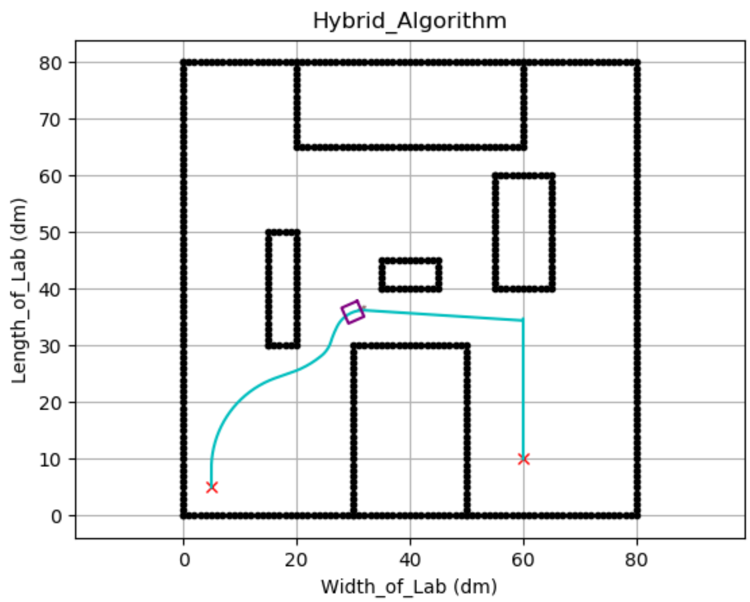

Figure 3.12: The example of case study 3.

The results of the three case studies show that the proposed path-planning algorithm has excellent performance for the mobile robotic system in a 2D environment from the above three cases. Since this algorithm is only used for generating the best path from one position to the other, it can be applied to various types of mobile robot systems without considering holonomic constraints.

3.5 Conclusions

In this chapter, we propose an improved path-planning algorithm based on a quintic B-spline curve to effectively generate smooth and collision-free paths for our designed Novel Self-adapting Intelligent Robotic Machine-Tending System. The advantage of our algorithm is that it makes the original path trimmed and smooth, so the mobile robotic system can continuously move following the kinematic and dynamic rules and slightly rotate its wheel, which extends the life of tires. However, this algorithm is not suitable for a very narrow path between obstacles.

In the next stage, this proposed path planning method will be used for autonomous docking and recharging for the mobile robot in real-world scenarios in manufacturing environments.

Chapter 4 : Vision and Lidar-based autonomous docking and recharging of a mobile robot for machine tending in autonomous manufacturing environments

4.1 Chapter Overview

The autonomous recharging process is an important part of a mobile robot's autonomous operation, allowing it to work continuously without any human intervention. Docking [125] can be understood as the navigation and localization of a robot toward the desired location. Docking requires an accurate pose estimate of the robot, often from a position close to the docking station through path planning [126]. Mobile robots are used across various applications [72], [127], [128], [129], [130], including surveillance, planetary exploration, dangerous environments, factory automation, search and rescue operations, indoor manufacturing environments, and so on. The role of mobile robots has become increasingly important for present and future applications. Thus, independent autonomous recharging becomes a fundamental requirement to ensure the autonomous operation of mobile robots in varying conditions. For a mobile robot to initiate the docking and recharging process, it needs to identify the charging station first, and then align itself with the charger autonomously by following a series of rotational and translational steps.

The location of the charging station (e.g., indoor or outdoor) plays an important role in the selection of sensors in a docking procedure. Outdoor environments are more complex, unpredictable, and dynamic due to the presence of moving objects and obstacles. Moreover, the performance of non-visual sensors like Lidars used for docking can be depreciated due to outdoor weather conditions such as snow, dust, and fog [131]. Based on the sensor implemented, the autonomous docking techniques in literature are divided into three categories: i) infrared (IR) sensor-based methods [132], ii) computer vision [133], and iii) laser-based approaches [134]. To receive IR signals properly, the IR receiver needs to be implemented at a specific location on the mobile platform, which limits the mechanical design of the mobile robots. Computer vision and laser-based techniques such as object localization and Lidar-based approaches are the most commonly used methods to solve odometry-related problems. However, both techniques have their respective limitations and benefits. Although Lidar [135] can extract different features of the environment, not being affected by the change in lighting conditions, and obtain more accurate range measurements than the camera, Lidar data is sparsely distributed and has a limited visibility range. Furthermore, its operations are based on collecting large amounts of data, which requires more computational power than the camera.. However, a standard camera without a 360-degree view has a limited visibility angle, resulting in a blind spot [133].

To overcome the challenges of conventional methods, the combination of different sensors has been investigated for years, and recent research has proved that the fusion approach yields better performance than a single-sensor method [135]; the limitations of the IR-based, laser-based, and vision-based methods for autonomous docking and recharging methods were overcome by combining multiple sensors. In [136], an attempt has been made to integrate a camera and IR sensor with laser range finders to improve the reliability of the autonomous docking process. In [137], a vision-based autonomous docking and recharging approach was applied to a security robot. An artificial landmark was installed on top of the charging station at the same height as the camera to assist the robot in detecting and locating the charging station area. The rotational and translational errors were compensated by using a virtual spring model motion control approach. The model [137] assumed that the robot and the charger could be connected by a virtual spring, and the compliant forces in the direction of the translation deformation and bending determined the motion control. However, the vision-based docking approach is prone to calibration errors, as demonstrated in [138], where a Faster R-CNN algorithm was used to detect arbitrary visual markers. The pose of the mobile robot was estimated using the solvePnP algorithm that related the 2D-3D point pairs. However, the solvePnP algorithm gave systematically inaccurate pose estimates in the x-direction and hence, proved to be ineffective for docking. Laser range finder techniques usually detect the charger based on a uniquely manufactured shape of the charging station to distinguish it from surrounding objects. One such example is the V-shaped recess on the MiR (mobile industrial robot) [32] by Fetch Robotics which required the charger to be placed separately from any laser-height obstacles to successfully detect the contour of the charger by the laser range finder. However, the requirement of a special shape adds to the charger station fabrication costs and limits mobile robots' practical application in unstructured environments. To solve this problem, a self-adhesive reflective tape is used to help the robot identify the charger as reported in [32]. With this reflection detection technique, the charger was easily distinguished from other similar objects in an unstructured environment that was verified by extensive experiments. Moreover, Lidar can be used for obstacle detection and avoidance, navigation, and pose estimation of the mobile robot without the use of additional hardware. In [139], a multi-sensor fusing method uses intensity and range data fusion with a covariance intersection approach to estimate robot pose during docking and recharging. Using the inverse perspective projection method, an artificial landmark was employed as a visual cue on the charging station to be identified by the robot. Then based on the laser range data, the geometrical relationship between the robot and charger station was estimated precisely with the covariance intersection method. Furthermore, in [136], ultrasonic sensors and IR sensors were integrated with a wheeled robot to assist in autonomous docking and recharging. Ultrasonic sensors were used to detect the charger and estimate the distance from it.

At the same time, the IR sensors were employed to adjust the orientation of the robot and achieve the exact position for the docking. In [140], automated guided vehicle (AGV) autonomous docking was investigated in an unstructured environment with human presence. An autonomous docking technique was implemented with a non-visual sensor like Lidar and AprilTag for charger detection. A deep learning network was used to detect and recognize humans and objects. Practical experiments verified that the AGV could co-exist with humans and perform autonomous docking in unstructured environments. With the development of deep learning techniques, deep learning-based approaches perform better in autonomous docking applications. In [140], the MobileNetv2-SSDLite deep learning framework was adopted to detect and recognize the specific person in the human-robot collaborative environment. Once the particular human is identified, the robot system can achieve automatic docking to the target person based on LiDAR and the RGB-D camera. In [26], Faster-RCNN is adopted to detect arbitrary dynamic obstacles. Given that high-resolution images from the camera can provide rich information, in [141], [142], authors proposed a fusion method to make use of images from a camera to enrich the raw 3D point clouds from LiDAR. The sparse convolutional neural network is adopted to predict the dense point clouds to enrich the raw point clouds and then employed to execute LiDAR SLAM. In [142], the Faster-RCNN model with a MobileNetv3-Large FPN backbone is used to identify the charging station. It has been proven that it can distinguish the charging station from other surrounding objects in most scenarios.

Based on previous studies, the autonomous docking and recharging process becomes more reliable and repeatable with a multi-sensor fusion approach in both structured and unstructured environments. However, the IR sensors require specific configurations such as signal receivers which is inconvenient and high costs [132]. Therefore, most current fusion methods consider combining the Lidar sensor with computer vision techniques because of their low cost and nondestructive abilities. However, computer vision techniques, especially deep learning-based object detection, require a large amount of proper task-oriented high-quality data for training and tuning to achieve the desired performance [95]. The changing lighting conditions and the shaking of the camera on mobile robots can also affect the performance of deep learning-based object detection models [143], which makes it difficult to implement computer vision techniques solely in real-world manufacturing applications.

Considering the aforementioned challenges, this chapter aims to develop a Lidar-camera data fusion method for mobile robots to achieve autonomous docking and re-charging in a complex manufacturing environment. This chapter contributes to the transition of state-of-the-art real-time object detection methods from general public datasets to real-world manufacturing tasks by combining deep learning-based techniques to identify the charging station in a complex manufacturing environment, followed by a Lidar-based approach to localize the detected wireless charger and dock the mobile robot to it for recharging. In this chapter, an indoor manufacturing environment with an enclosed space where the wireless charging station is set is considered to implement the docking procedure. The proposed method is analyzed and discussed based on the autonomous docking and recharging of a Husky robot by Clearpath Robotics. A YOLOv7-based method is used to detect the charging station for the robot to navigate to the desired location. Afterward, the Lidar sensor is used, integrating with the detected results and vision data, to determine the distance from the charger and side wall to achieve correct pose estimation and then successfully dock the robot to the charging station. The proposed method can be easily adapted to different types of wireless chargers and locations in a complicated manufacturing environment. The distance data between the Lidar and the camera can be calibrated to get ac-curate alignment and pose estimation.

The remainder of this chapter is organized as follows: Section 4.2 will present the related work; Section 4.3 will describe the mobile robot system used in the case study; Section 4.4 will explain the proposed method in detail; Section 4.5 will demonstrate the experiment results; and Section 4.6 will present the discussion and conclude this chapter.

4.2 Related Work

In this section, the recent docking and recharging methods for mobile robot systems in the manufacturing field based on Lidar and computer vision techniques are presented. Fan et al. [128] proposed a vision-based docking and recharging method that can be applied in a warehouse environment. This method used the AprilTag for the detection and identification of the robot's pose. It achieved around 97.33% docking success rate. In [138], authors proposed a Faster RCNN model to detect and localize the designed markers mounted on the docking station, combing with the solvePnP algorithm to navigate the mobile robot in a ROS simulation environment. This model has achieved an accuracy of 96.3% based on thirteen testing images. The detector takes around 35ms to process each image. Song et al. [140] adopted a single shot detector (SSD) to identify moving people and then dock to the target person for human-robot collaborative tasks in an unstructured environment. In [142], the SSD was developed to detect the charging stations in obstacle-free scenarios. This method can achieve a performance of 99.8% for successful docking to the charger. It takes an average of 12s to complete the docking procedures based on their designed scenarios.

Although these methods have made great contributions to autonomous docking and recharging applications, some limitations are observed. Most methods are evaluated in a simulation or laboratory environment instead of a manufacturing environment. In addition, two-stage deep learning models such as Faster RCNN, are inefficient compared to one-stage real-time models. Considering these limitations, the state-of-the-art real-time deep learning-based model, YOLOv7, is developed to distinguish and identify the target wireless charger from a complex manufacturing environment, integrated with a proposed Lidar-based approach to achieve efficient, low-cost, and robust docking and recharging tasks.

4.3 System Description

The autonomous mobile robot is shown in Figure 4.1. A Husky UGV field search robot by Clearpath Robotics is used to implement the Lidar-vision-based docking method and conduct autonomous charging experiments in indoor manufacturing environments. Figure 4.1 shows the Husky robot installed with an Ouster Lidar sensor and the Hikvision camera. The specification information of the Ouster Lidar and Hikvision camera is presented in Tables 4.1 and 4.2.

Table 4.1: The specifications of the Ouster Lidar.

Specifications	**Value**
Max range	200 m
Vertical field of view	$45°$
Channels of resolution	128
Max points	5.2 M/second
Max frame rate	20 Hz
Operating temperature	$-40℃ - 60℃$

Table 4.2: The specification of the Hikvision camera.

Specifications	**Value**
Max resolution	3840×2160
Vertical field of view	$56°$ to $17°$
Horizontal field of view	$108°$ to $30°$
Focal length	12 mm
Frame rate	30 fps
Max range	29 m

A ROS software development platform is used to program the docking process using the 3D Lidar sensor and control the robot's motion through the docking steps.

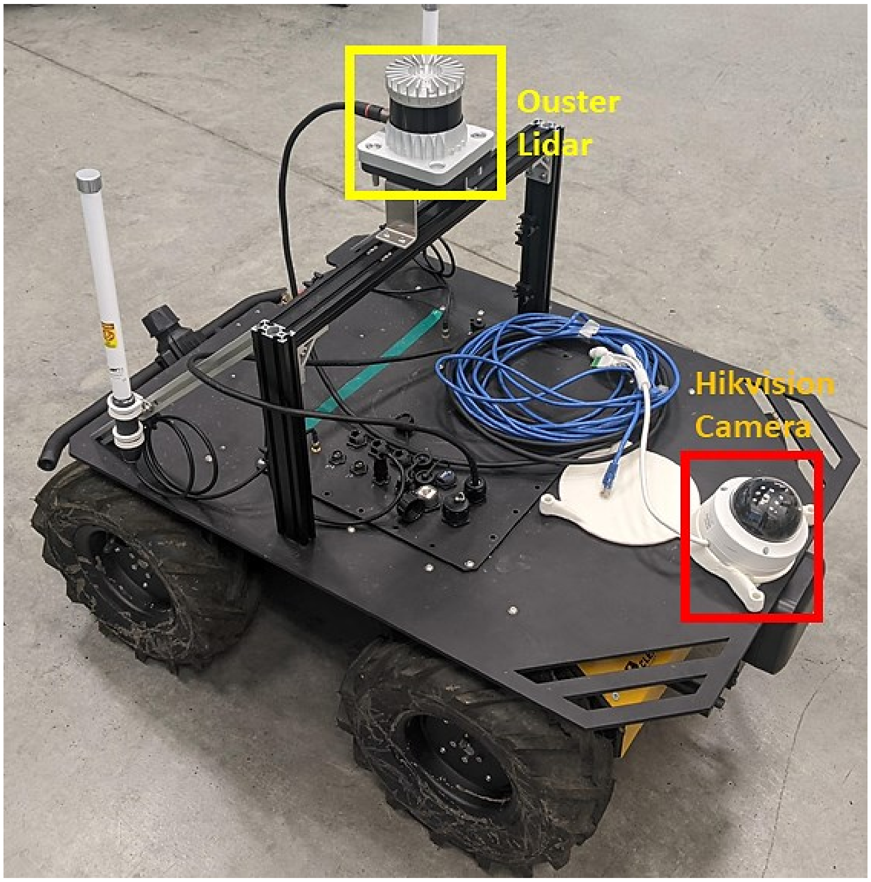

Figure 4.1: Husky robot setup with Lidar sensor and the Hikvision camera.

The wireless charging station used in the case study is presented in Figure 4.2, which is installed inside a custom sized modular structure. A ramp door placed in the front allows the robot to come out from the docking station to run missions and return for recharging as necessary.

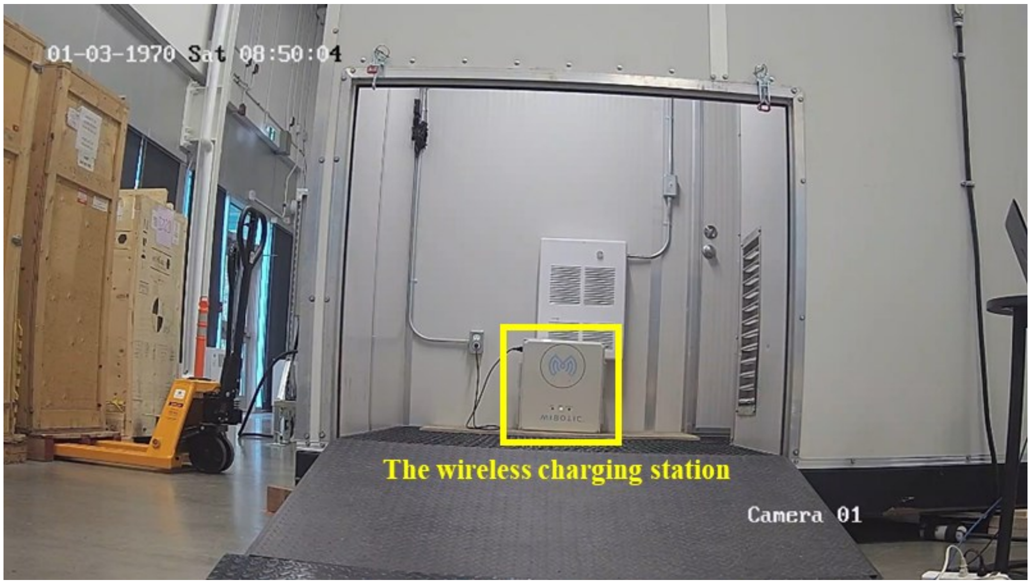

Figure 4.2: The charging station used in this study.

4.4 Proposed Autonomous Docking and Recharging Method

This section proposes a vision and lidar-based autonomous docking and recharging approach. The proposed method consists of three main steps: i) data collection, which adopts a Hikvision Camera and Ouster Lidar to capture the RGB images and depth in-formation respectively, ii) a deep learning-based object detection method, using the YOLOv7 model as the core architecture, to recognize the charging station in the manufacturing environment, iii) a Lidar-based approach to adjust the pose of the mobile robot and then dock it to the detected wireless charger. The flowchart of the proposed method is presented in Figure 4.3.

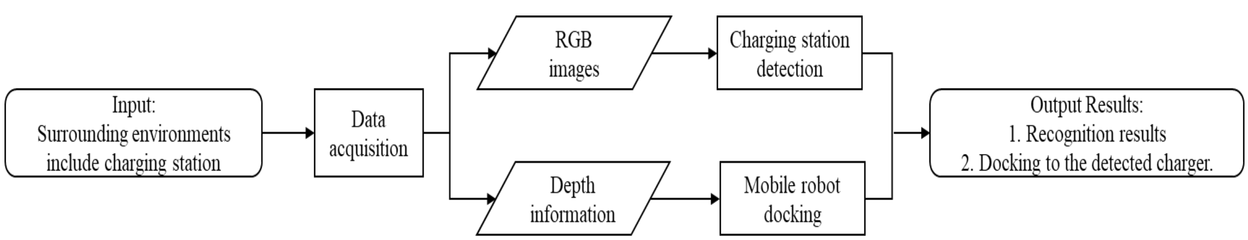

Figure 4.3: The block diagram of the proposed docking and recharging method.

4.4.1 YOLOv7 Architecture

YOLOv7 is a one-stage model and the latest algorithm for real-time object detection so far, and it performs well in both speed and accuracy [144]. The architecture of the proposed charging station detection method based on YOLOv7 is presented in Figure 4, which is composed of three main components: backbone, neck, and head. The convolutional back-bone module adopted the Darknet-53 [145] to extract image feature maps from the input image and transfer them to the neck layers. In the neck module, the Feature Pyramid Network (FPN) [146] is used to enhance the feature maps. These maps are then combined, fused, and passed to the subsequent layers. Finally, the head network predicts the bounding boxes and classes of the objects.

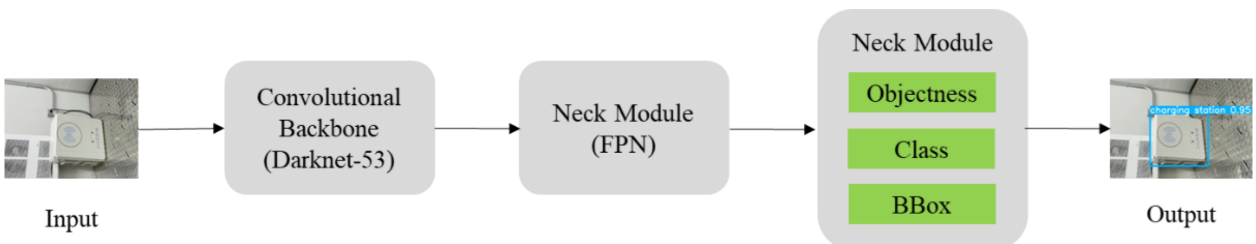

Figure 4.4: The flowchart of the charger detection method.

YOLOv7 adopts a developed Extended Efficient Layer Aggregation Network to improve inference efficiency. This network can quicken learning ability without disturbing or changing the original gradient propagation path. In addition, a novel scaling method, named corresponding compound model scaling, is proposed to address the issue of a larger width output of the computational block by directly scaling the depth of the con-catenation-based model. Moreover, several techniques have been used to improve inference accuracy while keeping low training costs. These techniques, called Bags of Freebies (BoF), include planned re-parameterization, dynamic label assignment, and batch normalization. After thoroughly investigating the re-parametrized convolution, the author demonstrated increased model accuracy when using the RepConv without an identity connection. Furthermore, batch normalization integrates the mean and variance of the data to adjust the bias and weight of the convolutional layer, which can immediately impact the training process by utilizing a higher training rate and faster convergence.

According to [147], YOLOv7 optimizes the inference process and improves detection accuracy and speed compared with other existing real-time object detection methods because of its more advanced network structure and training strategies. However, it has not been used yet in autonomous docking and recharging domains. In this chapter, YOLOv7 is adopted as the backbone architecture to detect and recognize the charging station.

4.4.2 Lidar and Vision Data Fusion Method for Autonomous Docking and Path Planning

In recent research, Lidar sensors and cameras are commonly used together in autonomous driving applications because a lidar sensor can collect 3D spatial information. In contrast, a low-cost camera captures the appearance and texture of the corresponding area in 2D images. Therefore, the fusion of Lidar and the camera data can improve object detection performance. Lidar-camera calibration estimates a transformation matrix that gives the relative rotation and translation between the 2D coordinates by the Hikvision camera and 3D spatial coordinates by Lidar as demonstrated in Equation 4.1 – 4.5 [148]. By coinciding the camera coordinate system with the global coordinate system, the transformation matrix can be derived in Equation 4.6. Then, 3D coordinates of the charging station can be calculated by Equation 4.7 – 4.9 based on the predicted bounding box in the image domain.

$$z_c \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} \frac{f_x}{dx} & 0 & u_0 \\ 0 & \frac{f_y}{dy} & v_0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} R_z R_y R_x & T \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} \tag{4.1}$$

$$R_x = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\theta_x & -\sin\theta_x \\ 0 & \sin\theta_x & \cos\theta_x \end{bmatrix} \tag{4.2}$$

$$R_y = \begin{bmatrix} \cos\theta_y & 0 & \sin\theta_y \\ 0 & 1 & 0 \\ -\sin\theta_y & 0 & \cos\theta_y \end{bmatrix} \tag{4.3}$$

$$R_z = \begin{bmatrix} \cos\theta_z & -\sin\theta_z & 0 \\ \sin\theta_z & \cos\theta_z & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{4.4}$$

$$T = [t_x \quad t_y \quad t_z]^T \tag{4.5}$$

$$z_c \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} \frac{f_x}{dx} & 0 & u_0 \\ 0 & \frac{f_y}{dy} & v_0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} X \\ Y \\ z_c \\ 1 \end{bmatrix} \tag{4.6}$$

$$X = \frac{u - u_0 \cdot z_c \cdot dx}{f_x} \tag{4.7}$$

$$Y = \frac{v - v_0 \cdot z_c \cdot dy}{f_y} \tag{4.8}$$

$$Z = z_c \tag{4.9}$$

where u and v are 2D coordinates from the camera; u_0 and v_0 is the origin of the camera coordinate system; X, Y and Z are the 3D global coordinates from the Lidar; f_x and f_y are the focal length along x and y directions. $R_z R_y R_x$ and T are the rotation matrix from the camera coordinate to the global coordinate, respectively. And z_c is the distance between the detected object and the camera. The illustration of the transformation process is presented in Figure 4.5.

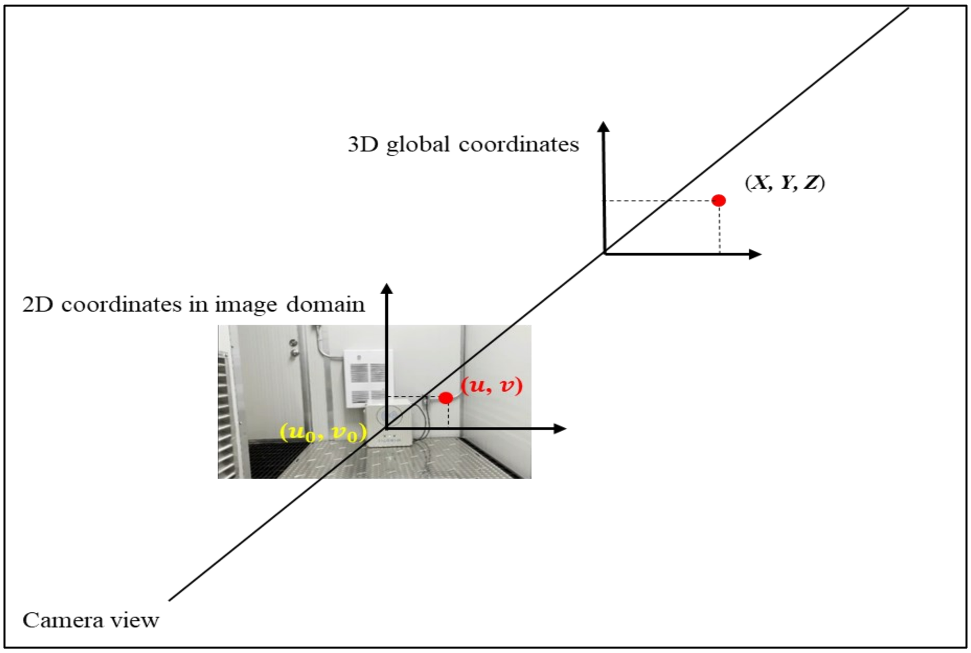

Figure 4.5: The illustration of the transformation process.

An ouster Lidar sensor is utilized to calculate distances from the robot frame of reference to the side wall and the depth or the distance to the charger. It is assumed that the charging station is enclosed within walls to simplify the pose estimation of the robot for the docking process. Two scenarios are considered for the Lidar-vision docking method implementation which is docking in an environment with only one charger and with three chargers as shown by the gazebo virtual environment setups in Figure 4.6.

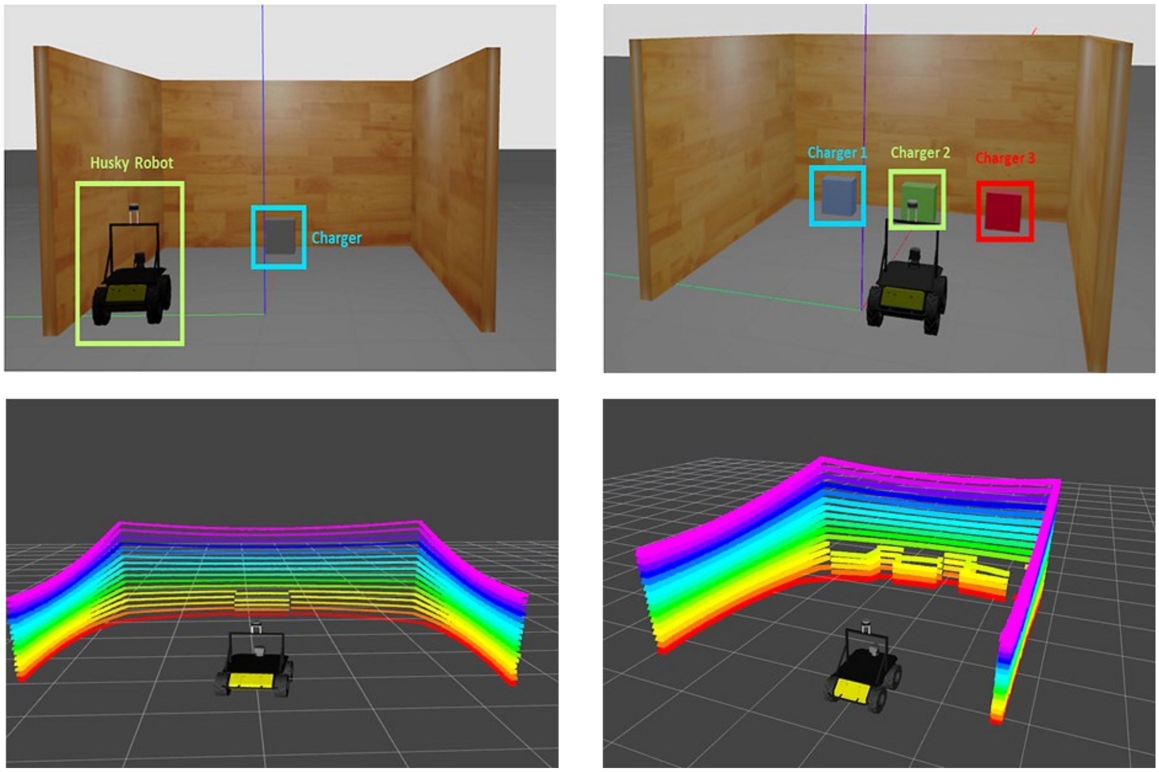

Figure 4.6: The docking station gazebo virtual environment setup with one charger (top-left) and three chargers (top-right), and Rviz lidar point cloud visualization for one charger (bottom-left) and three chargers (bottom-right).

In the case of three different chargers, the vision-based method will aid the robot in identifying the correct charger and autonomously dock with it. Rviz software is used to visualize lidar point cloud data of the charging stations for both one charger and three charger setups as demonstrated in Figure 4.6. The pose estimation and navigation for docking are applied with the lidar sensor data based on the information depicted in Figure 4.7. After the correct charging station is identified with the developed YOLOv7-based method, the lidar point cloud data is filtered to obtain two diagonal and two straight lines called: Front_laser, Back_laser, Wall_laser, and Charger_laser. Based on this information, a series of rotations and linear motion can be applied to the robot to move it to the desired location in front of the charger. The pseudo-code algorithm to implement the Lidar-based docking procedure is described in Table 4.3:

Table 4.3: Pseudo-code algorithm to implement the Lidar-based docking procedure.

State 1: Robot straightening
Initialize: Front_laser, Back_laser, Charger_laser, and Wall_laser
If (Front_laser – Back_laser) > 0 then rotate clockwise until Front_laser = Back_laser
else if (Front_laser – Back_laser) < 0 then rotate anti-clockwise until Front_laser = Back_laser
If Wall_laser > Known distance
Change state to 3
else if Wall_laser < Known_distance
Change state to 2
State 2: Robot turning left if to the right of the charger
Turn the robot anti-clockwise until Back_laser = Wall_laser
Then change state to 4
State 3: Robot turning right if to the left of the charger
Turn the robot clockwise until Front_laser = Wall_laser
Then change state to 4
State 4: Robot's linear motion
Move the robot in a linear motion until Wall_laser = Known_distance
Then change state to 5
State 5: Robot straightening the second time
If (Front_laser – Back_laser) > 0 then rotate clockwise until Front_laser = Back_laser
else if (Front_laser – Back_laser) < 0 then rotate anti-clockwise until Front_laser = Back_laser
Then change state to 6
State 6: Robot moving towards the charger
Move the robot in a linear motion until Charger_laser within 2 to 3 cm's away from the charger
Then change state to 7
State 7: Robot docking with the charger
Stop the robot's motion and change the status to docked

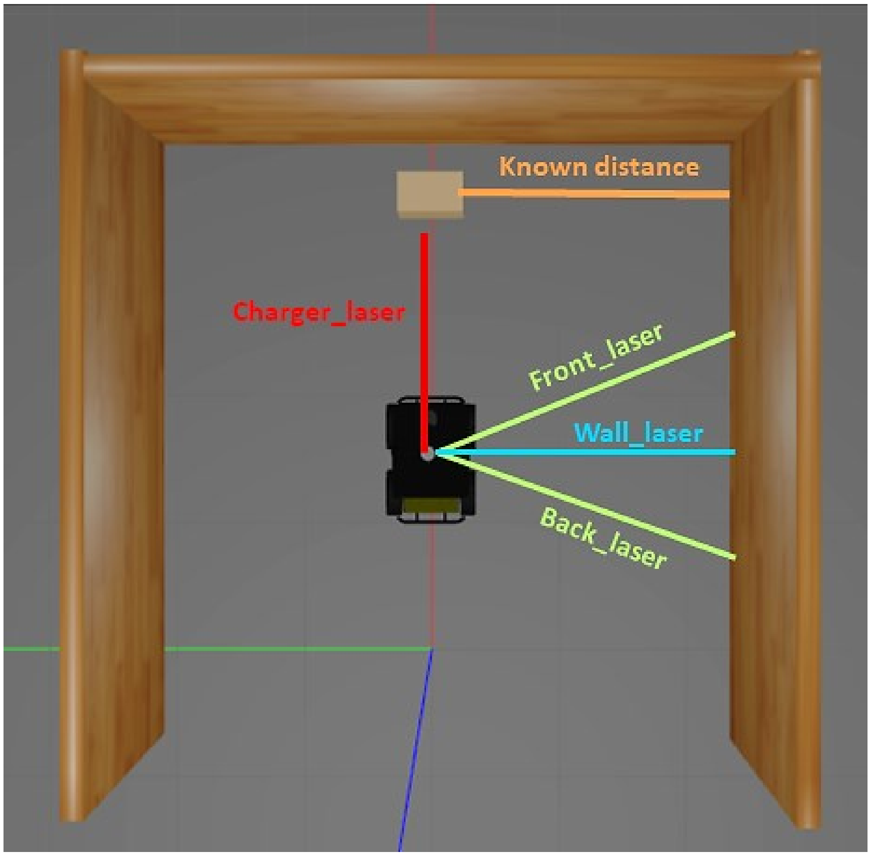

Figure 4.7: The Lidar-based docking method visualization.

The above algorithm was tested on the Husky robot and gave a fairly accurate pose estimation and localization for the docking, except for systematic errors based on the lidar data readings. The algorithm was tested for various initial poses and locations from the charger, a few of those case scenarios considered can be viewed in Figure 4.8. The known distance of the charger from the side wall can be determined using the vision-based method and matched with the Wall_laser to fuse the lidar-vision data. Moreover, once the robot is in the correct docking position for charging or close to the desired location, the lidar point cloud data and the camera-based 2D image can be calibrated to eliminate any errors and improve the pose estimation for autonomous docking of the robot.

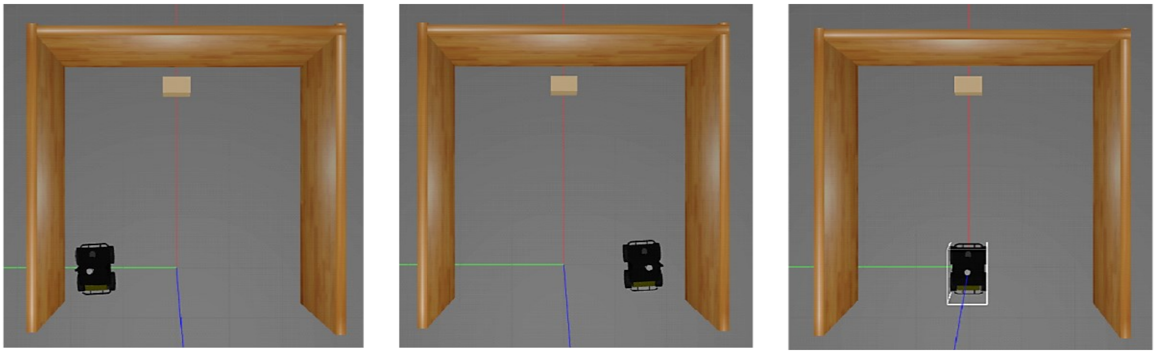

Figure 4.8: The Robot at different locations and orientations from the charger in a gazebo virtual environment setup.

4.5 Results and Discussion

4.5.1 Transfer Learning and Data Augmentation

Deep learning models frequently require extensive input images for the training process. However, gathering enough practical images for some applications can be difficult. Therefore, rather than building a model from scratch, transfer learning provides an alternative strategy for addressing this problem [55]. It adopts a pre-trained deep learning model as a template for another training task. The modified YOLOv7 model trained and tested on the Microsoft COCO dataset with the parameters was used in this study, significantly improving training efficiency. Due to the limited number of charging stations, images do not have extensive features. As a result, diversifying the training data is a common technique for improving generalization and reducing overfitting [22]. This study randomly introduces geometric distortions such as rotation, translation, scaling, and vertical flipping and image distortions such as Gaussian blur and noise.

4.5.2 Datasets Building

Since there are no public datasets for charging stations, used in the case study, a specific dataset was built for the experiments. The images of charging stations were collected through the

Hikvision camera mounted on the mobile robot. The created dataset has 240 images with a resolution of 1920 × 1018 pixels, shot from different angles and split into three sub-datasets: 160 training images, 40 validation images, and 40 testing images. These images in the dataset were annotated using LabelImg Software, which is an open-source annotation tool. The labelled images are shown in Figure 4.9.

Figure 4.9: The example of labelled images.

4.5.3 Training Environment and Parameters

The model for detecting and recognizing the dock and charging stations was trained and tested on a local Desktop with the specifications listed in Table 4.4. The pre-trained hyper-parameters for the dock and charging station detection are presented in Table 4.5.

Table 4.4: Training environment and specifications.

Specifications	**Value**
Operating System	Windows Server 2019
CPU	AMD Ryzen Threadripper 3970X 32-Core
GPU	NVIDIA GeForce RTX 3090
RAM	128 GB
CUDA Version	11.1
PyTorch Version	1.10.1

Table 4.5: Training Parameters.

Parameters	**Value**
Learning Rate	0.001
Learning Momentum	0.9
Batch Size	16
Epochs	100-300

4.5.4 Results and Analysis

4.5.4.1 Evaluation Metrics

This chapter adopted the mean average precision (mAP) as the evaluation metric. It is the area under the precision and recall (true positive rate) curve calculated by Equation 4.5-4.8 at different intersection-over-union (IoU) thresholds. mAP_0.5, at a 0.5 intersection-over-union (IoU) threshold, is commonly used as the evaluation metric. In addition, mAP_0.5:0.95, which is the average mAP over multiple IoU thresholds, can affect the modal with better performance. Therefore, both metrics will be considered in the training and testing procedures to evaluate the performance of dock and charging station detection.

$$IoU = \frac{Area\ of\ Overlap}{Area\ of\ Union} \tag{4.5}$$

$$AP = \int_0^1 (Precision\ \times Recall)\ d(Recall) \tag{4.6}$$

$$Precision = \frac{TP}{TP + FP} \tag{4.7}$$

$$Recall = \frac{TP}{TP + FN} \tag{4.8}$$

where *TP*, *FP*, and *FN* are true positive, false positive, and false negative of the predicted bounding box, respectively.

4.5.4.2 Results

Figure 4.10 depicts the training and validation loss for detecting the charging station. To optimize the proposed model, the loss function used in YOLOv7 needs to be minimized, it presents that around 300 epochs, the training and validation loss both decrease to a stable point with a minimal gap between the two final values. Figure 4.11 displays the model performance based on both performance metrics of the model in the validation. It can achieve about 99.4% mAP_0.5 and 86.5% mAP_0.5:0.95. During the training and validation, various epochs were tested. It can be observed that when the epoch is below 300, both training loss and validation loss continue to decrease at the end of curves, which indicates the proposed model can be further improved through further learning. However, with the epoch surpassing 300, the validation loss begins increasing, which leads to an overfit. Therefore, in the training and validation, the epoch of 300 times was chosen to obtain the optimized pre-trained model which can achieve the best performance.

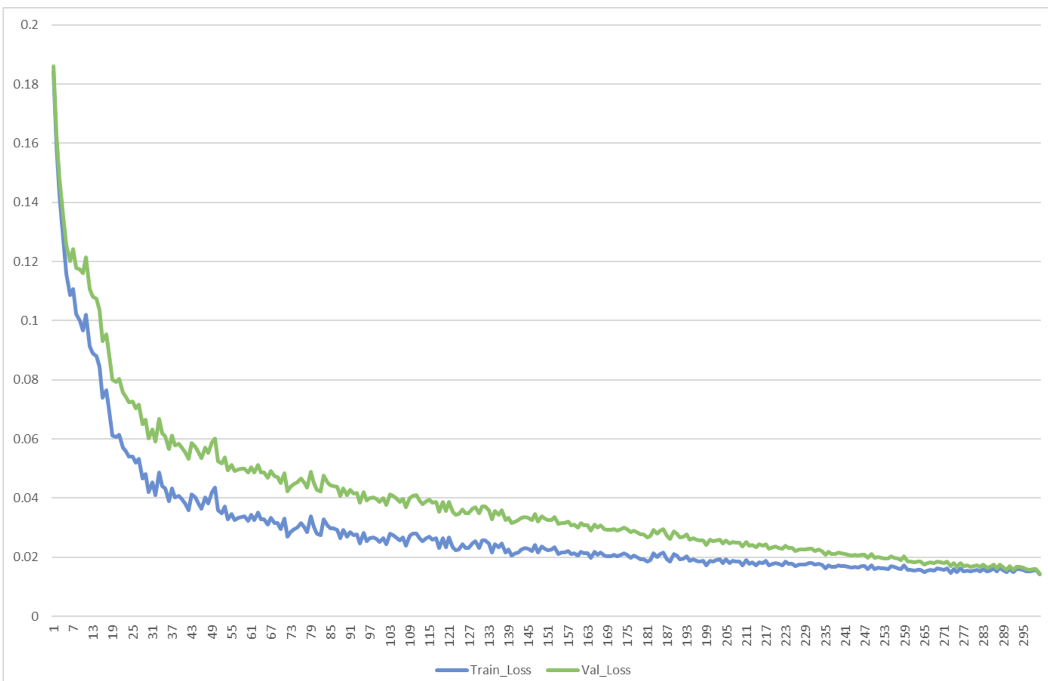

Figure 4.10: Training loss and validation loss of the charger detection model.

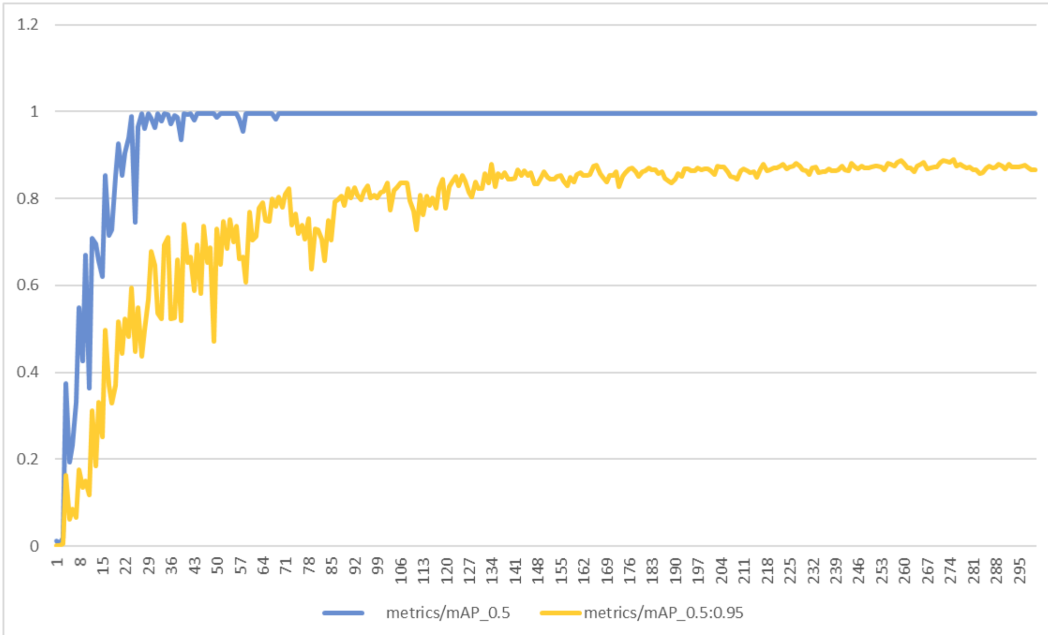

Figure 4.11: The results of both performance metrics.

In addition, an evaluation is performed for real-time charger detection while the mobile robot is moving based on the proposed method. Figure 4.12 depicts an example of the recognized results. A metric for evaluating the method performance in a practical environment is adopted, as shown in Equation 4.9.

$$Accuracy = \frac{N}{T} \tag{4.9}$$

Where N is the number of correct recognized images, and T is the total number of images used in the evaluation process. It can be observed that the accuracy of testing the developed charging station detection method in real-time scenarios can achieve an average of 95%.

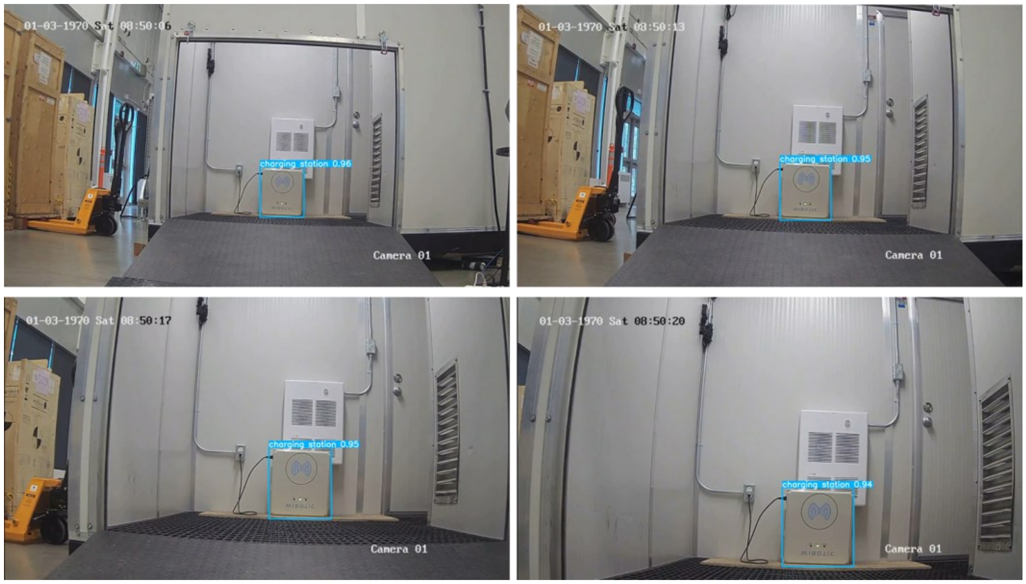

Figure 4.12: The example of real-time charging station detection.

In this chapter, the case study of the mobile robot in different positions and poses is conducted to evaluate the proposed Lidar-based approach in a real-world manufacturing environment. Figure 4.13 is an example that presents the different states of the mobile robot when performing the docking task. Based on the case study, the proposed method can achieve successful docking and recharging. The criteria to determine the success is the mobile robot can contact the charging station properly and be recharged successfully. The comparison results with the related work are presented in Table 4.6.

Table 4.6: Comparison of the existing methods.

Related work	**Method**	**Accuracy**
Kriegler et al. [141]	Faster RCNN	96.3%
Fan et al. [130]	ApriTag	97.33%
Romero et al. [147]	SSD	99.8%

-	Proposed	100%

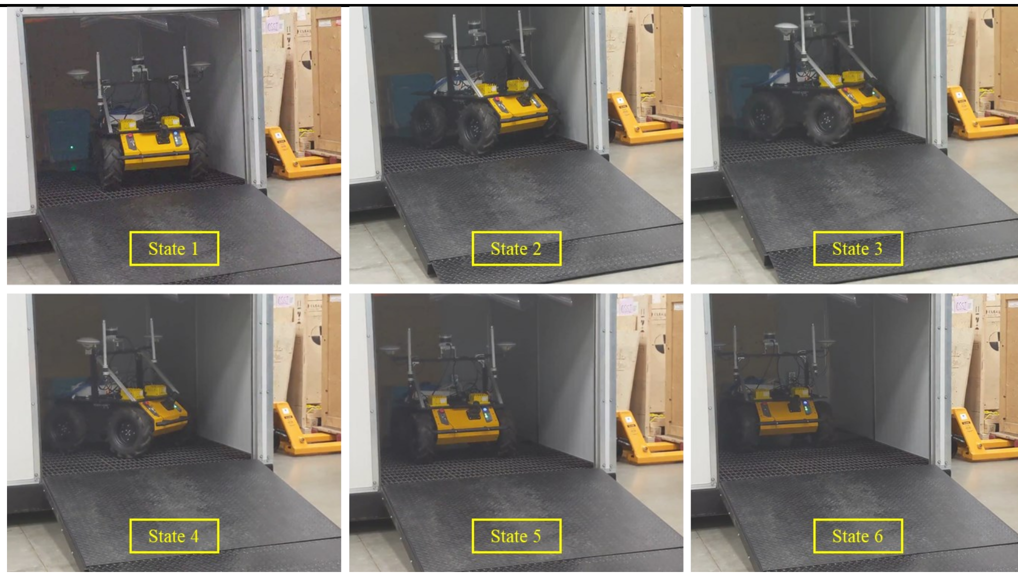

Figure 4.13: The example of autonomous docking procedures.

4.6 Conclusion

This chapter discusses the challenges faced by current autonomous docking and recharging methods in the context of mobile robots in manufacturing environments. Current state-of-the-art methods heavily rely on Lidar, which makes it expensive and time-consuming for the mobile robotic system to achieve autonomous docking and recharging applications. Therefore, a Lidar and vision data fusion method by combining deep learning object detection and Lidar-based docking approaches was proposed to address the aforementioned problems. A YOLOv7-based real-time object detection model was developed to identify the wireless charger. For evaluating the developed detection method, a set of testing images and real-time video frames captured through the Hikvision camera were used, and it achieved an average of 95% accuracy. The performance of the detection model of the charging station was compared with current methods. According to the comparison results, the proposed method outperformed other existing methods. A Lidar and vision data fusion approach was then developed to localize the wireless charger, and then navigate the mobile robot to achieve the docking to the charging station, reducing the computation costs for the system. Despite the advantages of the proposed method, it is limited by some challenges. For instance, the wireless charging station needs to be in an enclosed space, which can be used to calculate the Wall_laser distance in the proposed method. Moreover, the developed charging station detection method can be affected by the low illumination conditions in the manufacturing environment and blurring caused by the unstable movement of the mobile robot. Additionally, to achieve autonomous docking and recharging for mobile robots in any location, a path-planning algorithm to navigate the mobile robot from any position to the target position close to the charging station while avoiding obstacles in real-time is missing.

So far, this proposed Lidar-camera data fusion method for autonomous docking and recharging has only been validated on a 2-D camera and Lidar system. Future work will focus on the stereo camera and Lidar system to improve the performance of the developed method in a practical autonomous manufacturing environment. Furthermore, for the docking procedure itself, the calibration between the vision and Lidar data to improve the pose estimation of the robot with the charger needs to be implemented for future work.

Chapter 5 : An Intelligent Manufacturing Approach Based on a Novel Deep Learning Method for Automatic Machine and Working Status Recognition

5.1 Chapter Overview

With current labor shortages and high costs, workers' health, safety, and product quality may be compromised to compensate for manufacturing productivity [149]. Most manufacturing activities, such as machine tending tasks, are low-tech, repetitive, and dull, so workers can easily be replaced. With the development of robots, they are gradually being applied to machine tending tasks by collaborating with operators and even replacing workers to load and unload the raw parts of the machines. Recently, numerous research and developments have been carried out and contributed to the robotic system in smart manufacturing [150], [151], [152], [153], [154]. Since robots can sense and respond to different scenarios for various manufacturing tasks, collaborative robots (Cobots) have been widely implemented to relieve workers from risky, tedious, and repetitive manufacturing tasks [155]. In addition, it ensures high throughput rates and low costs [156].

Collaborative robots work in four different manners according to different collaborative environments [157], which are: (1) Coexistence: the robot and operator are in the same environment but generally do not interact with each other; (2) synchronized: the robot and worker are in the same workspace, but at different times; (3) cooperation: the worker and robot are in the same workspace at the same time, but each focuses on separate tasks; and, (4) collaboration: the robot and operator execute the same task, which means the action of the one has immediate consequences on the other. However, these four types of Cobots still suffer from some challenges and drawbacks. Safety is the primary concern. Although some perception sensors are utilized on Cobots to avoid injuries, operators must be careful all the time working with Cobots if they move too fast. Moreover, Cobots are often dedicated and fixed for a particular machine, and programmed for a specific task. It is an inefficient use of factory space and causes high costs to satisfy all manufacturing activities. In addition, it requires experienced engineers to reprogram the Cobots to meet different production requirements when changing manufacturing tasks.

The autonomous mobile manipulator (AMM) is proposed to extend the capabilities of conventional collaborative robots [15]. The architecture of AMM is a robot arm mounted upon a mobile platform, which combines locomotion capability with manipulation ability. Since it combines collaborative and mobile robot characteristics, it is more flexible and adaptable to changes in tasks or environments. Although AMM can perform more manufacturing tasks than traditional Cobots cannot, there is still a need for experienced operators to be onsite to assist the AMM in finding the target machines, checking the machine's working status, and taking emergency actions in case of problems. For example, the robot does not load raw materials properly.

As discussed in [158], smart devices and intelligent solutions can significantly improve the manufacturing process with artificial intelligence and imaging equipment development. They have gradually become a hot topic in smart manufacturing [159]. In [160], authors propose an intelligent perceiving and planning system based on deep learning for a 7-DoF manipulator with a vision system. The vision system and designed intelligent process enhance collaboration ability through recognizing the target objects and improving the efficiency of robot planning. In [161], a collaborative robotics framework for top-view surveillance is proposed. This study adopts pre-trained deep learning models for object detection and localization to assist human operators in managing and controlling different applications. However, there is a challenge to recognize objects because the appearance of objects changes significantly due to the change in camera position and shooting angle, which results in weak performance. In [56], authors develop a deep learning-based object detection method for the mobile robot manipulator in small and medium-sized enterprises production. This study applies region convolutional neural networks to recognize and localize the charging station and printing machine to achieve automatic tag production. In addition, it designs a human detection algorithm for the manipulator to increase safety while collaborating with operators.

Most recent studies and existing applications in smart manufacturing adopt mobile manipulators with the vision system and designed object detection methods to improve manufacturing production safety, efficiency, and intelligence. In those studies, humans often take great responsibility for controlling and operating robots through computer programs. Consequently, imitating human behavior, such as the "eye-brain-hand" process, is necessary to realize intelligent and autonomous manufacturing production truly and has become a widespread trend [160]. Therefore, this chapter proposes a deep learning-based intelligent manufacturing approach, aiming to achieve true intelligence and autonomous manufacturing production and machine tending. Compared with other solutions, the benefits of the proposed approach are low costs and high efficiency for small and medium-sized enterprises. In addition, this approach can be applied in lights-out manufacturing to reduce the workload for operators who need to tend machines and tackle the problems on the production floor. In the current study, an autonomous mobile manipulator with a vision system is adopted. The main objectives of this work are to develop an intelligent object detection method for the target CNC machine and HMI display detection in a complex environment and to design a text recognition method for the machine's working status recognition to further assist in autonomous robot decision-making and problem handling.

The remainder of this chapter is organized as follows: in Section 5.2, an overview of object detection and text recognition is presented. Section 5.3 explains the methodology of the proposed intelligent manufacturing approach for automatic machine detection and working status recognition. The case studies and results are demonstrated in Section 5.4. Section 5.5 outlines some limitations and potential future work. Finally, conclusions are presented in Section 5.6.

5.2 Literature Review

5.2.1 Object Detection

With the development of computer vision technology, object detection has been applied to many areas, including face detection, pedestrian detection, and traffic sign/light detection. Intelligent manufacturing mainly utilizes object detection techniques, such as quality management, product sorting, packaging, and assembly lines. There are two categories of object detection given in recent research: traditional object detection methods and deep learning-based detection methods.

5.2.1.1 Traditional Object Detection Methods

Most classical object detection methods are usually developed based on low-level and mid-level features, such as color, shapes, edges, and contours. In the late 1990s and early 2000s, there were several milestones in object detection methods dominated by hand-crafted features. Scale Invariant Feature Transform (SIFT) is one of them which transforms an image into a wide set of locally scale-invariant features. These scale-invariant features are invariant to image translation, rotation, scaling, illumination, occlusion, and 3D projection. Highly distinctive scale-invariant key points are further proposed to match individual features with features from known objects, and then the clusters belonging to the same object are identified through the Hough transform and finally verified by least squares. In [162], a local feature descriptor, Histogram of Oriented Gradients (HOG), was first proposed. It counts the occurrences of gradient orientation in localized portions based on a dense grid of uniformly spaced cells and uses overlapping local contrast normalizations to improve accuracy. Despite being similar to SIFT descriptors, it is a significant improvement and gives excellent results for human detection. Because of the advantages of HOG, it has been an important architecture for many object detection methods. As an extension of HOG, the Deformable Part-based Model (DPM) [163] is the best classical object detection method. DPM achieves improvement in both precision and efficiency compared with the previous methods. It considers an object as a global template covering an entire object and a collection of part templates. Then the models are trained discriminatively through a support vector machine. However, as the performance of handcrafted features reached its limits, object detection has fallen into a bottleneck period.

5.2.1.2 Deep Learning-Based Detection Methods

Deep learning-based object detection breaks the deadlocks as deep convolutional neural networks can learn an image's robust and high-level features. With the development of GPU computing resources and the availability of large-scale datasets, many deep learning-based object detection methods have been developed. This section introduces the milestone frameworks in object detection because almost all detectors proposed over the last seven years use one of them as the foundation. In general, deep learning-based object detection methods are grouped into two main categories: two-stage methods and one-stage methods.

Regions with Convolutional Neural Networks (R-CNN) is the first proposed deep learning-based object detection algorithm. The basic idea is extracting enormous proposals as candidate regions based on selective search [164], and then scaling all proposals to fixed-size images and feeding them into a pre-trained CNN model to obtain object features. Finally, using Support Vector Machine (SVM) classifiers to predict the presence of the object and to recognize the object category. However, the redundant feature computations on many overlapped proposals lead to a plodding detection speed. Therefore, Ross Girshick proposed Fast-RCNN [165] to improve R-CNN. It can train a detector and predict bounding boxes simultaneously under the same network configurations, which improves the detection speed.

Faster-RCNN [166] is proposed shortly after the Fast-RCNN. Its main contribution is the design of the Region Proposal Network (RPN). RPN can generate region proposals quickly and efficiently. Although Faster-RCNN breaks through the speed bottleneck, computation redundancy occurs at the subsequent detection stage. Joseph Redmon presented a method called You Only Look Once (YOLO) [167] to solve this problem. This algorithm applies a single neural network to simultaneously divide the image into candidate regions and predict probabilities containing the object for each region. Although YOLO improves the detection speed significantly, it suffers from a low accuracy in localization compared with previous methods, especially in some small object detection. Meanwhile, the proposed Single Shot Multi-Box Detector (SSD) [168] introduces multi-reference and multi-resolution detection techniques to improve detection accuracy while maintaining high detection speed.

Although SSD and YOLO-based methods achieve good accuracy and computation efficiency performance, those models require too much labelled data for training. However, sometimes much data is not available or easy to obtain for a specific object or task. These methods also suffer from bad performance for small object detection tasks. In addition, those milestone methods mainly focus on classifying and detecting different object categories, so it is not suitable for detecting a particular target CNC machine from many machines in the industrial environment. In this study, Siamese Neural Networks [169], [170] are integrated with the region proposal network to solve this problem. It helps build models with good accuracy and efficiency, even with fewer data and imbalanced class distribution.

5.2.2 Scene Text Recognition

Scene images contain abundant and precise information, especially the text in the scene images, which is helpful for people to understand the surrounding environment. Scene Text Recognition (STR) is developed as one of the research fields of computer vision to recognize the text from scene images and convert it into machine-readable information [170]. STR consists of two stages, which are text detection and text recognition. Text detection aims to determine whether there is text in a given image or video and to localize the text using bounding boxes. Text recognition aims to identify the detected text and translate the text into machine-readable information [171]. Recently, scene text recognition applications gained much popularity in many fields, such as car plate recognition, product labelling, sorting, and packaging.

Traditional text detection and recognition algorithms often adopt handcrafted features to separate the text and non-text regions in a scene image, requiring demanding and intricate image processing steps [172]. Since traditional methods are not robust, challenging to implement, and constrained by the complexity of creating handcrafted features, they can hardly deal with intricate circumstances. Therefore, this section only discusses recent deep learning-based methods.

5.2.2.1 *Text Detection*

Given the similarity between text detection and general object detection, most text detection methods use a general object detection framework as core architecture. Combining a random forest word classifier and convolutional neural network (CNN) was proposed to achieve a high recall rate and precision [173]. In [174], Faster-RCNN was enhanced by the inception region proposal network, responsible for obtaining text candidate regions. Then an iterative bounding box voting scheme was applied to ensure high recall and the best results. In [175], an image processing method named non-maximum suppression was proposed to reduce overlapped and redundant effects. An SSD-based text detection method called TextBoxes++ was proposed to detect arbitrary-oriented scene text with high accuracy and efficiency using a quadrilateral rectangle without post-processing steps involved, such as non-maximum suppression. However, the accuracy of region proposal-based methods heavily relies on the candidate regions' generation. Unlike general objects, the text usually has varying aspect ratios. Therefore, it is necessary to manually design anchors with different aspect ratios and scales, which makes text detection complicated and inefficient.

5.2.2.2 *Text Recognition*

Most text recognition methods recognize the scene text by grouping the recognized characters, making text recognition inefficient and difficult for real-time applications. In [176], an accurate scene text recognition method without character-level segmentation was proposed based on a Recurrent Neural Network (RNN). Shi [177] presented a convolutional recurrent neural network (CRNN) to recognize the scene text in arbitrary lengths in scene images by stacking CNN and RNN. This method is not limited to any predefined lexicon and can achieve remarkable performances in scene text recognition tasks. In [178], recursive recurrent neural networks with attention mechanisms [179] were developed to achieve good performance for dictionary-free scene text recognition. Liao [180] proposed a TextBoxes method by combining the SSD and CRNN to speed up the text recognition task to recognize text in scene images.

Although numerous studies have been done, the limitations and challenges of scene text recognition while applying it to real-world manufacturing tasks have not been researched. Efficiency and precision are two main concerns using text recognition methods to obtain the working state of a CNC machine. Text blurring often happens when the mobile robot moves due to camera shake and de-focus, which degrades recognition accuracy [181]. In addition, efficiency is a shortcoming of deep learning-based methods, making it challenging to deploy those methods on mobile devices and lightweight systems [182]. Therefore, this study integrated some pre-processing steps that reduce the complexity of images into the developed text recognition method to improve image quality. In addition, the combination of HOG and CRNN can realize real-time working status recognition of CNC machines while ensuring accuracy.

5.3 The Proposed Intelligent Manufacturing Approach

There are three main targets of this research. One is to automatically recognize the target CNC machine in a complex environment through a camera mounted on a moving mobile manipulator. Simultaneously, detect the human-machine interface (HMI) of the CNC machine. Once the HMI is detected, it can identify the working status by recognizing the text on the HMI display. This study proposes a novel deep learning-based approach to achieve these goals. First, the Siamese region proposal network (SiameseRPN) method was proposed to achieve the target CNC machine and HMI detection. Then, the detected HMI images are extracted, pre-processed, and used as inputs for working status recognition. Finally, the novel text recognition method by combining the projection-based segmentation and the convolutional recurrent neural network (CRNN) was developed to identify the working status of the target CNC machine. Figure 5.1

presents the flowchart of the proposed intelligent manufacturing approach. The following section explains the SiameseRPN method and the machine's working status recognition method.

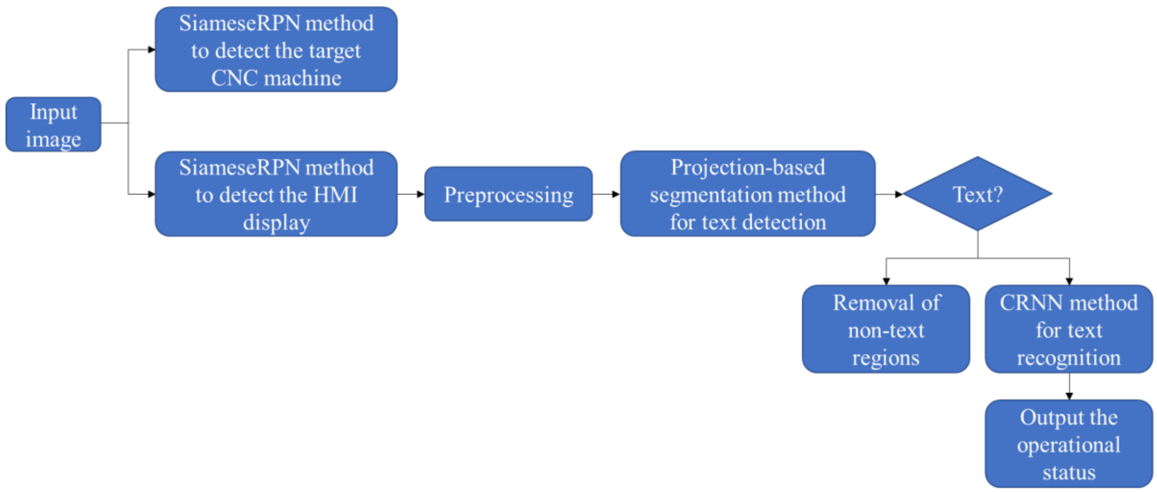

Figure 5.1: The flowchart of the proposed intelligent manufacturing approach.

5.3.1 Siamese Region Proposal Network (Siameserpn) Architecture

The Siamese Region Proposal Network is proposed for target detection and consists of two subnetworks: The Siamese Neural Network and the Region Proposal Network. The framework of SiameseRPN is shown in Figure 5.2.

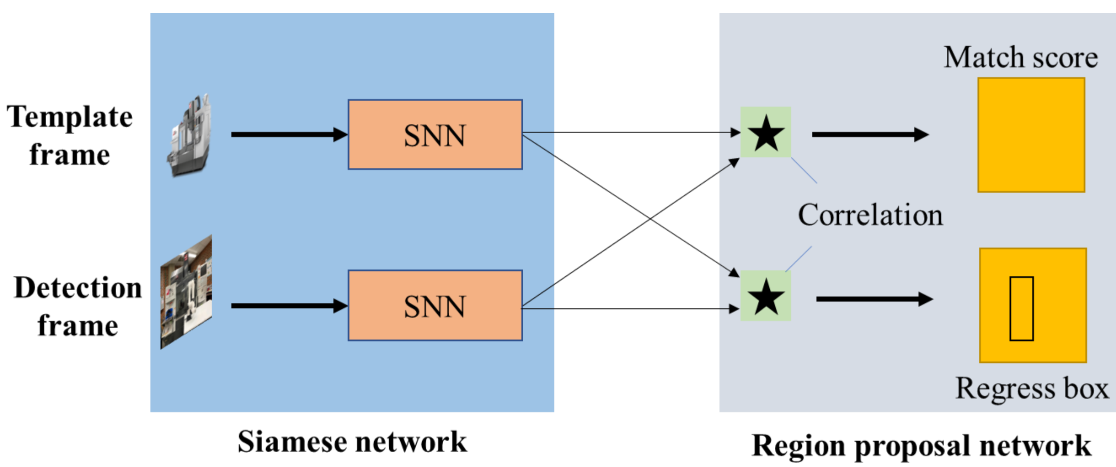

Figure 5.2: The architecture of the proposed SiameseRPN method.

5.3.1.1 Siamese Neural Network

The Siamese Neural Network (SNN) network [169] has been proven effective in the object-tracking domain. It uses two branches, which share the same parameters, to learn the similarity between each other. The objective of SNN is to understand the embedding space, which places similar items nearby. In other words, SNN is trained with positive and negative pairs of objects, where positive pairs correspond to samples that need to stay close in the embedding space while negative pairs need to stay far away. Although SNN has drawn significant attention because of its balanced accuracy and speed, it lacks bounding box regression. It has to do a multiscale test to locate the video targets, making it less elegant. This study integrated a region proposal network with SNN to address this drawback.

In this study, SNN adopts a convolutional neural network without padding to extract the features. Assume the term L_T represents the translation operator of the kernel; then remove all the paddings to satisfy the full convolution with strike *k*.

In Equation 5.1 [169], *h* is a fully convolutional function with integer stride k that maps signals to signals for any translation τ:

$$h(L_{k\tau}x) = L_{\tau}h(x) \tag{5.1}$$

In this subnetwork, the template frame and the detection frame are fed into the SNN as inputs. The template frame is the previous frame in the video and the detection frame is the current frame in the video. In addition, these two branches have the same hyperparameters and transformations. For convenience, two outputs of the SNN are represented by *M(z)* and *M(x)*, which are the feature map representations of the template branch and detection branch.

5.3.1.2 *Region Proposal Network*

The region proposal network (RPN) is first proposed in Faster-RCNN [166], which can extract precise candidate regions quickly and efficiently. Therefore, it makes proposal generation very effective while achieving high accuracy.

In this subnetwork, RPN consists of two parts: pairwise correlation and supervision. The supervision consists of two branches: regression and classification. The regression is to generate the candidate regions, and the classification is to distinguish the foreground and background of input images. If n anchors are automatically generated, RPN needs $2n$ representation channels for the classification task and $4n$ presentation channels for the regression task. Therefore, the pairwise correlation needs to divide $M(z)$ into two datasets $[M(z)]_{cls}$ and $[M(z)]_{reg}$, and divide $M(x)$ into $[M(x)]_{cls}$ and $[M(x)]_{reg}$. Then, $[M(z)]$ is used as the correlation kernel to compare similarity with $[M(x)]$. Finally, the regression correlation C_{reg} and classification correlation C_{cls} are computed through Equation 5.2 [176]:

$$C_{reg} = [M(z)]_{reg} * [M(x)]_{reg},\ C_{cls} = [M(z)]_{cls} * [M(x)]_{cls} \tag{5.2}$$

where $[M(z)]_{reg}$ and $[M(z)]_{cls}$ are used as convolutional kernels, and the sign * is the convolution operation.

Smooth L_1 loss function with normalized coordinates and cross-entropy loss function are used for regression and classification, respectively. If P_x, P_y, P_w, P_h represent the coordinates of the center point, the width and the height of predicted bounding boxes, and T_x, T_y, T_w, T_h represent those parameters of the ground truth. Then normalized distances between the prediction and ground truth can be computed. RPN is optimized by minimizing the loss function in Equation 5.3 [169]:

$$Loss = L_{cls} + \lambda L_{reg} \tag{5.3}$$

where λ is the hyper-parameter to balance the performance of regression and classification. L_{reg} is the loss function for regression, and L_{cls} is the loss function for classification.

5.3.2 Associative Recognition of Working Status

The study proposes a novel text recognition method combining HOG and CRNN to accurately and efficiently identify the working status of the CNC machine. This method consists of two stages: text detection using projection-based segmentation, and text recognition using convolutional recurrent neural network. The steps of the proposed text recognition method are presented in Figure 5.3.

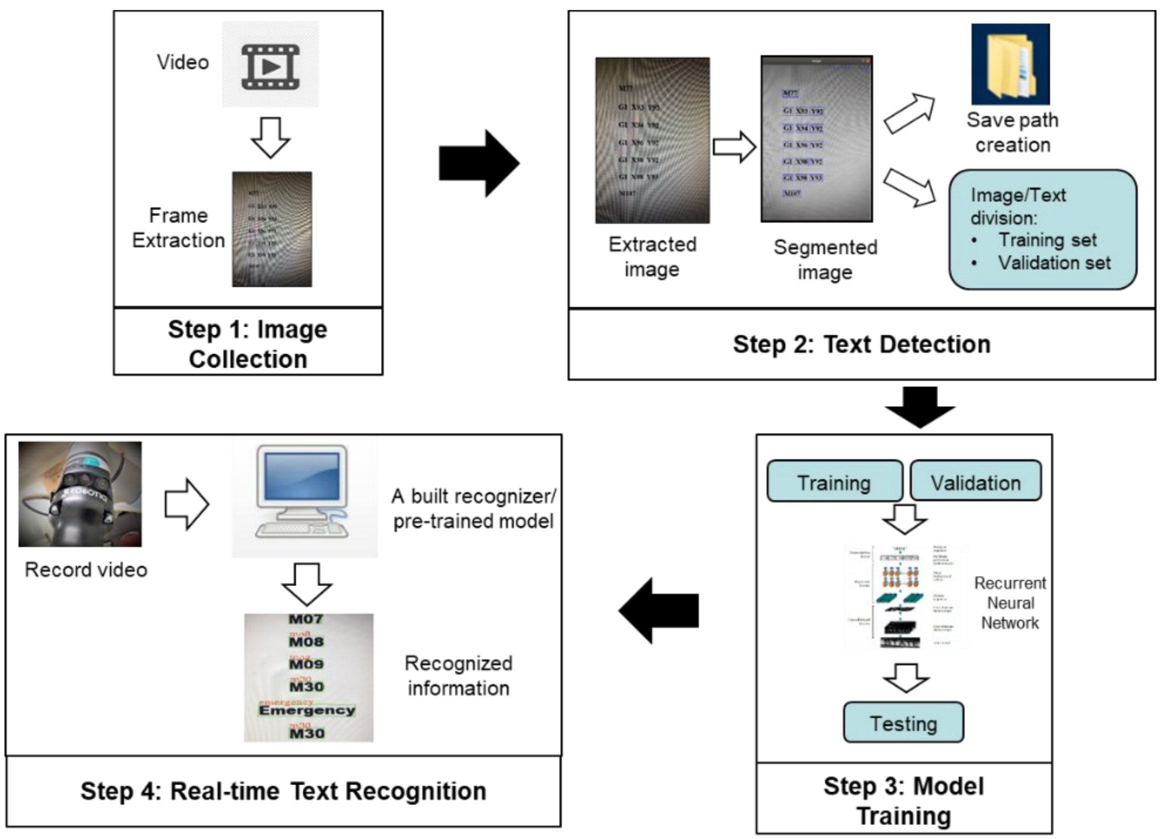

Figure 5.3: The procedure of the proposed text recognition method.

The proposed segmentation algorithm has three steps. First, the projection scans horizontally to determine the upper and lower bounds of the text area. Second, the projection scans vertically to define the left and right bounds of the text area, and then bounding boxes containing the text information are generated. Finally, the non-text regions are removed and the feature maps of text regions are extracted for recognition.

Before applying this segmentation algorithm, some pre-processing steps are adopted. First, transform the color images into grayscale images. The pixel value is from 0 to 255 in a grayscale image. Second, binarization will convert the grayscale images to binary images using thresholding. Pixels with a greater value than the threshold P are replaced with white color and other pixels are replaced with black color. Here the value of the black pixel is set to one, and the value of the white pixel is set to zero. That is:

$$Value\ (i, j) = 1,\ pixel\ (i, j) < P \tag{5.4}$$

$$Value\ (i, j) = 1,\ pixel\ (i, j) > P \tag{5.5}$$

$$0 < i < \text{width},\ 0 < j < \text{height} \tag{5.6}$$

Width and height are two image parameters, and m is a variable threshold according to the image background color. The number of black pixels accounted for by Equation (5.7):

$$pixelRow[j] = \sum_{i=\frac{m}{N}\times width}^{\frac{m+1}{N}\times width} Value(i,j) \tag{5.7}$$

$$0 < j < \text{height},\ m = 0, 1, 2, \ldots, N - 1 \tag{5.8}$$

Here divides the whole image into N parts with the same height and width. Then, an appropriate threshold Q is set to separate the text and non-text pixels. In Equations (5.9) and (5.10), when $pixelRow\ [j] > Q$, it is set to one, otherwise, it is set to zero:

$$Horizontal\ [j] = 1, pixelRow[j] > Q \tag{5.9}$$

$$Horizontal\ [j] = 1, pixelRow[j] < Q \tag{5.10}$$

$$0 < j < \text{height} \tag{5.11}$$

Horizontal [j] is calculated, traversing the entire array. When the array's value changes from zero to one, record this pixel as the start point. When the array's value changes from one to zero, record this pixel as the endpoint. Finally, the text bounding boxes are generated, and the three steps of text detection are depicted in Figure 5.4.

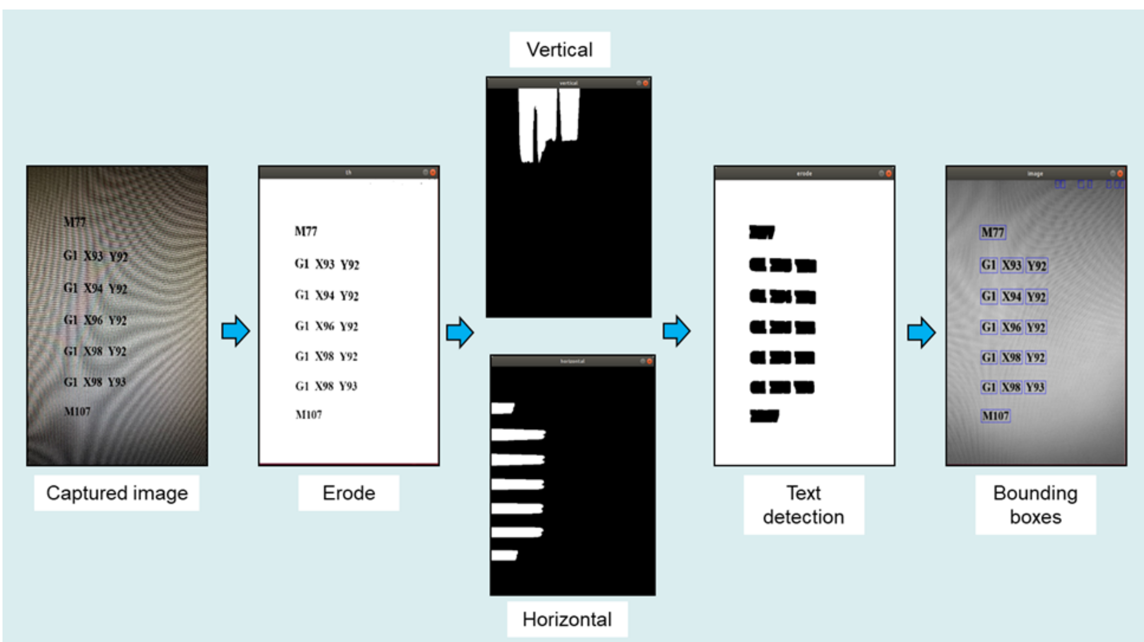

Figure 5.4: The steps of text detection.

After text detection, CRNN [177] is used to recognize the detected text. The framework of CRNN used in this study is shown in Figure 5.5. It consists of three subnetworks: convolutional layer, recurrent layer, and transcription layer. The convolutional layer automatically extracts feature map sequences from input images. Then the extracted feature sequences are fed into the recurrent network to predict each feature sequence. The transcription layer is used to translate each frame prediction generated in the recurrent layer and group them into a labelled sequence. The main benefit of the CRNN is that it can be trained with one loss function despite being composed of different network architectures.

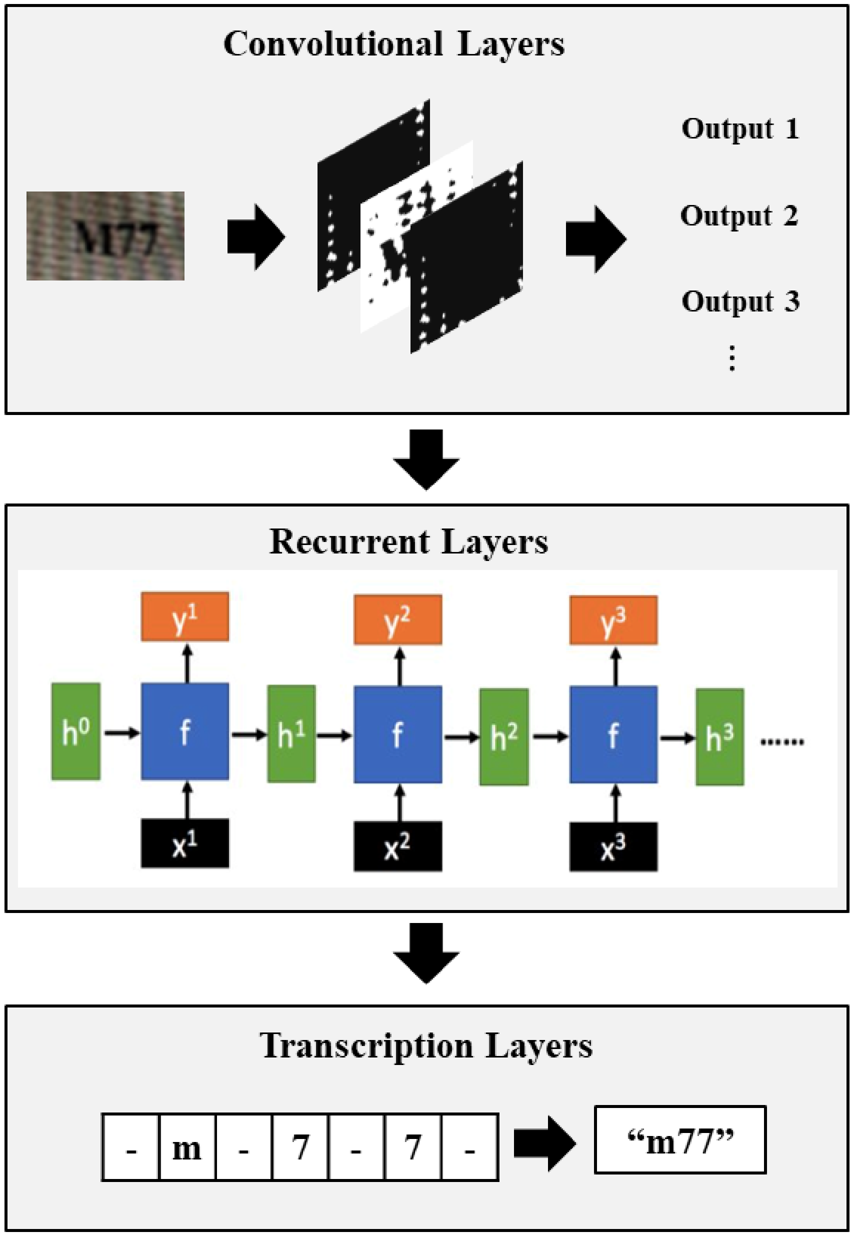

Figure 5.5: The architecture of the CRNN.

The convolutional layer in CRNN is modified from a standard CNN model by removing the fully connected layers. It extracts feature map representation from an input image. All the input images are scaled to the same size in the pre-processing step. Then feature vector sequence is obtained after the feature maps are extracted in the convolutional layer. To keep the order of the text unchanged, each feature vector is generated from left to right on the feature maps, and the *i*-

th feature vector is the concatenation of the i-th columns of all the feature maps. Finally, those sequences will be fed into the recurrent layer as inputs.

The feature maps are divided into many rectangle regions with a fixed width, which is the receptive field. Each feature vector sequence can represent one rectangle region, which is illustrated in Figure 5.6.

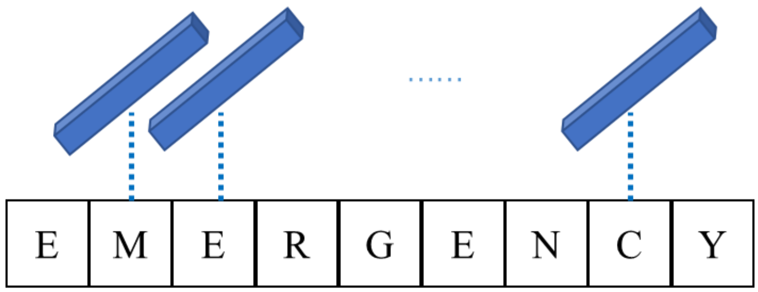

Figure 5.6: The receptive fields.

Although CNN has proven good performance in recognizing general objects, it is not suitable for text detection because of its various aspect ratios. Those feature maps extracted from the convolutional layer need to be transformed into sequential representations to avoid being affected by the length variation. Therefore, the recurrent layer modified from the recurrent neural network is adopted to solve this problem. The main advantage of RNN is that it has the stable ability to capture the context in a sequence, which is more useful than identifying each character individually because some ambiguous characters can be easily predicted by observing the contexts.

The transcription layer aims to convert the predicted labelled distribution into a label sequence with the highest conditional probability. Connectionist Temporal Classification (CTC) [183] is used to calculate conditional probability. The conditional probability in this study is defined as the label sequence conditioned on the per-frame prediction. The expression of the conditional probability is shown in Equation 5.7 [177]:

$$p(\boldsymbol{l}|\boldsymbol{y}) = \sum_{\pi:B(\pi)=l} p(\boldsymbol{\pi}|\boldsymbol{y}) \tag{5.7}$$

where $\boldsymbol{l}$ is the label sequence, $\boldsymbol{y}$ is the predicted labelled distribution obtained from the recurrent layer ($\boldsymbol{y} = y_1, y_2 \dots y_T$), and T is the length of the input sequence. B is a mapping function, which maps the sequence π onto $\boldsymbol{l}$ by removing the duplicate labels and blank labels. For example, B maps the sequence "-*m*-*7*-*7*-" ("- "represents blank label) onto "*m77*", which is shown in Figure 5.3.

5.3.3 Data Augmentation

Although this study adopts enough data to train the proposed model, images of CNC machines tend not to have extensive features due to the limited number of different CNC machines. As a result, increasing the diversity of the training data is widely used to improve the generalization and reduce overfitting [184]. Some geometric distortions are randomly added in this study, including rotation, translation, scaling, vertical flipping, and image distortions, such as Gaussian blur and noise.

5.3.4 Transfer Learning

Deep learning models often require a large number of input images as training data. However, it is tough to collect enough images for some applications. Transfer learning provides an alternative strategy to address this problem by using a pre-trained deep learning model as a starting point for another training task rather than building a model from scratch. This study adopted the modified AlexNet pre-trained from ImageNet with the parameters [185], significantly improving training efficiency.

5.4 Experiments and Results for the proposed method

5.4.1 Robot System Structure

Figure 5.7 shows the DOBOT CR5 manipulator used in the robot system. It is fixed on a moveable desk that can move around the working environment.

Figure 5.7: The structure of the robot system.

Specifications of this manipulator are demonstrated in Table 5.1. A webcam with 1920 × 1080 pixels resolution is mounted on the top of the robot arm, used to sense the surrounding environment.

Table 5.1: Specifications of the robot manipulator.

Parameter	**Value**
Weight	25 kg
Maximum Payload	5 kg

Max Reach	1096 mm
Voltage	DC48 V
Maximum Speed of TCP	3 m/s
Communication	TCP/IP, WIFI
Power	150 W
Axes	6

5.4.2 Training Details

The webcam was used to collect the CNC machine and HMI images. The dataset includes training and testing images with a resolution of 960 × 1080. Images were gathered from different positions in our lab. Two hundred images, including 160 training images and 40 testing images, are used to train and test the developed models. The experiments were conducted on a laptop equipped with an Intel Core i7-8750H 4.0 GHz CPU and a single NVIDIA GeForce GTX 1060 under the Ubuntu 18.04 64-bit operating system.

During the training process, Stochastic Gradient Descent (SGD) is adopted to train the proposed SiameseRPN based on the pre-trained SNN model using ImageNet. In addition, picking positive and negative training samples is also necessary. The criterion used in the target CNC machine detection and HMI detection tasks is based on intersection over union (IoU) and two thresholds T_{high} and T_{low}. When the predicted bounding boxes have $IoU > T_{high}$ with respect to the ground truth, they are considered positive samples, which means the correct results. When the $IoU < T_{low}$, those are considered negative samples and will be removed from the results; the T_{high} and T_{low} are set as 0.3 and 0.6 in this study, respectively.

5.4.3 Evaluation

This section evaluates the proposed methods using the video recorded by the webcam mounted on the robot manipulator in the lab environment. Figure 5.8 – 5.14 show some case studies for detecting the target CNC machine and HMI display where the target CNC machine is detected in the green bounding box, obstacles such as the table in the yellow bounding box, and the HMI in the red bounding box. The robot system moves from the lab door toward the target CNC machine and records real-time video through the mounted camera. Based on the proposed SiameseRPN method, the target CNC machine and HMI display are recognized and located using green and red bounding boxes, respectively. Figure 5.17 presents particular examples of recognizing the text information on the HMI display using the proposed working status recognition method, such as the basic instruction information and G-codes. Once the text is identified, it is converted into machine-readable text that assists the autonomous robot system in recognizing the real-time working status of the machine, which is significant for the robot system to achieve further decision-making and execution actions to tackle the emergency and ab-normal conditions in a completely autonomous environment.

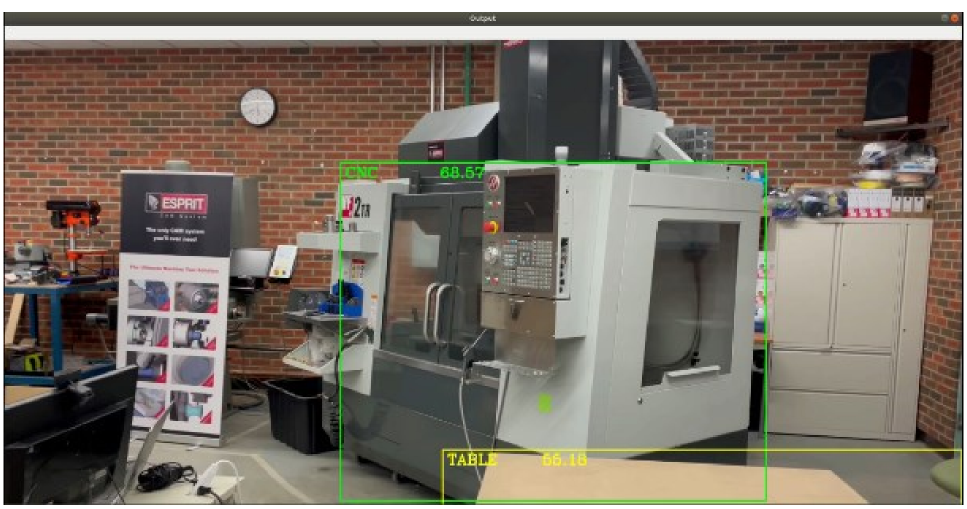

Figure 5.8: Validation results for the target CNC machine and HMI detection.

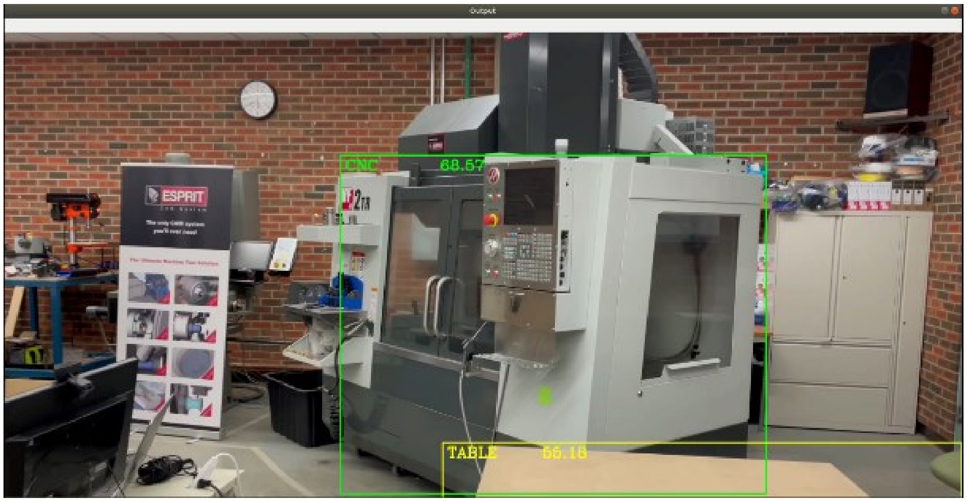

Figure 5.9: Validation results for the target CNC machine and HMI detection.

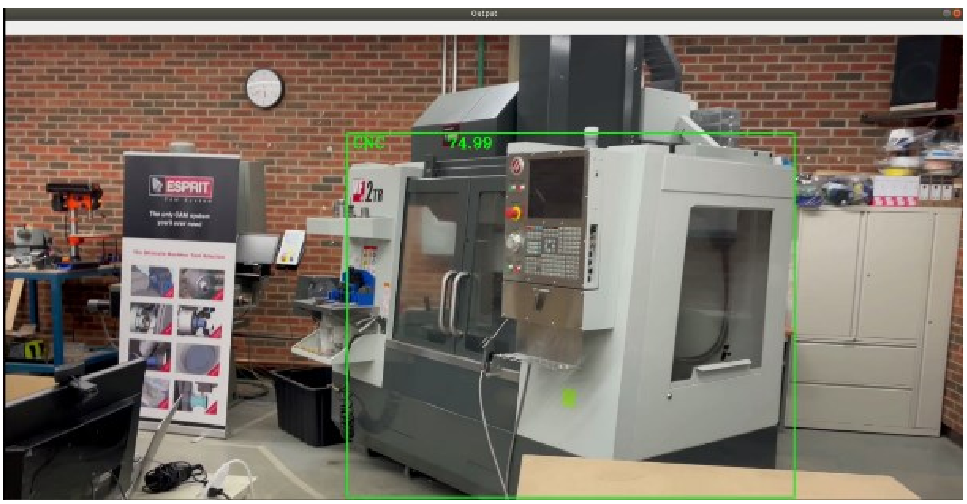

Figure 5.10: Validation results for the target CNC machine and HMI detection.

Figure 5.11: Validation results for the target CNC machine and HMI detection.

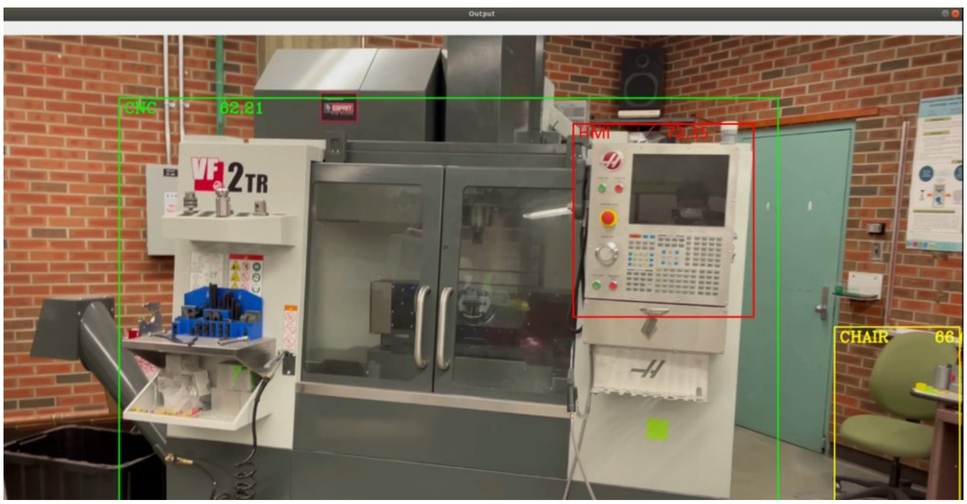

Figure 5.12: Validation results for the target CNC machine and HMI detection.

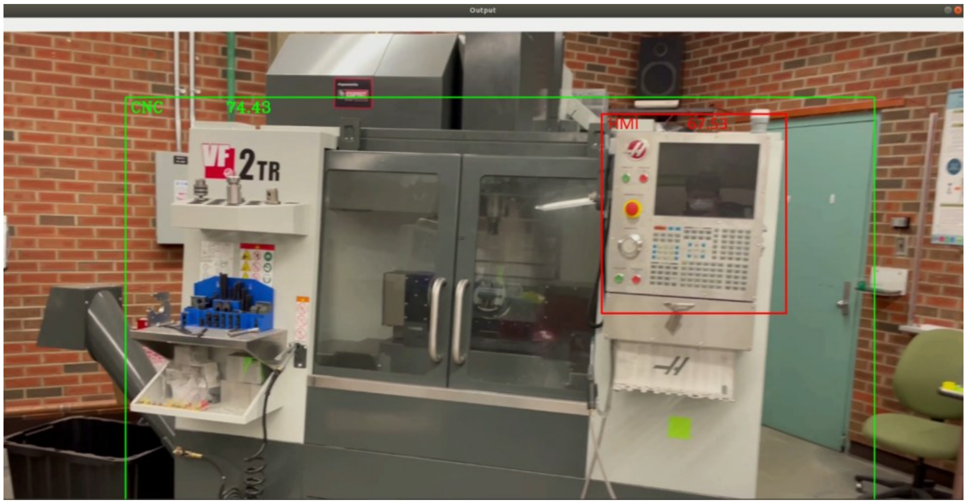

Figure 5.13: Validation results for the target CNC machine and HMI detection.

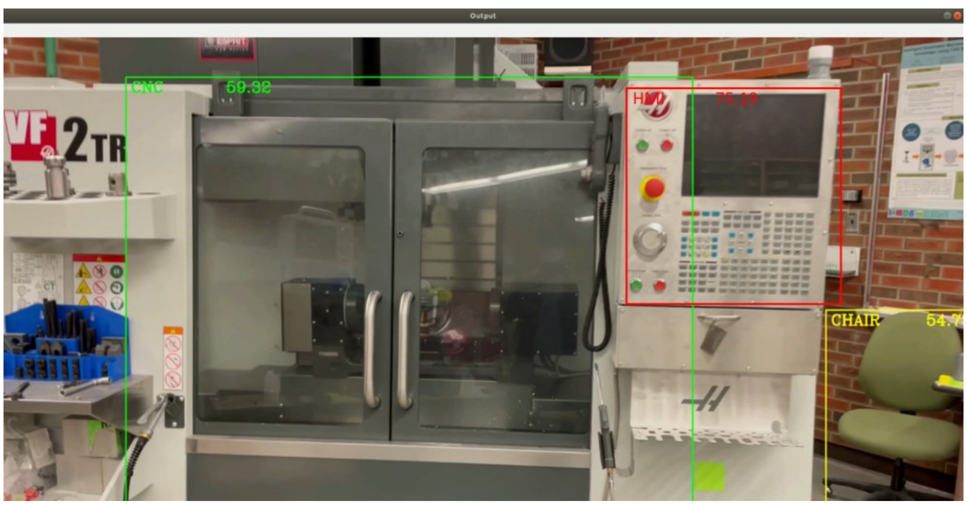

Figure 5.14: Validation results for the target CNC machine and HMI detection.

5.4.4 Results

5.4.4.1 Target Detection

For the target CNC machine detection, the training and validation loss of the model is plotted in Figure 5.15. It can be observed that the loss for training and validation achieved 0.127 and 0.112, respectively.

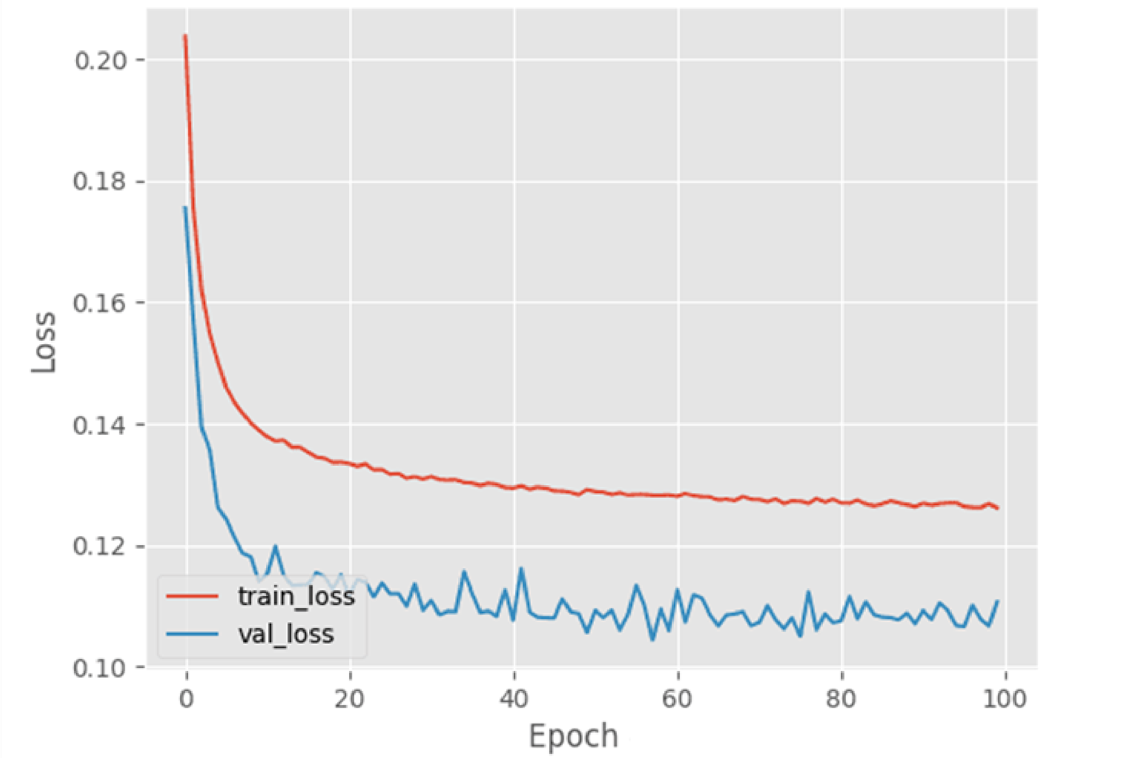

Figure 5.15: The training and validation loss for the target CNC machine detection.

In this chapter, two hundred frame images from the recorded video were used for validation. The accuracy of the proposed method is shown in Equation 5.12.

$$Accuracy = \frac{N}{T} \tag{5.12}$$

where N is the number of correctly detected objects, and T represents the total number of images used. Table 5.2 compares our proposed method with three milestone methods in CNC machine detection: Faster-RCNN, SSD, and YOLO.

Table 5.2: Comparison of accuracy with benchmark methods (the target CNC machine detection).

Method	**Accuracy**
Faster-RCNN	0.67

SSD	0.54
YOLO	0.53
The Proposed Method (this chapter)	**0.78**

The performance is acceptable considering the size of the dataset. Different parameter values, such as the batch size of training data, the learning rate of the model, and the activation function, have been applied, but they achieved limited benefit. Therefore, increasing the training dataset size would be better to improve the models' performance.

5.4.4.2 Associative recognition of working status

In this chapter, the proposed scene text recognition method was first validated in a virtual autonomous manufacturing environment and will be further modified and evaluated in real scenarios. Thus, a virtual environment is created in RoBIM simulation software. RoBIM is a robotics simulator that enables the manufacturer to plan, simulate, and control the industrial robot. It provides a virtual design and validation environment for linking the digital model to a robot's control system. In the established simulation environment, a Haas CNC machine and UR5 manipulator are used to perform the case study, which is shown in Figure 5.16.

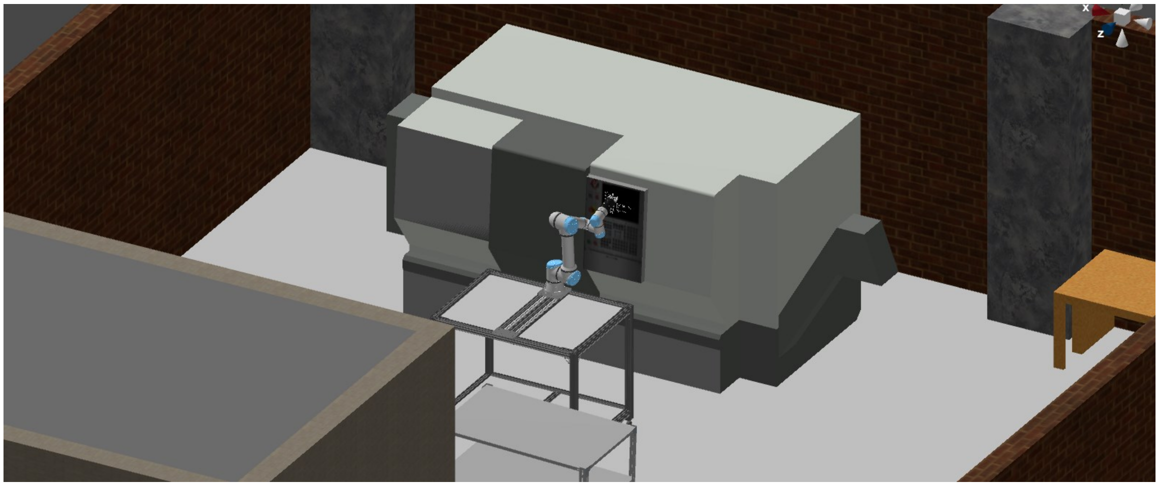

Figure 5.16: The virtual simulation environment.

The camera system built on the manipulator can capture the real-time scene images of the display panel. Based on the captured scene images, the proposed scene text recognition method is used to recognize and understand the information and alerts. To realize abnormal condition recognition, two simulated abnormal scenarios are designed. One scenario is that if the workpiece is not put in the right position, then the “wrong part position” will be shown on the screen. The other is that if the tool gets damaged and needs to be changed, then a “tool damage” alert will be presented on the screen. An example of recognizing the information on the CNC machine’s screen in a simulation environment is demonstrated in Figure 5.17. Some G-codes, execution commands, and alert information are shown on the CNC screen, which can be recognized through the proposed scene text recognition method by the robotic system and then facilitate the following decision-make of the robotic system to handle the happened the abnormal condition during the manufacturing process. Moreover, the experiments will also be carried out in the real system. However, compared with the simulation, the real system has some limitations, such as the background color of the display screen, the condition of the illumination in the working environment and the shake of the camera used in the robotic system.

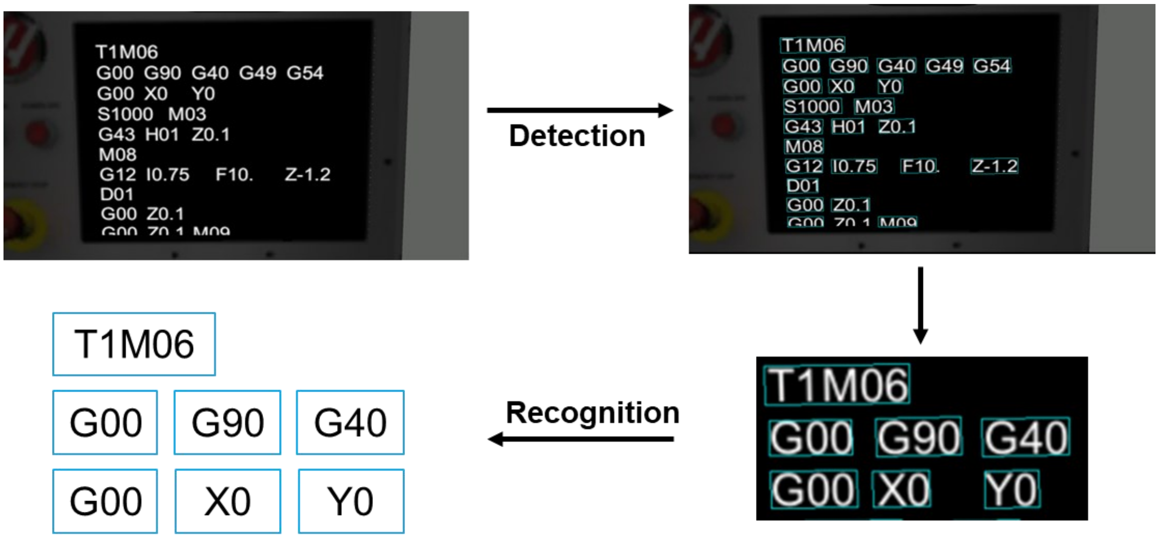

Figure 5.17: The example of the case study.

From the established simulation environment, 1000 images are obtained and then adopted for the model training and test. For the training set, 80% of the dataset is taken, and 20% for the test set. For validation purposes, a new dataset containing 20 images is used to evaluate the proposed method. The benchmark method RNN was compared with the proposed method in this study to estimate the performance of the proposed method. The results for both methods are presented in Table 5.3. It can be concluded that the proposed method improves the recognition performance in precision without the loss of efficiency.

Table 5.3: Compared results for RNN and the proposed method (working status recognition).

Method	**Accuracy**
RNN	78.2%
The Proposed Method (this chapter)	**85.7%**

5.5 Discussion and Limitations

This paper presented a novel framework to support autonomous machine tending in lights-out manufacturing, which minimizes the involvement of machine operators during the production process. Most machine-tending robot systems adopt collaborative robots that connect to the machine via Ethernet to perform various tasks efficiently under the inspection and tending of onsite operators [16], [42]. Autonomous mobile robots can use our proposed architecture to achieve fully autonomous machine tending through the integrated vision system and computer vision-based algorithms. The proposed intelligent manufacturing approach can truly achieve autonomous machines, HMI, and emergency detection without the operator's assistance to support various machine-tending tasks. The proposed methods are flexible, scalable, and adaptable; however, it has some limitations. Text detection and recognition are more sensitive to image quality than general object detection [186]. Sometimes, the mobile camera captures images and videos suffering from poor lighting conditions, such as shadow in the images or reflection of light due to inappropriate shooting distance or angle, making the feature extraction process challenging. Moreover, camera motion or shaking while capturing images can cause text to be blurred [182].

Further research to address the problems mentioned above is necessary to implement the proposed algorithms in autonomous machine-tending applications in a real-world industrial environment. Since the performance of deep learning models largely relies on the size of training data, a bigger and better dataset will be collected in the future and used to improve the proposed approach. Image deblurring [187], [188] and reflection removal [189], [190] methods will also be applied to improve the quality of the captured images. Moreover, a reference dictionary containing common abnormal conditions of CNC machines will be designed to match and correct text recognition results.

5.6 Conclusion

Smart manufacturing has been considered high in intelligence, efficiency, accuracy, productivity, and safety compared to traditional manufacturing. With robotics development, vision sensors and computer vision-based algorithms have been widely used in smart manufacturing to tackle complex and hazardous production tasks. Automatic working status recognition and emergency handling through autonomous robot systems have become critical steps in the autonomous manufacturing process, ensuring that machines work properly without human involvement. However, onsite workers still carry out the current inspection of the manufacturing process and machine tending by working with Cobots because of lacking an appropriate intelligent manufacturing approach for machine detection and working status recognition. This chapter developed an automatic deep learning-based approach to detect a particular CNC machine and the human-machine interface simultaneously in real-time in a complicated lab environment through a camera mounted on the mobile manipulator. In addition, it can identify the working status of the machine by automatically recognizing the text information on the HMI display. According to the validation results, the developed methods are proven to achieve good performances in both accuracy and efficiency. The proposed target CNC machine detection method is 16.5% more accurate than the milestone method Faster-RCNN. The developed machine's working status recognition method is 10% more accurate than the benchmark algorithm tested. However, the performance of deep learning-based methods often largely relies on the size of training data and image qualities, so a bigger and better dataset will be collected in the future and used to improve the proposed approach.

Chapter 6 : An Intelligent Operation Framework for Autonomous Robot-based Machine-Tending Applications

6.1 Chapter Overview

Machine tending is a common and significant part of the manufacturing industry through interaction and cooperation with the machine and surrounding environments [4]. In the past, most machine tending tasks, such as loading the raw parts on the machine and unloading produced parts onto the workstation, transferring workpieces between different machines, and quality inspection, were executed manually by laborers. Since the first robot was applied to die-casting machines in 1960, machine tending has become the typical application of the traditional industrial robot [60]. Currently, industrial robots have been widely used to replace workers from low-tech, dull, repetitive, and highly risky machine tending tasks and play a crucial role in addressing labor shortages and skill gaps. The most popular conventional industrial robots, such as KUKA and ABB, can only work in structured and pre-defined environments with long changeover time [33]. To assure safety, the working cell is surrounded by fences, which causes a waste of working space [9]. Traditional industrial robots are also limited in terms of their capabilities and need specialized programming whenever the workpieces change, or machine tending tasks change [68]. With advancements in AI and robotics technologies, collaborative robots have gradually replaced conventional industrial robots to deal with more complex machine-tending tasks by collaborating with experienced operators [11]. Although some perception sensors are installed on collaborative robots, operators still need to be careful all the time to avoid injuries when they interact with collaborative robots. In addition, collaborative robots have some limitations. For instance, each collaborative robot can only serve one machine because of lacking mobility, which leads to high costs for all factory machines to satisfy all machine-tending tasks [13]. Therefore, autonomous mobile manipulators play a vital role in shaping intelligent manufacturing and addressing the increasing demands to deal with more complex machine-tending tasks and interact with unstructured and challenging environments by extending the capabilities of collaborative robots [15].

So far, an intelligent framework for autonomous mobile manipulators that will meet the high-level and complex requirements and apply to different machine-tending tasks, such as self-recognized commands and execution autonomously, in changing environments is still lacking [30]. To achieve the mentioned tasks or similar tasks, mobile manipulators need to have the ability to achieve scene text recognition that can help machines autonomously analyze and understand the corresponding environments, object detection that helps machines identify the target buttons that can be used in the operation process, object location that helps mobile manipulator sense the target buttons, and motion planning that allows the mobile manipulator to execute machines' commands autonomously.

In the manufacturing industry field, most traditional object detection methods extract handcrafted features from the image such as scale-invariant feature transform (SIFT) [191] or speeded-up robust features (SURF) [192], to match the templates of targeted objects. However, these methods depend on objects' known features and position; they cannot be applied to detect objects with unknown features. Since convolutional neural networks can learn robust and high-level features from images, CNN-based methods, such as image classification [193], semantic segmentation [194], [195], instance segmentation [196], [197] and object detection [198], have been used in intelligent manufacturing to achieve object recognition. Once the target objects are detected from an unstructured environment via object detection methods, the bounding box will be used to determine the center point of the target objects. Currently, many object detection-based manufacturing applications have been developed.

In [61], a convolutional neural network was developed for tool condition monitoring applications in machine shops. This method can distinguish the conforming and non-conforming producing tools based on tool change policy and pre-trained model. In [148], a Mask-RCNN model was proposed to detect the damages on the pipeline and obtain the 2D pixel coordinates of the damaged segments for remanufacturing applications. In [30], a SiameseRPN method was developed to detect each specific CNC machine in machine shops, and it can be further used in intelligent machine-tending applications in smart factories for mobile manipulators to fulfill machine-tending tasks between machines. In [199], the authors developed a damage detection method based on the Faster R-CNN model, which can be used in laser cladding repair processes. This method achieved an accuracy of 88.7% in identifying the damaged area on the cylindrical component and a time reduction of 63% compared with current industrial practice.

Unlike general object detection tasks, text detection and recognition have more challenges, such as being more sensitive to varying aspect ratios and blurring [186]. Although CNN has proven good performance in recognizing general objects, it is unsuitable for text detection because of its various aspect ratios. Feature maps extracted from convolutional layers need to be transformed into sequential representations to avoid being affected by the length variation. Therefore, the recurrent neural network (RNN) [200] is adopted to overcome this problem. In [180], a TextBoxes method was proposed to detect text from scene images in arbitrary orientation. It can achieve both high accuracy and efficiency using deep convolutional neural networks as core architecture and default rectangles with different specifications. However, it fails to handle some cases, such as large character spacing because it is a character-based or word-based recognition. In [201], a connectionist text proposal network method is proposed which is more robust and suitable for different aspect ratios. It applies recurrent neural networks with convolutional networks which can generate character labels in the sequence, but it still suffers from the word-based text spotting limitation. To improve the text spotting speed and efficiency that is significant for practical applications, an efficient text detection method that improves efficiency while retaining accuracy was proposed in [202]. It achieves the leading performance on the speed with an FPS of 16.8. Since character segmentation is the main challenge for text recognition, a method named connectionist temporal classification (CTC) [203] was proposed. It can train the model using sequence labels. So far, many studies [204], [205] on recognition have been done by combining different neural networks with CTC to generate the feature sequences, and then predict feature sequences as a label distribution. Finally, translate those frame predictions into a final text label.

Benefits from the research on optical character recognition and scene text recognition, many text recognition-based applications have been developed in the manufacturing industry, such as recognizing the date of packaged goods and reading serial numbers of products [206]. In [207], the author proposed a connected-component-based character recognition algorithm for a display reader application to detect digits from a LED display, it suffers from some challenges, such as glare on the screen and low contrast of the images.

For most real-world text recognition applications in the manufacturing industry, some limitations and challenges have always been there affecting the accuracy and efficiency, like blurring and reflections. In addition, for practical applications, recognition speed and the robust are always crucial for real-time mobile systems.

Considering the aforementioned problems and challenges, this chapter develops a command recognition method by using preprocessing techniques to reduce the influences of image blurring and screen reflection. A postprocessing process is also used to improve the accuracy of the recognition. After recognizing and understanding the machine commands, a YOLOv7-based button detection method is applied to identify and localize target buttons related to commands for further manipulator operation. The chapter also proposes an intelligent operation framework for autonomous robot-based machine-tending applications based on scene text recognition and object detection techniques to achieve fully autonomous manufacturing production in smart factory environments.

Besides the Introduction section, this chapter is organized as follows. Section 6.2 presents the related work; Section 6.3 demonstrates the experimental setup. In Section 6.4, the methodology and main process of the proposed intelligent framework are described in detail. Section 6.5 presents the experiments and results to validate and evaluate the developed methods. Section 6.6 discusses the limitation of the study, and Section 6.7 concludes this research.

6.2 Related Work

This section presents the recent and relevant real-time object detection and scene text recognition methods that have been used in the manufacturing sector, especially in assisting in manufacturing tasks or machine-tending-related tasks.

Optical character recognition (OCR) or scene text recognition (STR) has recently been developed in manufacturing applications. In [208], authors use CRNN to recognize the tire text for anomaly detection, which can achieve an accuracy of 96% and show the possibility of being applied to real manufacturing applications such as tire quality inspection of online production. In [209], a PP-OCR method was proposed to read industrial sticker information, which can achieve an accuracy of 88%. This method adopts differentiable binarization [210], a segmentation network, a text detector, and CRNN as a text recognizer. In [86], an optical character recognition system is proposed to extract the printed identification on steel coils, which uses traditional image processing techniques to extract the features of captured images with text information. Then the convolutional neural network is used to recognize each single character image. This proposed method can achieve an accuracy of 98%. In [30], authors developed a text recognition method to identify the machine's working status. First, a projection-based segmentation was proposed to detect the text and non-text areas from captured images and extract the region of interest, which includes the text. After detection, a CRNN is applied to recognize the detected text, and this method can achieve an accuracy of 85.7%.

Real-time object detection and position estimation have also been hot research topics in manufacturing applications. Compared with two-stage deep learning methods, one-stage methods, especially YOLO variants, have been the state-of-the-art real-time object detectors due to their high compatibility with industrial requirements, such as lightweight, faster and robust architecture, more accurate detection performance, more efficient label assignment, more efficient training method and friendly deployment conditions [145]. In [22], YOLOv3 and transfer learning were used to detect target objects to assist in pick-and-place through a collaborative robot. This method can achieve an accuracy of 95.9% in detection and 79.9% in position estimation and grasping. In [58], YOLOv4 was adopted to detect the different types of industrial objects, achieving an accuracy of 86.3%. In [31], YOLOv5s is used to visually identify three different 3D printed industrial parts in uncontrolled conditions, which achieves 97.65% accuracy and 35ms runtime.

These methods have greatly contributed to manufacturing or machine-tending tasks, but some limitations are observed. For instance, most research has been done in a simulation or a lab environment with ideal conditions such as simple backgrounds and perfect imaging conditions

[186]. Although some latest text recognition methods [211], [212] have been proposed in the computer vision domain, they mainly focus on specific issues, such as multilingual or multi-oriented text recognition, which is not within the research scope of this study. Although the current state-of-the-art real-time object detection method for manufacturing applications is mainly based on YOLOv7, there is no benchmark dataset and proper model for button detection tasks in the CNC machining process. Considering these issues, this paper proposed an intelligent operation framework to tackle machine-tending tasks. This framework developed a text recognition approach by integrating image preprocessing, region proposal network, and recurrent neural network to recognize and understand commands from CNC machines. In addition, a proper dataset consisting of five different types of buttons from the Haas CNC machine was created, and a YOLOv7-based model was trained and tuned for button detection and localization for further manipulator operation.

6.3 System Setup

In this study, the experimental setup is shown in Figure 6.1. The Ufactory mobile manipulator and a Haas CNC machine control panel are used for the case study. A Logitech webcam with a resolution of 1080 × 720 is utilized to capture the frames from the CNC display, which will be used for command recognition. An Intel RealSense Depth Camera D455, shown in Figure 6.2, is mounted on the end-effector of the robot manipulator for target button detection and localization. Its specifications are presented in Table 6.1, and it is observed its ideal range is from 0.6m to 6m.

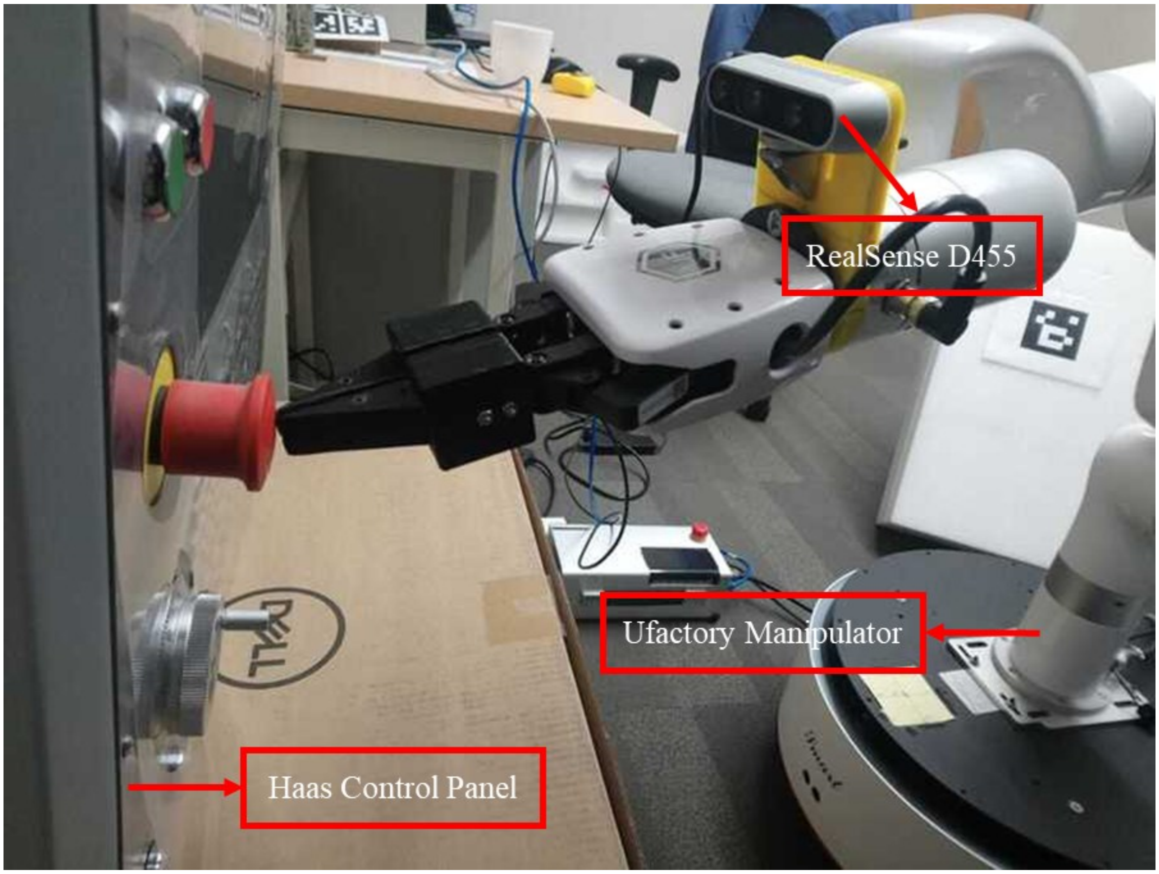

Figure 6.1: The experimental system.

Figure 6.2: The Intel RealSense Depth Camera D455.

Table 6.1: The specifications of the Intel RealSense Depth Camera D455.

Specifications	**Value**
Depth technology	Stereoscopic
Depth output resolution	1280 × 720

Depth frame rate	90 fps
RGB frame resolution	1280 × 800
RGB frame rate	30 fps
Ideal range	0.6 m to 6 m
Length × Depth × Height	124 mm × 26 mm × 29 mm
Connectors	USB-C* 3.1 Gen 1*

6.4 Research Methodology

The premier task in this study was to control the mobile manipulator to reach the target button successfully. The block diagram in Figure 6.3 illustrates the three sequential research modules. Each module and the elements are demonstrated in the next subsections.

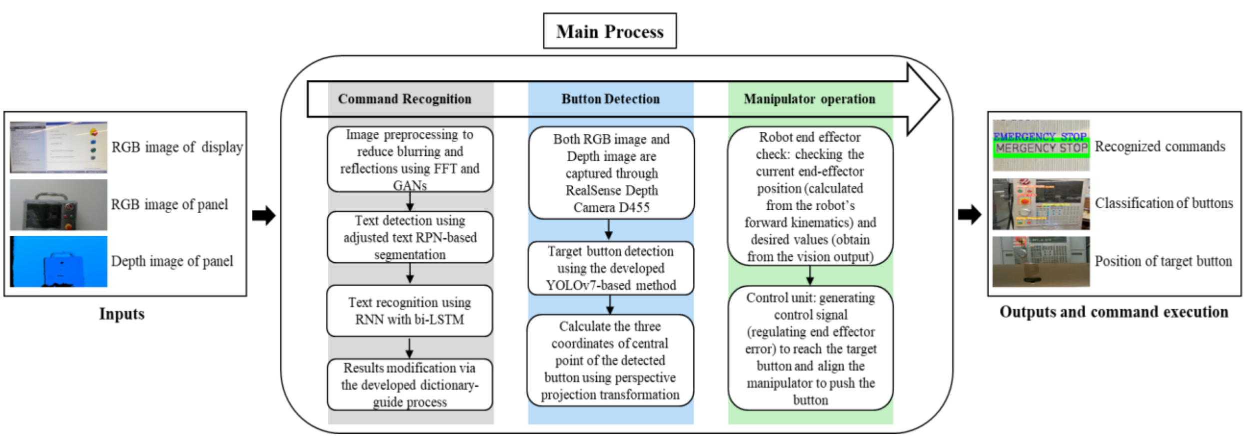

Figure 6.3: Research methodology outline.

6.4.1 Command Recognition

This section describes the main command recognition steps, depicted in Figure 6.4.

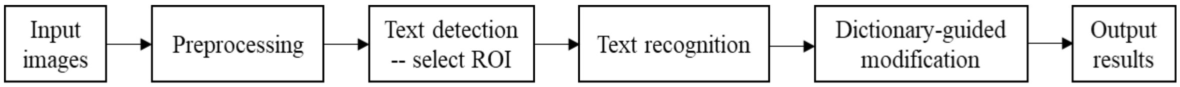

Figure 6.4: The main steps of command recognition.

6.4.1.1 Fast Fourier Transform (FFT) to deblur the image

Image blurring is a common problem caused by camera movement or jitter, and it causes a decline in image quality. In addition, low illumination conditions often require longer exposure time for the camera, which leads to a slight jitter. Therefore, image deblurring is important for practical applications in industry. Many deblurring algorithms have been developed recently, and the Fast Fourier Transform (FFT) is one of the most widely used image processing techniques. It is a convenient mathematical algorithm for computing the Discrete Fourier Transform (DST), which is the sampled Fourier Transform and is good enough to describe the spatial domain image fully. The principle of FFT is converting an image from its spatial domain into a representation in the frequency domain. By doing so, noises can be effectively mitigated or eliminated by instituting an amplitude-based threshold. Following this filtration, the processed image can subsequently be converted back to the spatial domain [213]. Therefore, the FFT is widely used in many applications, such as image compression, image filtering and image reconstruction. In this study, an image deblurring method based on FFT [213] is applied to calculate the blur filter kernel which is used to remove the noise and improve the quality of images. Figure 6.5 shows an example of image deblurring.

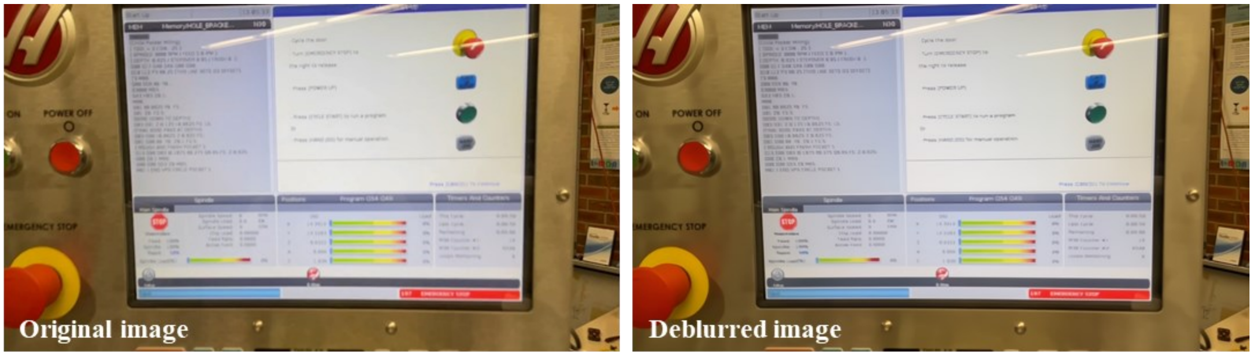

Figure 6.5: The example of image deblurring via FFT.

6.4.1.2 Reflection Removal via Generative Adversarial Networks (GANs)

Natural scene text recognition has become popular in manufacturing applications; however, it suffers from many problems affecting performance. Blurring and undesirable reflections are the main reasons causing the low quality of images, which decrease the performance of computer vision applications such as detection and recognition. When capturing images through a transparent material such as glass, screen, or display, the images often contain undesirable reflections. These reflections lower the quality and visibility of the images. There are some traditional methods to solve these problems. The camera can shoot the images in a dark environment or manually adjust the camera position and the shooting angle. However, these methods are inefficient and not applicable for practical applications under bright conditions and autonomous environments. Recently, many approaches have been developed to achieve reflection removal and they are generally divided into two categories: conventional methods and deep learning-based methods. Conventional methods adopted handcrafted features to separate the background and reflections in specific conditions [214]. Deep learning-based methods, particularly, GAN-based methods have achieved good results in reflection removal [215]; generative adversarial networks are a learning method that maps noise to an image.

In this study, the image is denoted as $\boldsymbol{I}$. The background layer is denoted as $\boldsymbol{B}$, and the reflection layer is denoted as $\boldsymbol{R}$. Then, the image $\boldsymbol{I}$ can be modelled as a linear combination of background $\boldsymbol{B}$ and reflection $\boldsymbol{R}$, shown below:

$$\boldsymbol{I} = \boldsymbol{B} + \boldsymbol{R} \tag{6.1}$$

This section applies a GAN-based method with gradient constraint [215] to remove the undesired reflection from the captured images by separating the background and reflection layer. In addition, gradient constraint loss is used to minimize the correlation between the background and reflection layer, which can remove reflections effectively in real-world images and achieve outstanding performance. Figure 6.6 shows an example of reflection removal.

Figure 6.6: The example of reflection removal.

6.4.1.3 Text Detection and Recognition

This study proposes a command recognition approach to accurately and efficiently identify the text information from the CNC machine display by combining the text detection branch and the text recognition branch.

The proposed text detection branch consists of three subnetworks. First, the adjusted region proposal network (RPN) was developed to predict the text regions on the input images. Unlike generic objects, the text is a sequence that does not usually have a well-defined boundary and center because it contains multi-level components like characters, words or text lines, which are not distinguished clearly between each other. It is observed that the RPN [166] is difficult to accurately find the horizontal boundaries of words or text lines. Therefore, to achieve text spotting at the text-line level instead of a single character or part of words, it is important to consider a text line as a sequence of fixed-width text proposals, where each text proposal represents a part of a text line. Therefore, each proposal in the adjusted RPN is defined as a fixed width of 16 pixels, the same as the stride in the last convolution of the backbone architecture VGG16 [216]. Next, an anchor mechanism is developed in which 10 vertical anchors with fixed widths, and different heights are used to predict the text/non-text score (s) and vertical coordinates (v) of each text proposal. Text proposals are generated based on the anchors with a text/non-text score larger than 0.7. These text proposals can be considered as a feature sequence $\boldsymbol{x_t} = x_1, \ldots, x_w$ and further fed into RNN for encoding.

Given that generated text proposals are detected separately and independently, a recurrent neural network (RNN) [200] is adopted to encode the information and group them into sequential context information for text recognition. The main advantage of RNN is that can capture the context in a sequence, which is more useful than identifying each character individually because some ambiguous characters can be easily predicted by observing the contexts. In the RNN layer, a bidirectional long short-term memory (LSTM) architecture [217] was used, with 256D output channels per direction, to capture range dependencies of the input sequential features. Then, internal hidden states $\boldsymbol{h_t} = h_1, \ldots, h_w$ are updated at each time in both directions and then fed into a fully connected layer, and the output layer provides the predictions of the t-th text proposal.

Finally, multi-task learning is integrated jointly to optimize the model parameters. The overall text detection loss function L_{dec} consists of two parts: L_S^{cl} calculates the errors of the text/non-text scores, and L_v^{re} compute the errors of vertical coordinates. To minimize the L_{dec}, the rule in [166] is followed:

$$L_{dec} = \frac{1}{N_S} \sum_i L_S^{cl}(s_i, s_i^*) + \frac{1}{N_v} \sum_j L_v^{re}(v_j, v_j^*) \tag{6.2}$$

where i is the index of an anchor in a mini-batch. s_i is the predicted probability of the anchor being a true text region. s_i^* is the ground truth. j is the index of an anchor in the set of valid anchors for vertical coordinates regression. The valid anchor is defined as the intersection-over-union (IoU) is larger than 0.5. v_j and v_j^* are the prediction and ground truth of the y-coordinates. L_S^{cl} is the text classification loss to distinguish text and non-text regions, and L_v^{re} is the bounding box regression loss. Both losses can be calculated through smooth function L_1 in [166]. N_S and N_v are the normalization parameters, representing the total number of anchors used for L_S^{cl} and L_v^{re}.

Figure 6.7 shows some examples of command detection results using TextBoxes++ [218], EAST [202] and the proposed method from the left to the right, respectively. The detected results show that the proposed text detection method can achieve more accurate results and the best performance for line-text command detection.

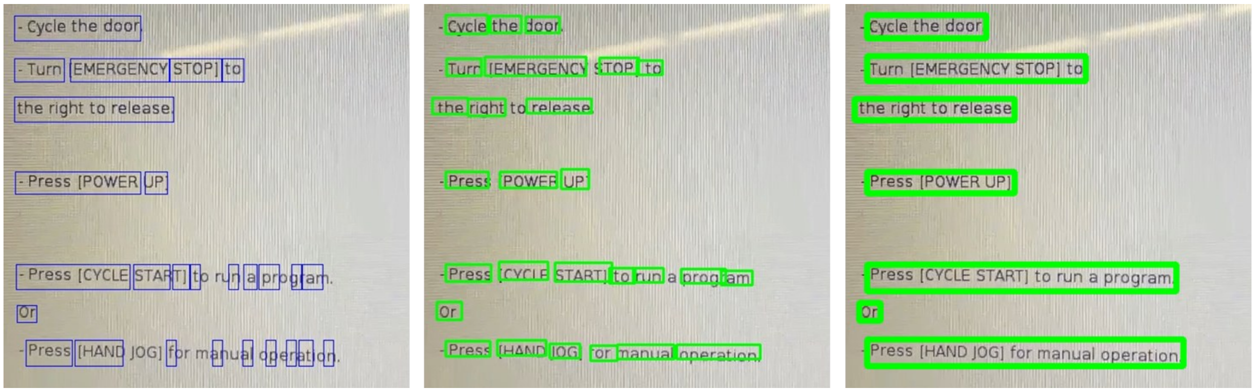

Figure 6.7: The examples of command detection methods.

After text detection, the text recognition branch predicts text labels based on the region features extracted in shared convolutions. This recognition branch shares the convolutional layers and recurrent layers in the text detection branch, followed by a CTC decoder [203].

The RNN layer in the detect branch can predict a label distribution $\boldsymbol{y_t} = y_1, \ldots, y_w$ based on the input sequential features. Then, the predicted label distribution is fed into the CTC layer, which is utilized to convert frame-wise classification scores to a label sequence with the highest conditional probability. Here, the conditional probability defined in CTC [203] is used for the ground truth label sequence $\boldsymbol{l} = \{l_1, \ldots, l_w\}$ conditioned on the predictions $\boldsymbol{y_t} = y_1, \ldots, y_w$. The conditional probability is the sum of probabilities of all paths π that are mapped by β onto $\boldsymbol{l}$, conforming to [203]:

$$p(\boldsymbol{l}|\boldsymbol{y}) = \sum\nolimits_{\pi:B(\pi)=l} p(\boldsymbol{\pi}|\boldsymbol{y}) \tag{6.3}$$

where is a sequence-to-sequence mapping function that defines a many-to-one map from the set of possible labels by removing the repeated labels first, and then removing blank labels. The objective of the training process is to maximize the log-likelihood of the conditional probability of ground truth. The text recognition loss L_{rec} is calculated by Equation 6.4:

$$L_{rec} = -\frac{1}{N}\sum_{n=1}^{N} log\ p(\boldsymbol{l}|\boldsymbol{y}) \tag{6.4}$$

where N is the number of text regions in an input image, therefore, the total loss is the adding up the detection loss L_{det} and the recognition L_{rec}:

$$L = L_{dec} + \lambda L_{rec} \tag{6.5}$$

where λ is a hyper-parameter, which controls the trade-off between two losses. In this study, λ is set to 1. Figure 6.8 shows the recognition results based on CRNN [177] and the developed command recognition method.

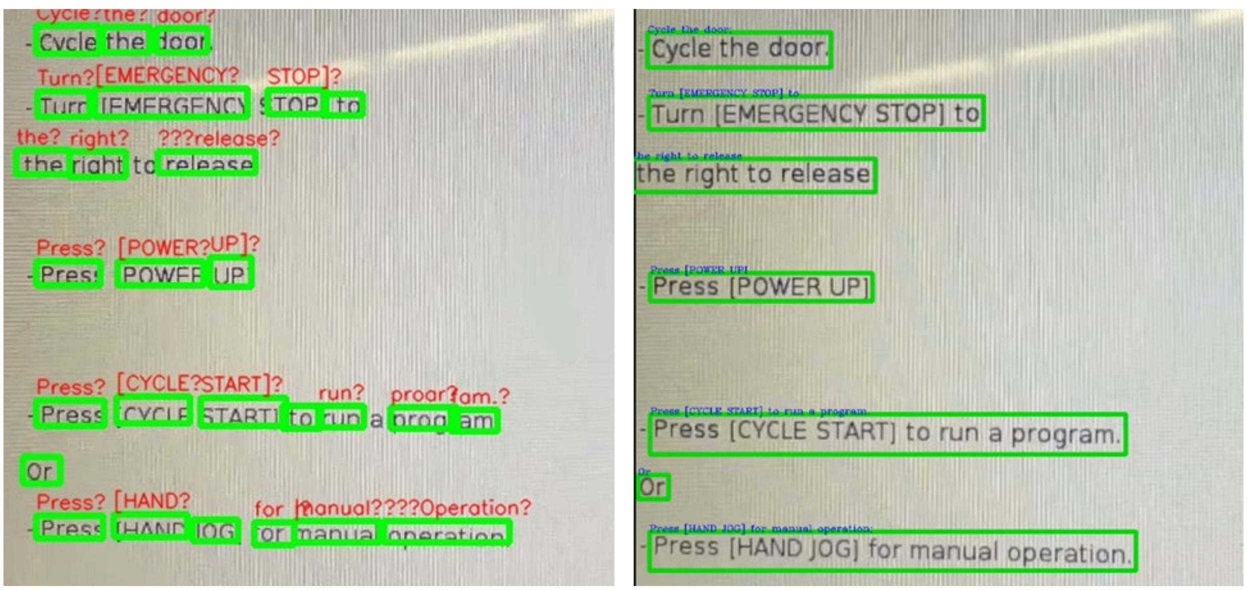

Figure 6.8: The examples of command recognition methods.

6.4.1.4 Dictionary-guided Modification

Text recognition accuracy is impossible to be 100% accurate every time using a camera in real-world scenarios because it is not only affected by the illumination conditions in the environment but also by the position and angle of the camera. Therefore, the modified method to automatically correct the outputs when recognition results are wrong is necessary. In this study, common instructions shown on the CNC display such as “Turn Emergency STOP to release”, “Press POWER UP”, “Press CYCLE START to run a program”, and etc., are used to create a data list in the dictionary. Once the command on the screen is recognized, the output is modified by comparing it with the instructions listed in the dictionary via cosine similarity metrics, which is shown in Equation 6.6 [219].

$$\boldsymbol{si}\text{milarity}\ (\mathbf{A},\ \mathbf{B}) = \cos(\theta) = \frac{\mathbf{A}\cdot\mathbf{B}}{\|\mathbf{A}\|\ \|\mathbf{B}\|} = \frac{\sum_{i=1}^{n} A_i B_i}{\sqrt{\sum_{i=1}^{n} A_i^2}\ \sqrt{\sum_{i=1}^{n} B_i^2}} \tag{6.6}$$

where $\boldsymbol{A}$ and $\boldsymbol{B}$ represent the vectors of the predicted commands and ground truth, respectively. Then set a threshold of 0.5 for the similarity as the standard metric. If the cosine similarity is larger than 0.5, replace the recognized results with the text in the dictionary. Therefore, this check and correct process can fix wrong recognition problems. Figure 6.9 shows the example of the dictionary process, the left image is the recognized result and the right image is the output after dictionary-guided modification.

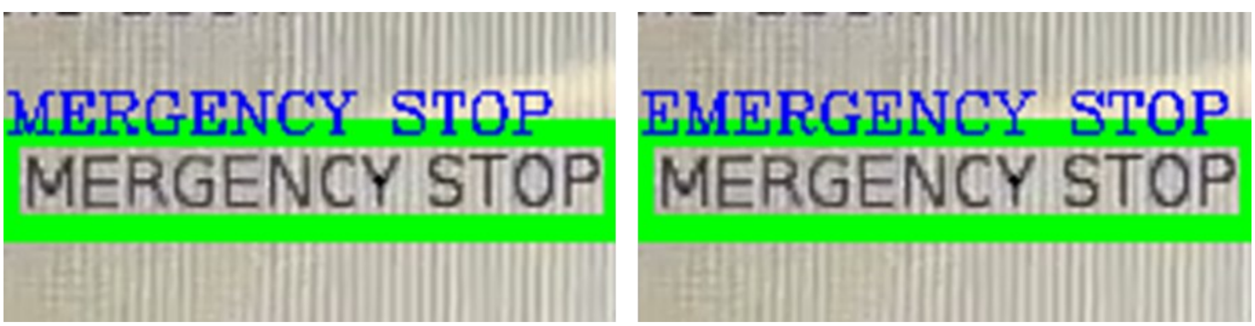

Figure 6.9: Recognition result correction through the dictionary process.

6.4.2 Button Detection and Localization

This section explains the target button detection and localization process, which is shown in Figure 6.10.

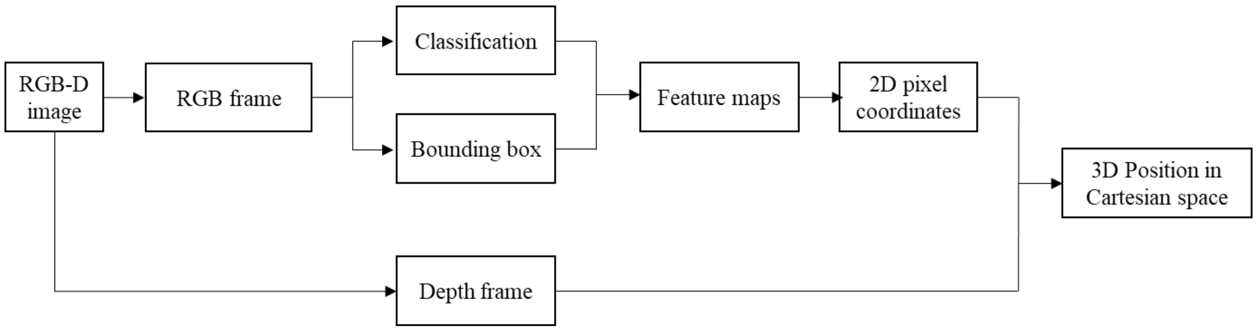

Figure 6.10: The main process of target button detection and localization.

6.4.2.1 Target Button Detection through YOLOv7

YOLOv7 has been proved the excellent performance for real-time object detection in terms of both speed and accuracy, and it can be trained much faster on small datasets without any pre-trained weights [76]. However, it has not been applied to detect the buttons on CNC machines in manufacturing-related applications. Therefore, a benchmark of the YOLOv7 on the CNC machine button datasets is essential to observe and investigate its performance on button detection tasks. In this study, the YOLOv7-based method is developed to detect the buttons on the Haas CNC machine for further robot manipulator operation.

YOLOv7 is a one-stage model with three main components: backbone, neck, and head [144]. The backbone extracts image feature maps and transfers them to the neck layers. These feature maps are then combined, fused, and passed to the subsequent layers. Finally, the head network predicts the bounding boxes and classes of the objects. YOLOv7 adopts a developed Extended Efficient Layer Aggregation Network (E-ELAN) to improve inference efficiency. It can enhance learning ability without disturbing or changing the original gradient propagation path. In addition, a novel model scaling method for concatenation-based models, named corresponding compound model scaling, is proposed to address the issue of a larger width output of the computational block. Moreover, several techniques have been used to improve prediction accuracy while keeping training costs low. These strategies, named Bags of Freebies (BoF), include planned re-parameterization convolution, dynamic label assignment, and batch normalization. After thoroughly investigating how the re-parametrized convolution is combined with other different networks, it is observed an increase in model accuracy when using the RepConv without an identity connection (RecpConvN). When training the deep networks, two heads, the lead head, and the auxiliary head, need to be used. The lead head represents the final output of the model, and the auxiliary head is used to assist in training. Previously, the most popular methods used two heads separately and then used their predictions and the ground truth to distribute soft labels. However, YOLOv7 proposed a novel label assignment method that guides both heads. Here, two types of label assigners were developed. One is the lead head guided label assigner, where the soft label is mainly generated based on the lead head and ground truth. The other is a coarse-to-fine lead head guided label assigner, where two different sets of soft labels, coarse label and fine label, are generated, where the fine label is the same as the soft label generated in the lead head guided assigner, while the coarse label is generated through relaxed rules on the positive sample assignment process. Furthermore, batch normalization integrates the mean and variance of the data to adjust the bias and weight of the convolutional layer, which can improve the training process by using a higher training rate and faster convergence [220]. This chapter mainly focuses on optimizing the training process to obtain better button detection accuracy and speed and precise localization of the target buttons in the next step.

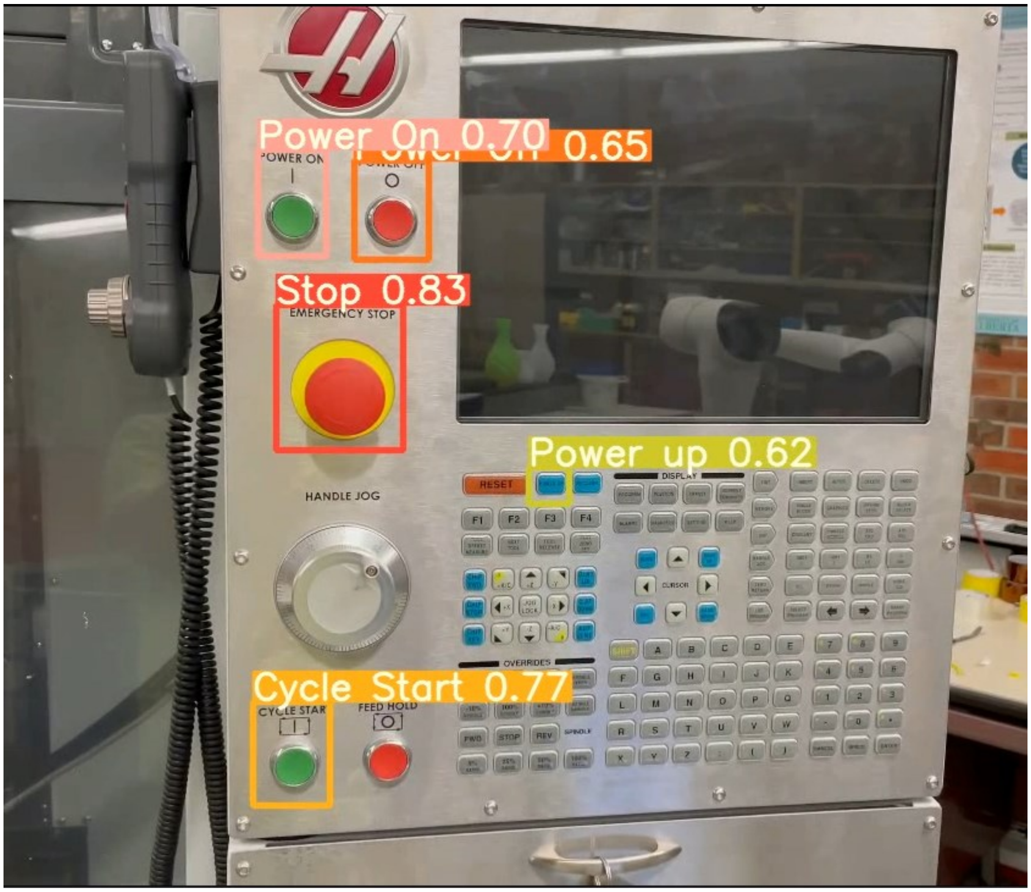

Figure 6.11: Examples of five different types of button detection.

6.4.2.2 *Calculate Three Coordinates of the Central Point of the Detected Target Button*

In this study, perspective projection transformation [221] is used to determine the 3D coordinates of the central point of the detected target button from the scene image. From the detected results, the 3D coordinates of the target point can be calculated from the equations below. From the depth camera image, the pixel's depth value with the central point *Z* can be determined. By substituting the *Z* value in the equation below, the *X* and *Y* coordinates of the target point can be obtained through the equations below:

$$X = Z\,\frac{u - c_x}{f_x} \tag{6.7}$$

$$Y = Z\,\frac{v - c_y}{f_y} \tag{6.8}$$

where f is the focal distance, (u, v) is the 2D image coordinates, (c_x, c_y) is the optical center of the camera, and (X, Y, Z) is the scene point position in the world coordinate system.

However, only using one-point coordinates could be unreliable because it is will be calculated incorrectly. In this study, ten central points will be measured through ten frames and then used to calculate the average position of the center. Even if several points have abnormal values as compared to other points, it is easy to exclude these abnormal points from the calculation to find the 3D coordinates of the center point through the equation below:

$$c = \frac{1}{N}\sum_{i=0}^{N}(x_i\ y_i\ z_i\ ) \tag{6.9}$$

where c is the average 3D coordinates of the central point for the detected target button. The example of detecting and localizing the emergency button from the Haas CNC machine control panel is shown in Figure 6.12.

Figure 6.12: The examples of emergency button detection and localization.

6.4.3 Control Scheme for Manipulator Operation

The robotic manipulator executes the task by using the end-effector to press the instruction buttons. It uses the information on the target button position obtained previously. Using the developed control scheme, position control of the endpoint of the manipulator is possible when the end effector moves to the desired position.

To achieve the successful execution of an interaction task, such as pushing a button using motion control, accurate planning is crucial. This planning necessitates a precise model of both the robot manipulator and the environment. While achieving a precise manipulator model is feasible, obtaining a detailed description of the environment is challenging.

For the analysis of the interaction between the manipulator and the environment, it is beneficial to examine the system's behavior under a position control scheme known as active compliance control [222]. The primary objective of such a controller is to achieve suitable active compliance that can be easily adjusted through the control software to fulfill the requirements of button-pushing tasks.

To derive a control law to push the button, an error definition in the operational space must be taken into account. Let's define the end-effector's rotation matrix as R_e and the position of the end-effector as o_e. Consequently, the homogeneous transformation matrix for the end-effector, T_e, can be expressed as follows:

$$T_e = [R_e \quad o_e \quad 0^T \quad 1\,] \tag{6.10}$$

Similarly, the desired end-effector's corresponding transformation matrix can be defined as:

$$T_d = [R_d \quad o_d \quad 0^T \quad 1\,] \tag{6.11}$$

Therefore, the relative transformation matrix of the end-effector with respect to the desired frame, T_e^d, is expressed as:

$$T_e^d = (T_d)^{-1} T_e = \left[R_e^d \quad o_{d,e}^d \quad 0^T \quad 1\right] \tag{6.12}$$

where

$$R_e^d = R_d^T R_e \text{ and } o_{d,e}^d = R_d^T(o_e - o_d) \tag{6.13}$$

Having the above relative rotation and position, the operational space error can be found as:

$$\tilde{x} = -\left[o_{d,e}^d \quad \phi_{d,e}\ \right] \tag{6.14}$$

where $\phi_{d,e}$ denotes the Euler angles of the relative rotation matrix.

Having the error in operational space and the robot's gravity matrix, the proportional derivative control law with gravity compensation term can be obtained as:

$$u = g(q) + J_{A_d}^T(q, \tilde{x})\big(K_p\tilde{x} - K_D J_{A_d}(q, \tilde{x})\dot{q}\big) \tag{6.15}$$

where J_{A_d} is the desired analytical Jacobian which can be calculated as:

$$J_{A_d}(q, \tilde{x}) = T_A^{-1}\big(\phi_{d,e}\big)[R_d^T \;\; 0 \;\; 0 \;\; R_d^T \;\;]J(q) \tag{6.16}$$

here, T_A denotes the analytical transformation matrix and J is the geometric Jacobian which depends on the robot's joint type.

$$T_A\big(\phi_{d,e}\big) = [I \; 0 \; 0 \;\; T(\phi_e) \;\;] \tag{6.17}$$

In the absence of interaction, the stability of the proposed control law can be proved using the following Lyapunov function [222]:

$$V(\dot{q}, \tilde{x}) = \frac{1}{2}\dot{q}^T B(q)\dot{q} + \frac{1}{2}\tilde{x}^T K_P \tilde{x} > 0 \tag{6.18}$$

where B denotes the inertia matrix of the robot's dynamics model presented by the Lagrangian method.

6.5 Results of the framework implementation

This section presents the performance of the proposed method and compares it with other existing similar methods. In addition, the validation and performance estimation of the proposed methods are presented through the case study.

Since there are no public datasets for command texts and control buttons from CNC machines, a specific dataset for each other is essential and built for the experiments. The images with commands and five different control buttons were collected using the Logitech camera. The created datasets for machine instructions and control buttons have 200 images and 1000 images with a resolution of 1080 × 720, respectively. Both datasets are split into three sub-datasets: 70%

training images, 20% validation images, and 10% testing images. These images in the dataset were annotated using LabelImg Software, which is an open-source annotation tool.

Both models for detecting machine commands and control buttons were trained and tested on a local Desktop with the specifications listed in Table 6.2. The pre-trained hyper-parameters are presented in Table 6.3.

Table 6.2: Training environment and computer specifications.

Specifications	**Value**
Operating System	Windows Server 2019
CPU	AMD Ryzen Threadripper 3970X 32-Core
GPU	NVIDIA GeForce RTX 3090
RAM	128 GB
CUDA Version	12.0
PyTorch Version	1.12.0

Table 6.3: Training Parameters.

Parameters	**Value**
Learning Rate	0.001
Learning Momentum	0.9
Batch Size	8-32
Epochs	100

This paper adopted the mean average precision (mAP) as the evaluation metric. It is the area under the precision and recall (true positive rate) curve calculated by Equation 6.18 at different intersection-over-union (IoU) thresholds. mAP_0.5, at a 0.5 intersection-over-union (IoU) threshold, is commonly used as the evaluation metric. In addition, mAP_0.5:0.95, which is the average mAP over multiple IoU thresholds, can affect the model with better performance. Therefore, both metrics will be considered in the training and testing procedures to evaluate the performance of developed methods.

$$IoU = \frac{Area\ of\ Overlap}{Area\ of\ Union} \tag{6.19}$$

$$AP = \int_0^1 (Precision\ \times Recall)\ d(Recall) \tag{6.20}$$

$$Precision = \frac{TP}{TP + FP} \tag{6.21}$$

$$Recall = \frac{TP}{TP + FN} \tag{6.22}$$

$$F1 - Score = \frac{2 \times Precision \times Recall}{Precision + Recall} \tag{6.23}$$

where *TP*, *FP*, and *FN* represent true positive, false positive, and false negative of the classification and predicted bounding box, respectively.

6.5.1 Performance of the Developed Command Detection and Recognition Method

This study adopted two hundred scene text images collected from the Haas CNC machine display to validate the developed command recognition method. The proposed method is compared with the previous works and other state-of-the-art methods, and the results are shown in Table 6.4 and Table 6.5. It can be concluded that the proposed method can achieve an accuracy of 100% in command recognition which improves text recognition performance in accuracy compared to other similar methods without the loss of efficiency. For the training process, 100 iterations were executed in 1.5 hours and the validation loss and precision were shown in Figures 6.13 and 6.14. Although different learning rates and batch sizes were tried during the training process, there were no significant changes in model precision and loss.

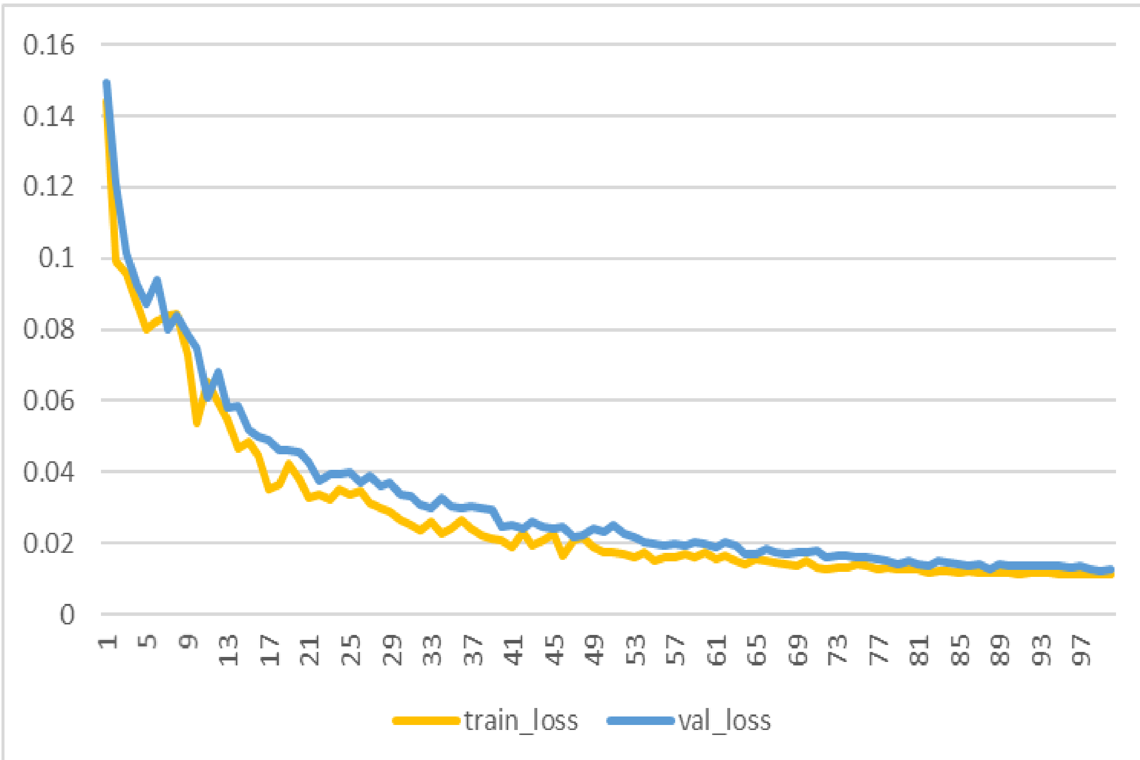

Figure 6.13: The training and validation loss of the proposed command recognition method.

Figure 6.14: The training and validation accuracy of the proposed command recognition method.

Table 6.4: Comparison results on different command detection methods.

Method	**P**	**R**	**F**
Fast R-CNN	0.79	0.71	0.75
TextBoxes++	0.92	0.86	0.89
EAST	0.83	0.78	0.80
Proposed	**0.99**	**0.98**	**0.98**

For command recognition, the result of the developed method is compared with the previous work and some similar relevant works, and the comparison is presented in Table 6.5.

Table 6.5: Comparison results on different command recognition methods.

Method	**Accuracy**	**Speed (second/image)**
CRNN	84%	0.3
Previous	97%	0.15
Proposed	**100%**	**0.12**

6.5.2 Performance of the Developed Button Detection Method

During the training process for the proposed button detection model, 1000 images were used as a dataset, 70% was used as the training set, 20% was used as the test set, and 10% was used as the validation set. 100 epochs were executed in 4.82 hours and the performance of the button detection method is presented in Figure 6.15. Although different learning rates, batch sizes and different iterations were tried during the training process, there was no significant change in the precision and loss, only changes in the rate of convergence. In addition, the summary of performance metrics of the proposed button detection method is shown in Table 6.6.

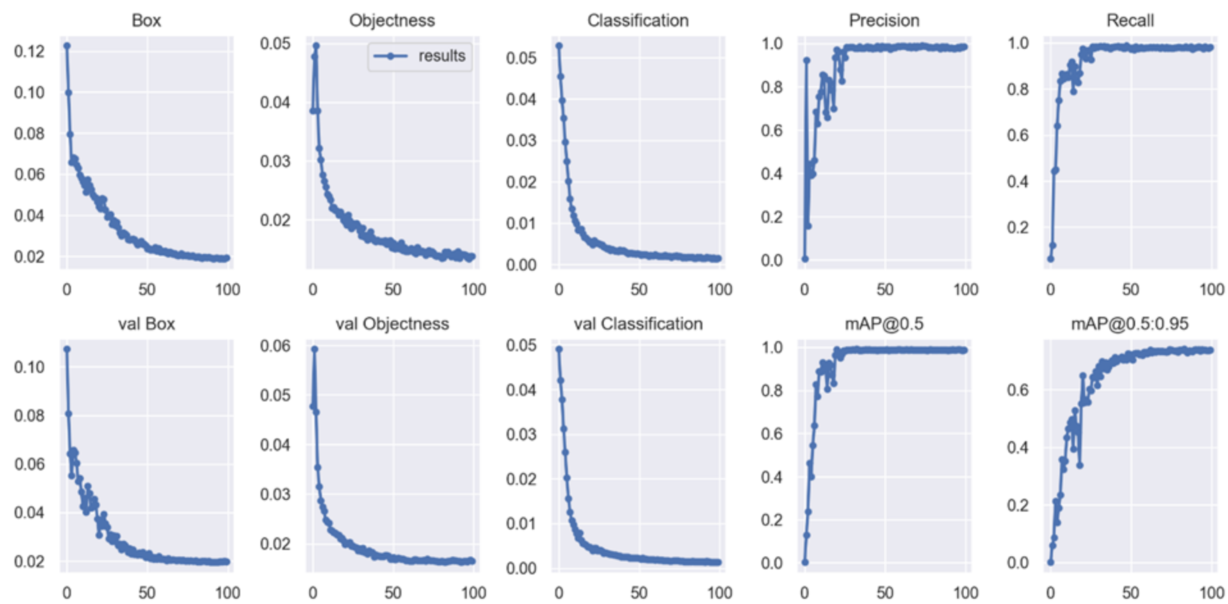

Figure 6.15: The performance of the proposed button detection method.

Table 6.6: Summary of the performance metrics of button detection.

Button Class	**mAP_0.5**	**mAP_0.5:0.95**	**Precision**	**Recall**	**F1-Score**
Emergency Stop	0.996	0.784	0.998	0.99	0.994
Power On	0.984	0.749	0.998	0.979	0.988
Power Off	0.978	0.719	0.98	0.968	0.974
Power Up	0.989	0.701	0.971	0.978	0.974
Cycle Start	0.993	0.738	0.982	0.982	0.982
Overall	**0.988**	**0.738**	**0.986**	**0.981**	**0.983**

In addition, a case study was carried out to validate the accuracy of the proposed button localization method. Moving the depth camera at five different positions to detect and localize the emergency stop button on the control panel, then recording the predicted position of the central point of the button and comparing it with the measured actual position of the button (ground truth). The results are presented in Table 6.7.

Table 6.7: Comparison results of predicted position with ground truth.

Case Number	**Ground Truth (mm)**	**Predicted Position (mm)**	**Reachable**
1	(-5, 25, 550)	(-6.9, 25.8, 551.7)	Yes
2	(40, 32, 203)	(41.6, 33.6, 205.3)	Yes
3	(100, 31, 607)	(102.1, 31.4, 610.5)	Yes
4	(-102, 22, 399)	(-100.7, 22.8, 402.8)	Yes
5	(41, 20, 401)	(44.3, 20.2, 405.1)	Yes

6.5.3 Performance of the Proposed Control Scheme

This case study elucidates the application of the button detection and localization methodology proposed in this paper, with a specific focus on navigating the robot end-effector towards a designated target: the emergency button. In the beginning, set up a fixed position (152, 76, 460) as the initial position of the end-effector of the manipulator. The robot will always start from its initial position, press the target button and then return to the initial position to wait to execute the subsequent operations.

Figure 6.16 provides a graphical representation of the end-effector's Cartesian coordinates (x, y, z) throughout this manipulator operation. It is observed that the robot arm is at state A and initiates movement from its initial position at approximately 3.7 seconds and succeeds in reaching the emergency button at (553,6, 179.6, 511.2) by 8.04 seconds, which is state B. Subsequently, the robot maintains its position momentarily until 10.5 seconds and then moves back to its initial position in preparation for subsequent operations, which is shown in state D.

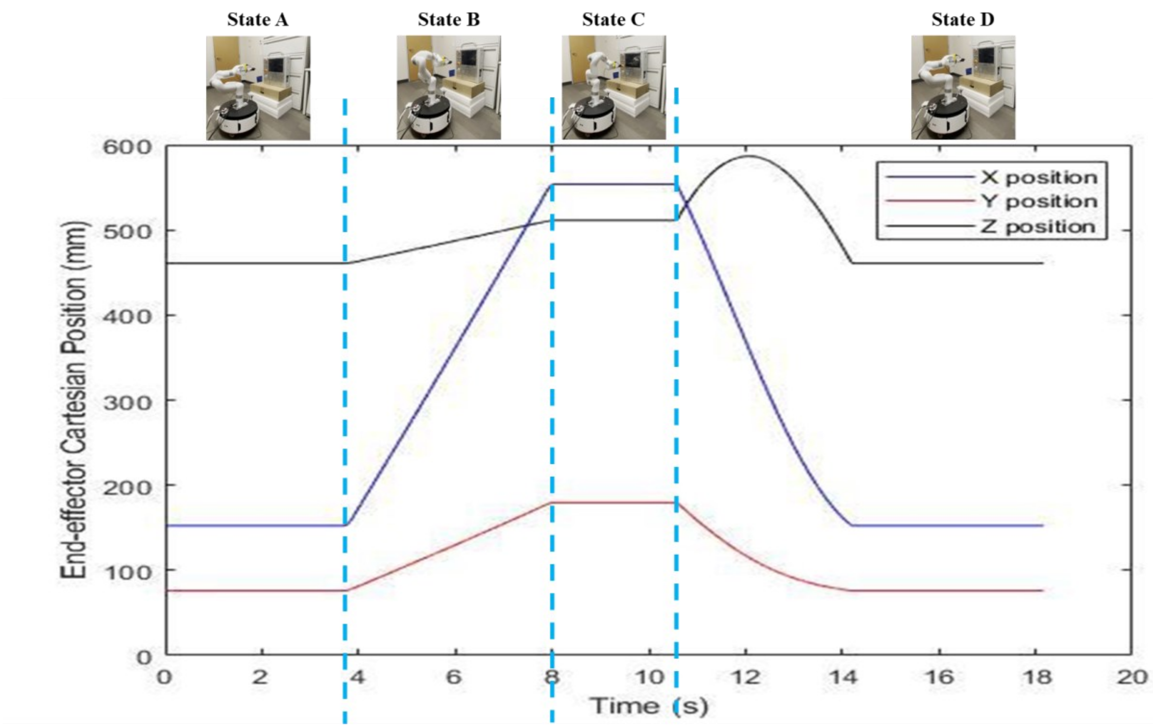

Figure 6.16: Cartesian coordinates of the end-effector position.

Additionally, Figure 17 illustrates the end-effector's rotations, expressed in terms of roll, pitch, and yaw. It shows that the initial values of roll, pitch, and yaw are 0, -85, and -180 degrees, respectively. When the robot arrives at the position of the emergency button, the orientation of the end-effector is 160, -78, and 30 degrees, respectively.

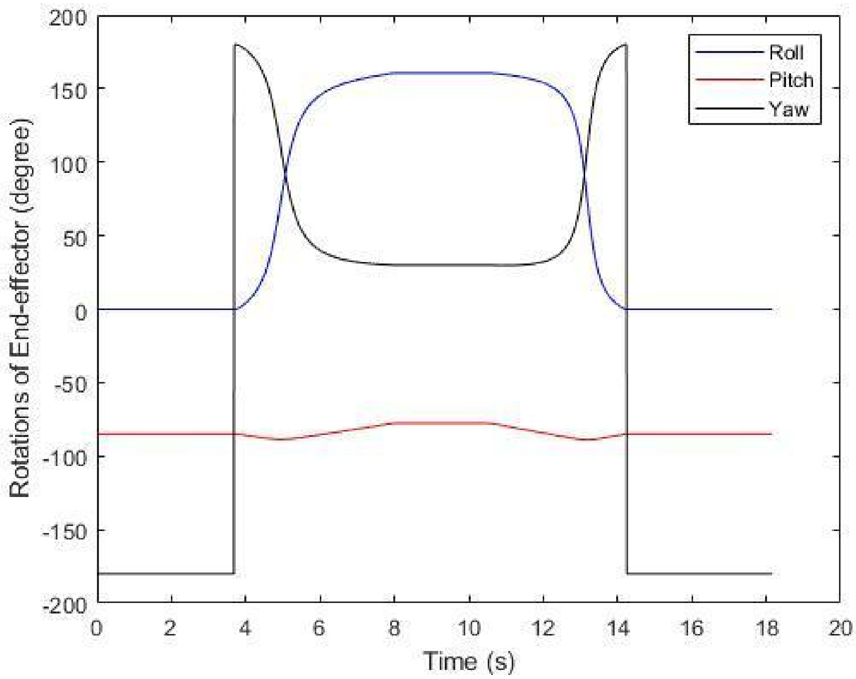

Figure 6.17: Orientation of the end-effector.

Figure 6.18 offers insights into the state of the robot's six joints during this operation. Before the experiment starts, the initial state of the six joints is (0,0,0,0,0,0). When the end-effector reaches the emergency button, the state of the six joints is (16.1, -94.5, -90.8, 0.6, 86.7, 0). Finally, the effector returns to the initial state.

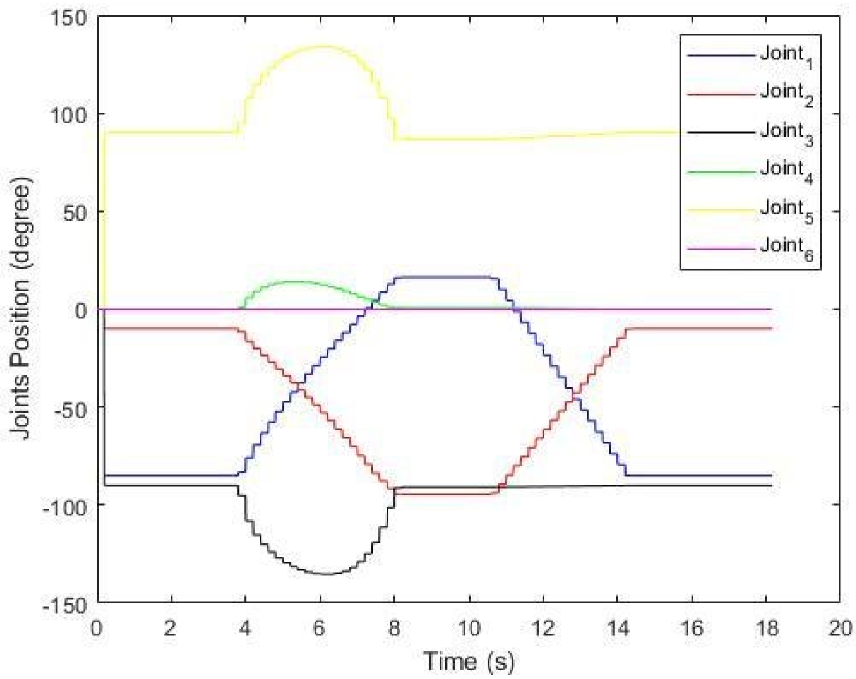

Figure 6.18: States of six joints.

6.6 Discussion

In summary, this chapter presents an intelligent operation framework for autonomous robotic machine-tending applications, which can perform machine-tending tasks with full autonomy, negating the need for human intervention.

Despite all the benefits of the proposed framework, certain limitations persist. Primarily, the case study conducted within this research was centered on the Haas CNC machine. Given the inherent dependency of deep learning-based methods on high-quality labelled training datasets, the developed button detection method might cause higher error rates when applied to other CNC machine brands. Additionally, this study solely incorporated five distinct control button categories from the Haas CNC machine, which might not be fully representative of various machines. Moreover, the framework is restricted to executing preliminary operations, such as navigating the manipulator to engage specific buttons in response to emergencies and lacks the capacity for mature and intricate decision-making. Figure 6.19 presents the temporal sequences of the robot manipulation in response to an emergency in the case study. It can be observed that the field of view of the depth camera does not encompass the entire control panel. Consequently, it necessitates posture adjustments to discern all significant buttons. As the robot manipulator advances towards the emergency button, the accuracy in detection and position estimation diminishes because of the constraint attributable to the intrinsic limitations of the RealSense depth camera D455. This leads the robot system to return to its initial position to obtain better results following each task completion.

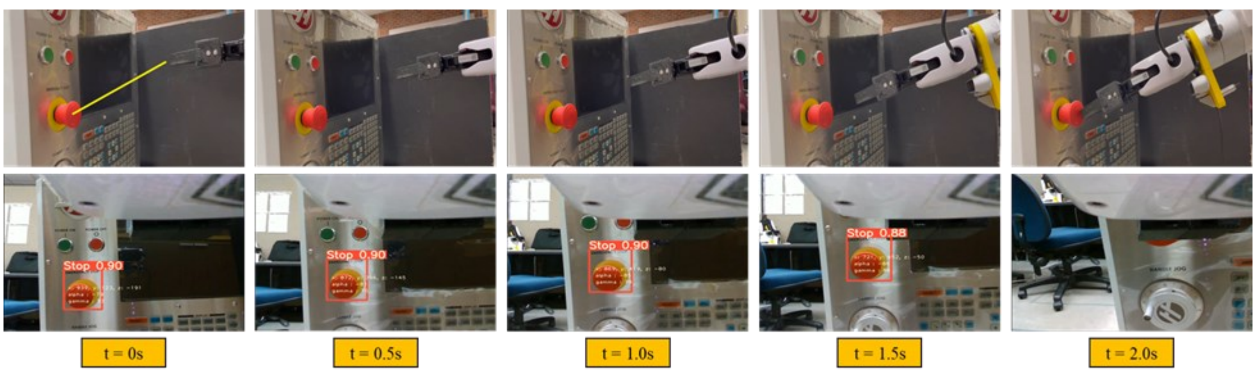

Figure 6.19: Temporal sequence of the robot manipulation.

6.7 Conclusion

This chapter discusses the major challenges confronting current robotic machine-tending systems and proposes an innovative framework for autonomous robotic machine-tending systems to overcome these impediments. In the proposed framework, an efficient text spotting approach is developed, which integrates the adjusted text region proposal network with recurrent neural network and bi-LSTM architecture. It facilitates the recognition and comprehension of the instructions from the CNC machine display. Furthermore, the study puts forth a YOLOv7-based approach for button detection and localization, aiming to identify critical control keys on the CNC panel and subsequently calculate their spatial coordinates. Additionally, an accompanying control scheme is presented, executing the recognized commands by operating the robotic machine-tending system to click the corresponding buttons. To validate its efficacy, the performance of the framework was compared with existing related works. The results revealed that the proposed methods enshrined in the developed framework have outperformed the existing methods in terms of precision and adaptability. Thus, the study indicates that this intelligent operation framework possesses the requisite accuracy and flexibility to be implemented to perform machine-tending tasks in real-world manufacturing scenarios and autonomous environments.

The future work will aim to augment the autonomous robotic machine-tending systems to encompass capabilities such as autonomous docking and recharging. This advancement enables robotic machine-tending systems to work continuously without interruption, thereby reducing human oversight. Furthermore, to enhance the versatility and efficacy of the proposed methods, the training and testing datasets will be enriched with images characterized by intricate backgrounds and various buttons derived from different CNC machine brands. Moreover, more decision-making strategies will be considered and incorporated. This will facilitate a more seamless, streamlined, and adaptive manipulation of the robotic machine-tending system, equipping it to handle the various scenarios encountered by machines in case of emergency.

Chapter 7 : Conclusions, Discussions & Future Work

7.1 Conclusions

The manufacturing sector is a fundamental pillar of worldwide economies, contributing markedly to global economic growth. However, the manufacturing industry is persistently confronted with issues impeding its development and expansion, such as manpower shortages, safety concerns, high initial investment for installation, and long return on investment. Within this context, machine tending has become a crucial component of the manufacturing process and potentially serves as a viable solution to the aforementioned predicaments. Over the past five years, the implementation of automated machine-tending systems has widely extended from simulation or laboratory environments to practical applications in manufacturing workshops as robotics and artificial intelligence develop rapidly. To fully benefit from the potential of machine-tending applications, it is necessary to comprehend and tackle the challenges associated with machine tending. Therefore, a preliminary investigation was conducted, and a systematic literature review based on the Protocol of Preferred Reporting Items for Systematic Review and Meta-Analyses was completed to analyze scientific literature related to machine tending in the last five years. The findings of this review elucidate the prevailing trends in emerging technologies that are advancing the autonomy of machine tending. A noteworthy observation is that most research and applications are currently in their prototypical stage. Additionally, it deliberates some challenges and potential future perspectives for achieving fully autonomous machine tending, and so far, an intelligent framework to achieve fully autonomous machine tending is still lacking.

An intelligent framework for autonomous machine-tending applications is proposed to address the research gaps and overcome the challenges in realizing the autonomous machine-tending process. This framework is composed of four main steps.

An autonomous charging strategy is proposed to charge machine-tending systems to achieve continuous and smooth working human intervention in an uncontrolled manufacturing environment. This autonomous charging strategy proposed an improved intelligent robot path planning method based on RRT and quantic B-spline curve techniques to enable machine-tending systems to move between workstations and charge stations without collision. This method can generate a trimmed and smooth path for mobile platforms moving continuously following the kinematic and dynamic rules. Then, a YOLOv7-based method is developed to recognize and localize the charger in real-time. Finally, a vision and lidar-based docking strategy is proposed to dock the machine-tending system to the target charger. The developed YOLOv7 model was trained and tested on the created datasets in this work. According to the results, it can achieve 99.4% mean average precision accuracy of the charger detection and 100% accuracy in docking the mobile platform to the target charger in real experiments.

Secondly, a deep learning-based intelligent machine detection method is developed to detect a particular CNC machine in real-time in a complicated manufacturing environment. This method is named SiameseRPN, and it consists of two subnetworks: the region proposal network and the Siamese neural networks. It is noted that it performs better to distinguish one specific CNC machine from a group of similar machines compared with other object detection methods according to the validation results. Compared with other milestone object detection methods such as Faster-RCNN, SSD, and YOLO, the developed method can improve performance in both efficiency and accuracy. It achieves 11% higher accuracy than Faster-RCNN.

Once the target CNC machine is detected, an OCR-based method is applied to inspect the working status of the machine, given that scene text recognition has a promising capability in detecting and understanding the commands and instructions from the machine's display. In this step, the command recognition method is developed by combining the text detection and recognition branches. Its architecture includes three parts: the adjusted text region proposal network, the recurrent neural network, and the connectionist temporal classification. In addition, several pre-processing techniques, including FFT and GANs, are used to improve the quality of captured real-time images by removing noises and reflections to improve text recognition accuracy. Furthermore, the recognized results are fed into the dictionary-guided procedure to modify and correct the output results. This proposed method can achieve an accuracy of 100% in recognizing machine instructions.

In the last step, the recognized commands from the previous procedure are used as a reference for detecting and localizing the target buttons mentioned in the commands to achieve the basic operation of machine-tending systems next. Here, a benchmark dataset for five different machine control keys from the Haas CNC machine is created. A benchmark YOLOv7-based button detection and localization method is proposed, trained, and tested on the created dataset for future utilization in machine-tending tasks through transfer learning. According to the analyzed results, this proposed method can achieve an overall accuracy of 98.8%.

7.2 Research Contributions

This research contributes to machine tending by proposing an intelligent framework to automate the machine-tending process by integrating novel developed methods with emerging technologies. The primary contributions of this thesis are summarized as follows:

- Conduct a systematic literature review of current machine-tending systems applications. It can contribute to the evolution of machine-tending systems and applications by investigating the impacts of emerging trends of advanced technologies and providing a comprehensive understanding of the current research status, bottlenecks, and future directions of machine tending due to the lack of a comprehensive literature review. It is observed that the majority of the research and applications are still in the conceptual or prototypical stages. Moreover, an intelligent framework to achieve fully autonomous machine tending is still missing and lacks more research work and validation.
- Developed an autonomous charging strategy for machine-tending systems, which enables the system to move between different machines and charging stations, detect the charger, and dock to the charger. This method can achieve a good accuracy (99.4% mAP_0.5 and 86.5% mAP_0.5:0.95) to recognize the target charging with a small training dataset and low cost for the system based on transfer learning techniques.
- Developed a deep learning-based method for automatic machine detection to identify the target CNC machine from many similar machines in the complex manufacturing environment. The author created the specific dataset, and 200 images of the Haas CNC machine in total with a solution of 960 × 1080 were shot from different angles and positions in our lab. The architecture consists of two subnetworks: the region proposal network and the Siamese neural network, and this model is trained based on the created dataset. This method can achieve the 0.127 training loss and 0.112 validation loss, respectively.
- Developed a deep learning-based command recognition method for work status inspection with three subnetworks: the convolutional network for feature extraction, the recurrent neural network to predict the label sequence, and the connectionist temporal classification

to predict the output text. A proper text dataset was created by collecting 200 images captured from the CNC machine display and labelled manually using the LabelImg Software. This unified model is trained based on the created dataset, achieving 0.02 training loss and 100% accuracy. For the testing, it can also achieve an overall 99% accuracy in detection and 100% accuracy in recognition, respectively. According to the experiment results, this method improves the command detection and recognition compared with benchmark text detection methods such as TextBoxes++ and EAST, and benchmark text recognition methods such as CRNN.

- Developed a real-time button detection and localization method that can identify the target button after obtaining the machine instructions and provide the position information to the machine-tending system to execute the related commands autonomously. This step creates a benchmark dataset of five different categories of control buttons from the Haas CNC machine due to the lack of a proper public dataset. In the dataset, 1000 images with a resolution of 1080 × 720 are collected and annotated manually, including the five most commonly used buttons: emergency stop button, power on button, power off button, power up button, and cycle start button. In addition, this method uses the YOLOv7-tiny as the backbone architecture, and the spatial three-dimensional coordinates of the central point of the detected target button can be calculated based on the image coordinates of the bounding box and depth information obtained from the depth camera. This method shows an overall 98.6 precision in detecting and distinguishing the different target buttons. In addition, this method is estimated in case studies by moving the depth camera to five different positions. It is observed that the end-effector of the machine-tending robot arm can reach the target button successfully although there are small errors between the predicted position and ground truth.

7.3 Limitations and Future Work

Even though the proposed intelligent framework for autonomous machine tending achieved promising results, the research presented is confronted by particular limitations and can be addressed in future work. The works as expanding solutions from the author's perspective to improve and enhance the developed intelligent framework related to each process are presented in Figure 7.1.

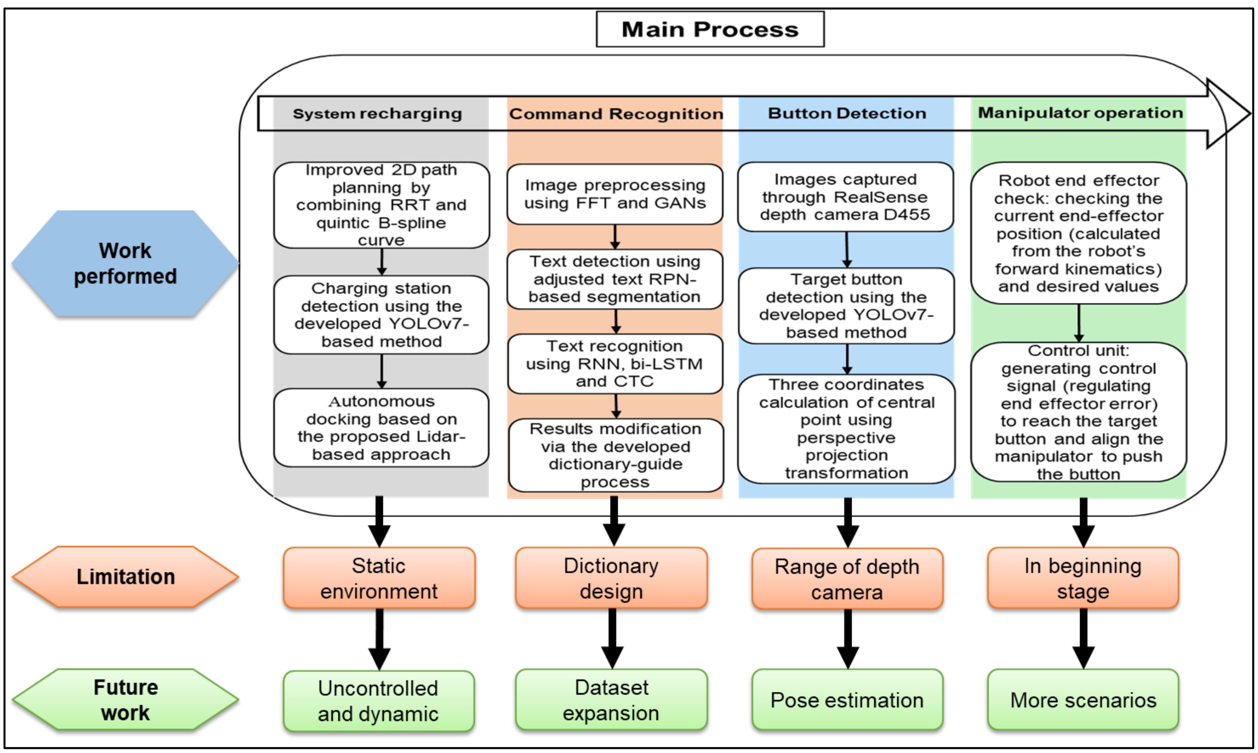

Figure 7.1: The diagram of future work.

The limitations and future work are explained in details as below:

- For system recharging process, the proposed methods present a good performance in terms of recognizing the target charging station and docking the mobile robot system to the wireless charger in a manufacturing environment. However, the improved path-planning method is limited by the static environment. Considering most of practical machine-tending tasks are performed in an uncontrolled and dynamic environment, a dynamic path-planning method is capable of moving mobile robot systems between different workstations and charging stations while avoiding unknown moving obstacles in real-time needs to be developed. In addition, a strategy to determine when the mobile robot system needs to be recharged is also necessary.
- Implementing the autonomous machine detection model to a practical manufacturing application requires more fine-tuning to achieve reliable and accurate results. The tunning process is carried out by training the detection model on real-life data, and the performance of the model depends on large amounts of high-quality data. Therefore, less data might lead to large biases and failure to achieve the desired results. To overcome these challenges, a customized dataset with more annotated real-life data needs to be created in the future development of the machine detection method.
- In the command recognition process, one limitation of deploying the proposed method is the design of the guided dictionary. Different brands of CNC machines have their own expressions of instructions, which might cause the low accuracy to adopt the proposed command recognition method directly to other machines. Consequently, a reference dictionary needs to be designed considering instructions from various brands of CNC machines to improve the generalization ability of the developed model.

- For the button detection and localization process, the implementation of the developed method is limited by the range of the Intel RealSense Depth camera used in the study. In addition, although image-to-spatial mapping based on the position of the bounding box and depth information can predict the 3D coordinates of buttons accurately, it can also lead to failure for the end-effector to reach the target button because the pose of the button is also a significant factor in the real-world scenarios. Therefore, a proper pose estimation method for the target button needs to be developed in future work to over this challenge.
- For instruction execution prosess, some machine-tending systems must make decisions to interact with complex environments and deal with various situations. Therefore, the ability of task planning is a demand. Further development of decision-making strategies for machine-tending systems using ML- and DL-techniques such as Q learning is required in the future.
- Currently, the proposed intelligent framework is the first real attempt to develop fully autonomous machine tending; therefore, many complicated scenarios will need to be considered in the future to make this a system to replace humans in lights-out manufacturing. It is validated and estimated based on ideal problems and scenarios with a good amount of labelled data. However, the real-world scenarios are different, such as imperfect lighting conditions and object occlusion, which leads to the difficulty of implementing the framework into practical machine-tending applications based on real-life data. Task-oriented methods can be a solution. Therefore, more examination and experiments are required to explore the different ML or DL structures and parameters to achieve satisfactory performance for each machine-tending task, which can be further applied to the same or similar tasks.

Reference

[1] R. S. Chains, I. N. The, N. E. W. Era, and O. F. Manufacturing, *The 2022 World Manufacturing Report*. 2022.

[2] S. Helper, T. Krueger, and H. Wial, “Why Does Manufacturing Matter? Which Manufacturing Matters? A Policy Framework,” *SSRN Electronic Journal*, no. February, pp. 1–53, 2021, doi: 10.2139/ssrn.3798089.

[3] ILO, “ILO Monitor on the world of work . Ninth edition,” no. May, pp. 1–25, 2022.

[4] B. Rooks, “Machine tending in the modern age,” *Industrial Robot: An International Journal*, vol. 30, no. 4, pp. 313–318, Aug. 2003, doi: 10.1108/01439910310479568.

[5] G. Erdoğan, “Land selection criteria for lights out factory districts during the industry 4.0 process,” *Journal of Urban Management*, vol. 8, no. 3, pp. 377–385, Dec. 2019, doi: 10.1016/j.jum.2019.01.001.

[6] M. M. Erdoğdu, “Lights-Out Manufacturing and Foreign Direct Investment Decline: Human Resource-Based Avenues for Technological Diffusion in Developing Countries,” in *International Political Economy Series*, Palgrave Macmillan, 2023, pp. 97–119. doi: 10.1007/978-3-031-25832-9_5.

[7] S. K. Gupta, “Towards next generation pandemic proof factories,” *Smart Sustain Manuf Syst*, vol. 4, no. 3, pp. 260–263, 2020, doi: 10.1520/SSMS20200058.

[8] J. Lee, H. Davari, J. Singh, and V. Pandhare, “Industrial Artificial Intelligence for industry 4.0-based manufacturing systems,” *Manuf Lett*, vol. 18, pp. 20–23, 2018, doi: 10.1016/j.mfglet.2018.09.002.

[9] D. Li and Y. Huang, "A general robot application platform for machine tending," in *2008 IEEE Conference on Robotics, Automation and Mechatronics*, IEEE, Sep. 2008, pp. 1–5. doi: 10.1109/RAMECH.2008.4690867.

[10] I. Iglesias, M. A. Sebastián, and J. E. Ares, "Overview of the State of Robotic Machining: Current Situation and Future Potential," *Procedia Eng*, vol. 132, pp. 911–917, 2015, doi: 10.1016/j.proeng.2015.12.577.

[11] Y. L. Kim, K. H. Ahn, and J. B. Song, "Direct teaching algorithm based on task assistance for machine tending," *2016 13th International Conference on Ubiquitous Robots and Ambient Intelligence, URAI 2016*, pp. 861–866, 2016, doi: 10.1109/URAI.2016.7733996.

[12] A. Klimchik, A. Ambiehl, S. Garnier, B. Furet, and A. Pashkevich, "Experimental study of robotic-based machining," *IFAC-PapersOnLine*, vol. 49, no. 12, pp. 174–179, 2016, doi: 10.1016/j.ifacol.2016.07.591.

[13] V. Annem, P. Rajendran, S. Thakar, and S. K. Gupta, "Towards Remote Teleoperation of a Semi-Autonomous Mobile Manipulator System in Machine Tending Tasks," in *Volume 1: Additive Manufacturing; Manufacturing Equipment and Systems; Bio and Sustainable Manufacturing*, American Society of Mechanical Engineers, Jun. 2019, pp. 1–12. doi: 10.1115/MSEC2019-3027.

[14] J. Zhang and X. Fang, "Challenges and key technologies in robotic cell layout design and optimization," *Proc Inst Mech Eng C J Mech Eng Sci*, vol. 231, no. 15, pp. 2912–2924, 2017, doi: 10.1177/0954406216642473.

[15] F. Jia, J. Tzintzun, and R. Ahmad, "An Improved Robot Path Planning Algorithm for a Novel Self-adapting Intelligent Machine Tending Robotic System," in *Mechanisms and Machine Science*, vol. 86, Springer International Publishing, 2020, pp. 53–64. doi: 10.1007/978-3-030-45402-9_7.

[16] S. Al-Hussaini *et al.*, "Human-Supervised Semi-Autonomous Mobile Manipulators for Safely and Efficiently Executing Machine Tending Tasks," 2020, [Online]. Available: http://arxiv.org/abs/2010.04899

[17] M. Molina and P. Santamaria, "Behavior coordination for self-adaptive robots using constraint-based configuration," Mar. 2021, [Online]. Available: http://arxiv.org/abs/2103.13128

[18] H. Chen *et al.*, "From automation system to autonomous system: An architecture perspective," *Journal of Marine Science and Engineering*, vol. 9, no. 6. MDPI AG, Jun. 01, 2021. doi: 10.3390/jmse9060645.

[19] P. Avgeriou, S. Mahdavi-Hezavehi, and D. Weyns, "A Classification Framework of Uncertainty in Architecture-Based Self-Adaptive Systems With Multiple Quality Requirements," in *Managing Trade-offs in Adaptable Software Architectures*, Elsevier, 2017, pp. 45–77. doi: 10.1016/B978-0-12-802855-1.00003-4.

[20] R. Y. Zhong, X. Xu, E. Klotz, and S. T. Newman, "Intelligent Manufacturing in the Context of Industry 4.0: A Review," *Engineering*, vol. 3, no. 5, pp. 616–630, 2017, doi: 10.1016/J.ENG.2017.05.015.

[21] M. J. Page *et al.*, "The PRISMA 2020 statement: An updated guideline for reporting systematic reviews," *The BMJ*, vol. 372, 2021, doi: 10.1136/bmj.n71.

[22] W. Sebbata, M. A. Kenk, and J. F. Brethé, "An adaptive robotic grasping with a 2-finger gripper based on deep learning network," *IEEE International Conference on Emerging Technologies and Factory Automation, ETFA*, vol. 2020-Septe, pp. 620–627, 2020, doi: 10.1109/ETFA46521.2020.9212163.

[23] X. Yang, Z. Zhou, J. H. Sørensen, C. B. Christensen, M. Ünalan, and X. Zhang, "Automation of SME production with a Cobot system powered by learning-based vision," *Robot Comput Integr Manuf*, vol. 83, no. July 2022, p. 102564, 2023, doi: 10.1016/j.rcim.2023.102564.

[24] M. El Helou *et al.*, "A Modular Smart Vision System for Industrial Inspection and Control of Conformity," *Smart Sustain Manuf Syst*, vol. 6, no. 1, pp. 177–189, 2022, doi: 10.1520/SSMS20220003.

[25] S. Comari *et al.*, "Mobile cobots for autonomous raw-material feeding of automatic packaging machines," *J Manuf Syst*, vol. 64, no. July, pp. 211–224, 2022, doi: 10.1016/j.jmsy.2022.06.007.

[26] A. Sardelis *et al.*, "2-Stage vision system for robotic handling of flexible objects," *Procedia CIRP*, vol. 97, no. March, pp. 491–496, 2020, doi: 10.1016/j.procir.2020.07.008.

[27] X. Wu, P. Li, J. Zhou, and Y. Liu, "A cascaded CNN-based method for monocular vision robotic grasping," *Industrial Robot*, vol. 49, no. 4, pp. 645–657, 2022, doi: 10.1108/IR-10-2021-0236.

[28] D. K. Moru and D. Borro, “A machine vision algorithm for quality control inspection of gears,” *International Journal of Advanced Manufacturing Technology*, vol. 106, no. 1–2, pp. 105–123, 2020, doi: 10.1007/s00170-019-04426-2.

[29] G. Zhou *et al.*, “A framework of industrial operations for hybrid robots,” *2021 26th International Conference on Automation and Computing: System Intelligence through Automation and Computing, ICAC 2021*, pp. 1–6, 2021, doi: 10.23919/ICAC50006.2021.9594267.

[30] F. Jia, A. Jebelli, Y. Ma, and R. Ahmad, “An Intelligent Manufacturing Approach Based on a Novel Deep Learning Method for Automatic Machine and Working Status Recognition,” *Applied Sciences*, vol. 12, no. 11, p. 5697, Jun. 2022, doi: 10.3390/app12115697.

[31] J. Zheng, L. Chen, Y. Li, Y. A. Khan, H. Lyu, and X. Wu, “An Intelligent Robot Sorting System By Deep Learning On RGB-D Image,” *2023 22nd International Symposium INFOTEH-JAHORINA, INFOTEH 2023*, no. March, pp. 1–8, 2023, doi: 10.1109/INFOTEH57020.2023.10094093.

[32] K. Aliev and D. Antonelli, *Analysis of cooperative industrial task execution by mobile and manipulator robots*, vol. 1. Springer International Publishing, 2019. doi: 10.1007/978-3-030-18715-6_21.

[33] R. Arrais *et al.*, “Application of the Open Scalable Production System to Machine Tending of Additive Manufacturing Operations by a Mobile Manipulator,” *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial*

Intelligence and Lecture Notes in Bioinformatics), vol. 11805 LNAI. pp. 345–356, 2019. doi: 10.1007/978-3-030-30244-3_29.

[34] R. Yang, T. P. Nguyen, S. H. Park, and J. Yoon, "Automated picking-sorting system for assembling components in an IKEA chair based on the robotic vision system," *Int J Comput Integr Manuf*, vol. 35, no. 6, pp. 583–597, 2022, doi: 10.1080/0951192X.2021.1992658.

[35] Y. Ma, W. Zhu, and Y. Zhou, "Automatic grasping control of mobile robot based on monocular vision," *International Journal of Advanced Manufacturing Technology*, vol. 121, no. 3–4, pp. 1785–1798, 2022, doi: 10.1007/s00170-022-09438-z.

[36] X. Zhang, K. Eltouny, X. Liang, and S. Behdad, "Automatic Screw Detection and Tool Recommendation System for Robotic Disassembly," *J Manuf Sci Eng*, vol. 145, no. 3, pp. 1–8, 2023, doi: 10.1115/1.4056074.

[37] M. Hirz, B. Walzel, and H. Brunner, "Autonomous Charging of Electric Vehicles in Industrial Environment," *Tehnicki Glasnik*, vol. 15, no. 2, pp. 220–225, 2021, doi: 10.31803/tg-20210428191147.

[38] A. A. Transeth, A. Stepanov, A. S. Linnerud, K. Ening, and T. Gjerstad, "Competitive high variance, low volume manufacturing with robot manipulators," *2020 3rd International Symposium on Small-Scale Intelligent Manufacturing Systems, SIMS 2020*, vol. m, 2020, doi: 10.1109/SIMS49386.2020.9121464.

[39] D. Guerra-Zubiaga *et al.*, "Computer Vision and Machine Learning To Create an Advanced Pick-and-Place Robotic Operation Using Industry 4.0 Trends," *ASME*

International Mechanical Engineering Congress and Exposition, Proceedings (IMECE), vol. 2-A, pp. 1–10, 2022, doi: 10.1115/IMECE2022-89743.

[40] M. A. Riedlinger, M. Ghafarian Tamizi, J. Tikekar, and M. Redeker, “Concept for a Distributed Picking Application Utilizing Robotics and Digital Twins,” *IEEE International Conference on Emerging Technologies and Factory Automation, ETFA*, vol. 2022-Septe, pp. 1–4, 2022, doi: 10.1109/ETFA52439.2022.9921659.

[41] C. Holst, T. B. Yavuz, P. Gupta, P. Ganser, and T. Bergs, “Deep learning and rule-based image processing pipeline for automated metal cutting tool wear detection and measurement,” *IFAC-PapersOnLine*, vol. 55, no. 2, pp. 534–539, 2022, doi: 10.1016/j.ifacol.2022.04.249.

[42] M. Necula, E. Câmpean, and L. Morar, “DEFINING THE CHARACTERISTICS NEEDED FOR THE COBOTS DESIGN USED FOR THE SUPPLY OF CNC MACHINES,” vol. 65, pp. 171–178, 2022.

[43] M. Zemzemoglu and M. Unel, “Design and Implementation of a Vision Based In-Situ Defect Detection System of Automated Fiber Placement Process,” *IEEE International Conference on Industrial Informatics (INDIN)*, vol. 2022-July, pp. 393–398, 2022, doi: 10.1109/INDIN51773.2022.9976182.

[44] Y. Mao, C. Chen, and H. Jiang, “Design and Implementation of Sorting System based on Machine Vision,” *2022 7th International Conference on Big Data Analytics, ICBDA 2022*, pp. 259–264, 2022, doi: 10.1109/ICBDA55095.2022.9760324.

[45] H. Ma, L. Li, F. Shao, and X. Liu, *Design of a Comprehensive Experimental Platform for Intelligent Robots Based on Machine Vision*, vol. 880 LNEE. Springer Singapore, 2022. doi: 10.1007/978-981-19-0572-8_49.

[46] Z. Mei, D. Li, T. Chen, and C. Yu, "Design of vision system of transceiver robot based on Halcon," *2022 IEEE International Conference on Artificial Intelligence and Computer Applications, ICAICA 2022*, pp. 204–207, 2022, doi: 10.1109/ICAICA54878.2022.9844593.

[47] F. D'Souza, J. Costa, and J. N. Pires, "Development of a solution for adding a collaborative robot to an industrial AGV," *Industrial Robot*, vol. 47, no. 5, pp. 723–735, 2020, doi: 10.1108/IR-01-2020-0004.

[48] H. A. Phan *et al.*, "Development of a Vision System to Enhance the Reliability of the Pick-and-Place Robot for Autonomous Testing of Camera Module used in Smartphones," *7th International Conference on Engineering and Emerging Technologies, ICEET 2021*, no. October, pp. 1–6, 2021, doi: 10.1109/ICEET53442.2021.9659578.

[49] J. Park, M. B. G. Jun, and H. Yun, "Development of robotic bin picking platform with cluttered objects using human guidance and convolutional neural network (CNN)," *J Manuf Syst*, vol. 63, no. February, pp. 539–549, 2022, doi: 10.1016/j.jmsy.2022.05.011.

[50] T. Jadhav, M. Jadhav, A. Chitre, and A. Patil, "Development of Vision Based Test Jig and Mechanism for Automatic Inspection and Sorting of Industrial Objects," *2022 6th International Conference on Computing, Communication, Control and*

Automation, ICCUBEA 2022, pp. 1–5, 2022, doi: 10.1109/ICCUBEA54992.2022.10010756.

[51] M. Wu, F. Guo, J. Wu, Y. Xiao, M. Jin, and Q. Zhang, "Emergency Stop System of Computer Vision Workstation Based on GMM-HMM and LSTM," *2023 9th International Conference on Automation, Robotics and Applications, ICARA 2023*, pp. 150–154, 2023, doi: 10.1109/ICARA56516.2023.10125926.

[52] F. A. Jafar, N. A. Zakaria, A. Z. M. Noor, and K. Yokota, "Environmental Visual Features Based Place Recognition in Manufacturing Environment," in *Intelligent Manufacturing and Mechatronics. SympoSIMM 2021*, vol. 1, M. N. Ali Mokhtar, Z. Jamaludin, M. S. Abdul Aziz, M. N. Maslan, and J. A. Razak, Eds., in Lecture Notes in Mechanical Engineering, vol. 1. , Singapore: Springer Nature Singapore, 2022, pp. 47–59. doi: 10.1007/978-981-16-8954-3_6.

[53] A. A. Pradhan, W. C. Martin, J. D. Ruiz, and P. E. Deierling, "Framework for automated robotic arm manipulation in variable industrial environments," *ASME International Mechanical Engineering Congress and Exposition, Proceedings (IMECE)*, vol. 2B-2021, pp. 1–9, 2021, doi: 10.1115/IMECE2021-71479.

[54] K. Wu and J. Cao, "High Robust Vision System of Robot in Intelligent Manufacturing," *Proceedings - 2020 Chinese Automation Congress, CAC 2020*, no. 2019, pp. 4210–4215, 2020, doi: 10.1109/CAC51589.2020.9327823.

[55] R. G. Lins, P. R. M. de Araujo, and M. Corazzim, "In-process machine vision monitoring of tool wear for Cyber-Physical Production Systems," *Robot Comput*

Integr Manuf, vol. 61, no. March 2019, p. 101859, 2020, doi: 10.1016/j.rcim.2019.101859.

[56] Z. Zhou, L. Li, A. Fürsterling, H. J. Durocher, J. Mouridsen, and X. Zhang, “Learning-based object detection and localization for a mobile robot manipulator in SME production,” *Robot Comput Integr Manuf*, vol. 73, no. July 2021, p. 102229, 2022, doi: 10.1016/j.rcim.2021.102229.

[57] Q. Chen, B. Heydari, and M. Moghaddam, “Leveraging task modularity in reinforcement learning for adaptable industry 4.0 automation,” *Journal of Mechanical Design, Transactions of the ASME*, vol. 143, no. 7, 2021, doi: 10.1115/1.4049531.

[58] D. Horvath, G. Erdos, Z. Istenes, T. Horvath, and S. Foldi, “Object Detection Using Sim2Real Domain Randomization for Robotic Applications,” *IEEE Transactions on Robotics*, vol. 39, no. 2, pp. 1225–1243, 2023, doi: 10.1109/TRO.2022.3207619.

[59] J. Cho, S. Kang, and K. Kim, “Real-time precise object segmentation using a pixel-wise coarse-fine method with deep learning for automated manufacturing,” *J Manuf Syst*, vol. 62, no. November 2021, pp. 114–123, 2022, doi: 10.1016/j.jmsy.2021.11.004.

[60] F. Jia, Y. Ma, and R. Ahmad, “Vision-Based Associative Robotic Recognition of Working Status in Autonomous Manufacturing Environment,” *Procedia CIRP*, vol. 104, no. March, pp. 1535–1540, 2021, doi: 10.1016/j.procir.2021.11.259.

[61] H. Mamledesai, M. A. Soriano, and R. Ahmad, “A Qualitative Tool Condition Monitoring Framework Using Convolution Neural Network and Transfer Learning,” *Applied Sciences*, vol. 10, no. 20, p. 7298, Oct. 2020, doi: 10.3390/app10207298.

[62] V. Alcácer and V. Cruz-Machado, “Scanning the Industry 4.0: A Literature Review on Technologies for Manufacturing Systems,” *Engineering Science and Technology, an International Journal*, vol. 22, no. 3, pp. 899–919, 2019, doi: 10.1016/j.jestch.2019.01.006.

[63] A. Bonci, P. D. C. Cheng, M. Indri, G. Nabissi, and F. Sibona, “Human-robot perception in industrial environments: A survey,” *Sensors*, vol. 21, no. 5, pp. 1–29, 2021, doi: 10.3390/s21051571.

[64] Y. K. Park, J. Park, S. Kim, D. Lee, S. Lee, and J. Um, “Development of Digital twin for Plug-and-Produce of a Machine tending system through ISO 21919 interface,” *Int J Comput Integr Manuf*, vol. 00, no. 00, pp. 1–16, 2023, doi: 10.1080/0951192X.2023.2189306.

[65] E. Matheson, R. Minto, E. G. G. Zampieri, M. Faccio, and G. Rosati, “Human-robot collaboration in manufacturing applications: A review,” *Robotics*, vol. 8, no. 4, pp. 1–25, 2019, doi: 10.3390/robotics8040100.

[66] Y. Domae, “Recent trends in the research of industrial robots and future outlook,” *Journal of Robotics and Mechatronics*, vol. 31, no. 1, pp. 57–62, 2019, doi: 10.20965/jrm.2019.p0057.

[67] A. Zeng *et al.*, “Robotic pick-and-place of novel objects in clutter with multi-affordance grasping and cross-domain image matching,” *International Journal of*

Robotics Research, vol. 41, no. 7, pp. 690–705, 2022, doi: 10.1177/0278364919868017.

[68] D. Rodriguez-Guerra, G. Sorrosal, I. Cabanes, and C. Calleja, “Human-Robot Interaction Review: Challenges and Solutions for Modern Industrial Environments,” *IEEE Access*, vol. 9, pp. 108557–108578, 2021, doi: 10.1109/ACCESS.2021.3099287.

[69] L. Gualtieri, E. Rauch, and R. Vidoni, “Emerging research fields in safety and ergonomics in industrial collaborative robotics: A systematic literature review,” *Robot Comput Integr Manuf*, vol. 67, no. April 2020, 2021, doi: 10.1016/j.rcim.2020.101998.

[70] K. A. Demir, G. Döven, and B. Sezen, “Industry 5.0 and Human-Robot Co-working,” *Procedia Comput Sci*, vol. 158, pp. 688–695, 2019, doi: 10.1016/j.procs.2019.09.104.

[71] E. Coronado, T. Kiyokawa, G. A. G. Ricardez, I. G. Ramirez-Alpizar, G. Venture, and N. Yamanobe, “Evaluating quality in human-robot interaction: A systematic search and classification of performance and human-centered factors, measures and metrics towards an industry 5.0,” *J Manuf Syst*, vol. 63, no. January, pp. 392–410, 2022, doi: 10.1016/j.jmsy.2022.04.007.

[72] P. K. R. Maddikunta *et al.*, “Industry 5.0: A survey on enabling technologies and potential applications,” *J Ind Inf Integr*, vol. 26, no. July 2021, p. 100257, 2022, doi: 10.1016/j.jii.2021.100257.

[73] S. Boøgh, M. Hvilshoøj, M. Kristiansen, and O. Madsen, “Identifying and evaluating suitable tasks for autonomous industrial mobile manipulators (AIMM),”

International Journal of Advanced Manufacturing Technology, 2012, doi: 10.1007/s00170-011-3718-3.

[74] E. A. Oyekanlu *et al.*, "A review of recent advances in automated guided vehicle technologies: Integration challenges and research areas for 5G-based smart manufacturing applications," *IEEE Access*, vol. 8, pp. 202312–202353, 2020, doi: 10.1109/ACCESS.2020.3035729.

[75] F. Arvin, S. Watson, A. E. Turgut, J. Espinosa, T. Krajník, and B. Lennox, "Perpetual Robot Swarm: Long-Term Autonomy of Mobile Robots Using On-the-fly Inductive Charging," *Journal of Intelligent and Robotic Systems: Theory and Applications*, vol. 92, no. 3–4, pp. 395–412, 2018, doi: 10.1007/s10846-017-0673-8.

[76] M. Vajgl, P. Hurtik, and T. Nejezchleba, "Dist-YOLO: Fast Object Detection with Distance Estimation," *Applied Sciences*, vol. 12, no. 3, p. 1354, Jan. 2022, doi: 10.3390/app12031354.

[77] M. Javaid, A. Haleem, R. P. Singh, S. Rab, and R. Suman, "Exploring impact and features of machine vision for progressive industry 4.0 culture," *Sensors International*, vol. 3, no. November 2021, p. 100132, 2022, doi: 10.1016/j.sintl.2021.100132.

[78] R. L. Silva, M. Rudek, A. L. Szejka, and O. Canciglieri Junior, *Machine vision systems for industrial quality control inspections*, vol. 540. Springer International Publishing, 2018. doi: 10.1007/978-3-030-01614-2_58.

[79] L. Pérez, Í. Rodríguez, N. Rodríguez, R. Usamentiaga, and D. F. García, "Robot guidance using machine vision techniques in industrial environments: A comparative review," *Sensors (Switzerland)*, vol. 16, no. 3, 2016, doi: 10.3390/s16030335.

[80] G. J. Awcock and R. Thomas, *Applied Image Processing*. London: Macmillan Education UK, 1995. doi: 10.1007/978-1-349-13049-8.

[81] Z. Zou, K. Chen, Z. Shi, Y. Guo, and J. Ye, "Object Detection in 20 Years: A Survey," *Proceedings of the IEEE*, vol. 111, no. 3, pp. 257–276, 2023, doi: 10.1109/JPROC.2023.3238524.

[82] T. T. H. Nguyen, A. Jatowt, M. Coustaty, and A. Doucet, "Survey of Post-OCR Processing Approaches," *ACM Comput Surv*, vol. 54, no. 6, 2021, doi: 10.1145/3453476.

[83] Q. J. Zhao, P. Cao, and D. W. Tu, "Toward intelligent manufacturing: label characters marking and recognition method for steel products with machine vision," *Adv Manuf*, vol. 2, no. 1, pp. 3–12, 2014, doi: 10.1007/s40436-014-0057-2.

[84] M. Shahin, F. F. Chen, and A. Hosseinzadeh, "Machine-based identification system via optical character recognition," *Flex Serv Manuf J*, no. 0123456789, 2023, doi: 10.1007/s10696-023-09497-8.

[85] J. Koponen, K. Haataja, and P. Toivanen, "A novel deep learning method for recognizing texts printed with multiple different printing methods," *F1000Res*, vol. 12, p. 427, 2023, doi: 10.12688/f1000research.131775.1.

[86] T. Caldeira, P. M. Ciarelli, and G. A. Neto, "Industrial Optical Character Recognition System in Printing Quality Control of Hot-Rolled Coils Identification," *Journal of*

Control, Automation and Electrical Systems, vol. 31, no. 1, pp. 108–118, 2020, doi: 10.1007/s40313-019-00551-1.

[87] G. Zeba, M. Dabić, M. Čičak, T. Daim, and H. Yalcin, “Technology mining: Artificial intelligence in manufacturing,” *Technol Forecast Soc Change*, vol. 171, no. February, 2021, doi: 10.1016/j.techfore.2021.120971.

[88] T. Wuest, D. Weimer, C. Irgens, and K. D. Thoben, “Machine learning in manufacturing: Advantages, challenges, and applications,” *Prod Manuf Res*, vol. 4, no. 1, pp. 23–45, 2016, doi: 10.1080/21693277.2016.1192517.

[89] M. Melhem, B. Ananou, M. Ouladsine, and J. Pinaton, “Regression Methods for Predicting the Product’s Quality in the Semiconductor Manufacturing Process,” in *IFAC-PapersOnLine*, Elsevier B.V., 2016, pp. 83–88. doi: 10.1016/j.ifacol.2016.07.554.

[90] M. Papananias, T. E. McLeay, M. Mahfouf, and V. Kadirkamanathan, “A Bayesian framework to estimate part quality and associated uncertainties in multistage manufacturing,” *Comput Ind*, vol. 105, pp. 35–47, 2019, doi: 10.1016/j.compind.2018.10.008.

[91] C. Jiang, J. Wan, and H. Abbas, “An Edge Computing Node Deployment Method Based on Improved k-Means Clustering Algorithm for Smart Manufacturing,” *IEEE Syst J*, vol. 15, no. 2, pp. 2230–2240, 2021, doi: 10.1109/JSYST.2020.2986649.

[92] Y. Lecun, Y. Bengio, and G. Hinton, “Deep learning,” *Nature*, vol. 521, no. 7553, pp. 436–444, 2015, doi: 10.1038/nature14539.

[93] R. Xu and W. Yan, "A Comparison of GANs-Based Approaches for Combustor System Fault Detection," *Proceedings of the International Joint Conference on Neural Networks*, 2020, doi: 10.1109/IJCNN48605.2020.9207487.

[94] R. Singh, R. Garg, N. S. Patel, and M. W. Braun, "Generative Adversarial Networks for Synthetic Defect Generation in Assembly and Test Manufacturing," *ASMC (Advanced Semiconductor Manufacturing Conference) Proceedings*, vol. 2020-Augus, pp. 1–5, 2020, doi: 10.1109/ASMC49169.2020.9185242.

[95] L. Zhou, L. Zhang, and N. Konz, "Computer Vision Techniques in Manufacturing," *IEEE Trans Syst Man Cybern Syst*, vol. 53, no. 1, pp. 105–117, 2023, doi: 10.1109/TSMC.2022.3166397.

[96] S. H. Huang *et al.*, "Learning Gentle Object Manipulation with Curiosity-Driven Deep Reinforcement Learning," Mar. 2019, [Online]. Available: http://arxiv.org/abs/1903.08542

[97] J. Wang, Y. Ma, L. Zhang, R. X. Gao, and D. Wu, "Deep learning for smart manufacturing: Methods and applications," *J Manuf Syst*, vol. 48, pp. 144–156, 2018, doi: 10.1016/j.jmsy.2018.01.003.

[98] C. Wu, H. Du, Q. Wu, and S. Zhang, "Image text deblurring method based on generative adversarial network," *Electronics (Switzerland)*, vol. 9, no. 2, pp. 1–14, 2020, doi: 10.3390/electronics9020220.

[99] A. Amanlou, A. A. Suratgar, J. Tavoosi, A. Mohammadzadeh, and A. Mosavi, "Single-Image Reflection Removal Using Deep Learning: A Systematic Review," *IEEE Access*, vol. 10, pp. 29937–29953, 2022, doi: 10.1109/ACCESS.2022.3156273.

[100] D. Lange, “Cognitive robotics: Making robots sense, understand, and interact,” *Computer (Long Beach Calif)*, vol. 52, no. 12, pp. 39–44, 2019, doi: 10.1109/MC.2019.2942579.

[101] A. E. Salman and M. R. Roman, “Augmented reality-assisted gesture-based teleoperated system for robot motion planning,” *Industrial Robot: the international journal of robotics research and application*, vol. 50, no. 5, pp. 765–780, Aug. 2023, doi: 10.1108/IR-11-2022-0289.

[102] R. Jeong *et al.*, “Learning Dexterous Manipulation from Suboptimal Experts,” pp. 1–20, Oct. 2020, [Online]. Available: http://arxiv.org/abs/2010.08587

[103] S. Jang, “MURM: Utilization of Multi-Views for Goal-Conditioned Reinforcement Learning in Robotic Manipulation,” 2023.

[104] P. Bergmann, M. Fauser, D. Sattlegger, and C. Steger, “MVTEC ad-A comprehensive real-world dataset for unsupervised anomaly detection,” *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, vol. 2019-June, pp. 9584–9592, 2019, doi: 10.1109/CVPR.2019.00982.

[105] T. Chen, S. Kornblith, M. Norouzi, and G. Hinton, “A simple framework for contrastive learning of visual representations,” *37th International Conference on Machine Learning, ICML 2020*, vol. PartF16814, no. Figure 1, pp. 1575–1585, 2020.

[106] M. Hvilshøj, S. Bøgh, O. S. Nielsen, and O. Madsen, “Autonomous industrial mobile manipulation (AIMM): Past, present and future,” *Industrial Robot*, 2012, doi: 10.1108/01439911211201582.

[107] F. Adăscăliţei, I. Doroftei, and G. Asachi, “Practical Applications for Mobile Robots based on Mecanum Wheels-a Systematic Survey,” 2011.

[108] C. N. To *et al.*, “Autodriver Autonomous Vehicles Control Strategy,” in *Procedia Computer Science*, Elsevier B.V., 2018, pp. 870–877. doi: 10.1016/j.procS.2018.08.021.

[109] G. Antonelli, S. Chiaverini, and G. Fusco, “A fuzzy-logic-based approach for mobile robot path tracking,” *IEEE Transactions on Fuzzy Systems*, vol. 15, no. 2, pp. 211–221, Apr. 2007, doi: 10.1109/TFUZZ.2006.879998.

[110] J. Lee, Os. Kwon, L. Zhang, and S. E. Yoon, “A selective retraction-based RRT planner for various environments,” *IEEE Transactions on Robotics*, vol. 30, no. 4, pp. 1002–1011, 2014, doi: 10.1109/TRO.2014.2309836.

[111] B. R. Donald, H. H. Nagel, D. G. Bobrow, and B. R. Donald, “A Search Algorithm for Motion Planning with Six Degrees of Freedom* Recommended by,” *Artif Intell*, vol. 31, no. 3, pp. 295–353, 1987.

[112] Q. Wang *et al.*, “Hysteresis identification of joint with harmonic drive transmission based on Monte Carlo method,” *Mechatronics*, vol. 99, p. 103166, May 2024, doi: 10.1016/j.mechatronics.2024.103166.

[113] J.-C. Latombe, “Robot Motion Planning: A Distributed Representation Approach,” *International Journal of Robotics Research (IJRR)* , vol. 10, no. 6, pp. 628–649, 1991.

[114] S. Carpin and G. Pillonetto, “Motion planning using adaptive random walks,” *IEEE Transactions on Robotics*, vol. 21, no. 1, pp. 129–136, Feb. 2005, doi: 10.1109/TRO.2004.833790.

[115] L. Kavraki and J.-C. Latombe, “Randomized Preprocessing of Configuration Space for Path Planning: Articulated Robots,” in *International Conference on Intelligent Robots and Systems (IROS)*, 1994, pp. 1764–1771.

[116] N. M. Amato and Y. Wu, “Randomized roadmap method for path and manipulation planning,” *Proc IEEE Int Conf Robot Autom*, vol. 1, no. April, pp. 113–120, 1996, doi: 10.1109/robot.1996.503582.

[117] L. E. Kavralu, P. Svestka, J.-C. Latombe, and M. H. Overmars, “Probabilistic Roadmaps for Path Planning in High-Dimensional Configuration Spaces,” in *IEEE Transactions on Robotics and Automation*, 1996, pp. 566–580.

[118] Steven M. LaValle, “Rapidly-Exploring Random Trees: A New Tool for Path Planning,” 1998.

[119] S. Karaman and E. Frazzoli, “Sampling-based algorithms for optimal motion planning,” *Int J Rob Res*, vol. 30, no. 7, pp. 846–894, 2011, doi: 10.1177/0278364911406761.

[120] M. Elbanhawi and M. Simic, “Sampling-based robot motion planning: A review,” *IEEE Access*, vol. 2, pp. 56–77, 2014, doi: 10.1109/ACCESS.2014.2302442.

[121] D. Hsu, J.-C. Latombe, and H. Kurniawati, “On the Probabilistic Foundations of Probabilistic Roadmap Planning,” *International Journal of Robotics Research (IJRR)*, vol. 25, no. 7, pp. 627–643, 2007.

[122] E. T. A. S. A. Corominas Murtra, “Autonomous Navigation for Urban Service Mobile Robots,” in *The 2010 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2010, pp. 4141–4146.

[123] X. P. Xu and F. G. Lang, "Quintic B-spline method for function reconstruction from integral values of successive subintervals," *Numer Algorithms*, vol. 66, no. 2, pp. 223–240, 2014, doi: 10.1007/s11075-013-9731-x.

[124] L. A. Piegl and W. Tiller, "Curve interpolation with arbitrary end derivatives," *Eng Comput*, vol. 16, no. 1, pp. 73–79, 2000, doi: 10.1007/s003660050038.

[125] J. Vargas, S. Alsweiss, O. Toker, R. Razdan, and J. Santos, "An Overview of Autonomous Vehicles Sensors and Their Vulnerability to Weather Conditions," *Sensors*, vol. 21, no. 16, p. 5397, Aug. 2021, doi: 10.3390/s21165397.

[126] E. Rashid, M. D. Ansari, V. K. Gunjan, and M. Ahmed, "Improvement in extended object tracking with the vision-based algorithm," *Studies in Computational Intelligence*, vol. 885, no. February, pp. 237–245, 2020, doi: 10.1007/978-3-030-38445-6_18.

[127] C. Yao, Y. Li, M. D. Ansari, M. A. Talab, and A. Verma, "Optimization of industrial process parameter control using improved genetic algorithm for industrial robot," *Paladyn*, vol. 13, no. 1, pp. 67–75, 2022, doi: 10.1515/pjbr-2022-0006.

[128] F. Guangrui and W. Geng, "Vision-based autonomous docking and re-charging system for mobile robot in warehouse environment," *2017 2nd International Conference on Robotics and Automation Engineering, ICRAE 2017*, vol. 2017-Decem, pp. 79–83, 2018, doi: 10.1109/ICRAE.2017.8291357.

[129] F. Rubio, F. Valero, and C. Llopis-Albert, "A review of mobile robots: Concepts, methods, theoretical framework, and applications," *Int J Adv Robot Syst*, vol. 16, no. 2, pp. 1–22, 2019, doi: 10.1177/1729881419839596.

[130] R. Abbasi, P. Martinez, and R. Ahmad, "The digitization of agricultural industry – a systematic literature review on agriculture 4.0," *Smart Agricultural Technology*, vol. 2, no. January, p. 100042, 2022, doi: 10.1016/j.atech.2022.100042.

[131] Y. Liu, "A Laser Intensity Based Autonomous Docking Approach for Mobile Robot Recharging in Unstructured Environments," *IEEE Access*, vol. 10, no. May, pp. 71165–71176, 2022, doi: 10.1109/ACCESS.2022.3188099.

[132] M. Doumbia, X. Cheng, and V. Havyarimana, "An Auto-Recharging System Design and Implementation Based on Infrared Signal for Autonomous Robots," in *2019 5th International Conference on Control, Automation and Robotics (ICCAR)*, IEEE, Apr. 2019, pp. 894–900. doi: 10.1109/ICCAR.2019.8813317.

[133] R. C. Luo, C. T. Liao, and S. C. Lin, "Multi-sensor fusion for reduced uncertainty in autonomous mobile robot docking and recharging," *2009 IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS 2009*, pp. 2203–2208, 2009, doi: 10.1109/IROS.2009.5354445.

[134] S. Khan, D. Wollherr, and M. Buss, "Modeling Laser Intensities for Simultaneous Localization and Mapping," *IEEE Robot Autom Lett*, vol. 1, no. 2, pp. 692–699, 2016, doi: 10.1109/LRA.2016.2516592.

[135] H. Tibebu, V. De-Silva, C. Artaud, R. Pina, and X. Shi, "Towards Interpretable Camera and LiDAR Data Fusion for Autonomous Ground Vehicles Localisation," *Sensors*, vol. 22, no. 20, p. 8021, Oct. 2022, doi: 10.3390/s22208021.

[136] M. V. S. Rao* and Dr. M. Shivakumar, "Sensor Guided Docking of Autonomous Mobile Robot for Battery Recharging," *International Journal of Recent Technology*

and Engineering (IJRTE), vol. 8, no. 4, pp. 3812–3816, 2019, doi: 10.35940/ijrte.d8176.118419.

[137] R. C. Luo, C. T. Liao, and K. C. Lin, "Vision-based docking for automatic security robot power recharging," in *IEEE Workshop on Advanced Robotics and its Social Impacts, 2005.*, IEEE, 2017, pp. 214–219. doi: 10.1109/ARSO.2005.1511654.

[138] A. Kriegler and W. Wöber, "Vision-based Docking of a Mobile Robot," *Joint Austrian Computer Vision and Robotics Workshop*, pp. 6–12, 2020, doi: 10.3217/978-3-85125-752-6-03.

[139] U. Kartoun *et al.*, "Vision-Based Autonomous Robot Self-Docking and Recharging," in *2006 World Automation Congress*, IEEE, Jul. 2006, pp. 1–8. doi: 10.1109/WAC.2006.375987.

[140] K. T. Song, C. W. Chiu, L. R. Kang, Y. X. Sun, and C. H. Meng, "Autonomous Docking in a Human-Robot Collaborative Environment of Automated Guided Vehicles," *2020 International Automatic Control Conference, CACS 2020*, 2020, doi: 10.1109/CACS50047.2020.9289713.

[141] J. Yue, W. Wen, J. Han, and L.-T. Hsu, "LiDAR Data Enrichment Using Deep Learning Based on High-Resolution Image: An Approach to Achieve High-Performance LiDAR SLAM Using Low-cost LiDAR," Aug. 2020, [Online]. Available: http://arxiv.org/abs/2008.03694

[142] A. M. Burgueño-Romero, J. R. Ruiz-Sarmiento, and J. Gonzalez-Jimenez, "Autonomous Docking of Mobile Robots by Reinforcement Learning Tackling the Sparse Reward Problem," *Lecture Notes in Computer Science (including subseries*

Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), vol. 12862 LNCS, pp. 392–403, 2021, doi: 10.1007/978-3-030-85099-9_32.

[143] M. L. Smith, L. N. Smith, and M. F. Hansen, “The quiet revolution in machine vision - a state-of-the-art survey paper, including historical review, perspectives, and future directions,” *Comput Ind*, vol. 130, p. 103472, 2021, doi: 10.1016/j.compind.2021.103472.

[144] C.-Y. Wang, A. Bochkovskiy, and H.-Y. M. Liao, “YOLOv7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors,” pp. 1–15, Jul. 2022, [Online]. Available: http://arxiv.org/abs/2207.02696

[145] B. K. Reddy, S. Bano, G. G. Reddy, R. Kommineni, and P. Y. Reddy, “Convolutional Network based Animal Recognition using YOLO and Darknet,” *Proceedings of the 6th International Conference on Inventive Computation Technologies, ICICT 2021*, pp. 1198–1203, 2021, doi: 10.1109/ICICT50816.2021.9358620.

[146] J. Xiao *et al.*, “Tiny object detection with context enhancement and feature purification,” *Expert Syst Appl*, vol. 211, no. October 2021, p. 118665, 2023, doi: 10.1016/j.eswa.2022.118665.

[147] H. Zhao, “YOLOv7-sea : Object Detection of Maritime UAV Images based on Improved,” pp. 233–238.

[148] Y. Zheng, H. Mamledesai, H. Imam, and R. Ahmad, “A Novel Deep Learning-based Automatic Damage Detection and Localization Method for Remanufacturing/Repair,” *Comput Aided Des Appl*, vol. 18, no. 6, pp. 1359–1372, Feb. 2021, doi: 10.14733/cadaps.2021.1359-1372.

[149] P. Martinez, M. Al-Hussein, and R. Ahmad, "Intelligent vision-based online inspection system of screw-fastening operations in light-gauge steel frame manufacturing," *International Journal of Advanced Manufacturing Technology*, vol. 109, no. 3–4, pp. 645–657, 2020, doi: 10.1007/s00170-020-05695-y.

[150] T. M. Wang, Y. Tao, and H. Liu, "Current Researches and Future Development Trend of Intelligent Robot: A Review," *International Journal of Automation and Computing*, vol. 15, no. 5, pp. 525–546, 2018, doi: 10.1007/s11633-018-1115-1.

[151] S. M. Khansari-Zadeh and A. Billard, "Learning stable nonlinear dynamical systems with Gaussian mixture models," *IEEE Transactions on Robotics*, vol. 27, no. 5, pp. 943–957, 2011, doi: 10.1109/TRO.2011.2159412.

[152] A. Schmitz, P. Maiolino, M. Maggiali, L. Natale, G. Cannata, and G. Metta, "Methods and technologies for the implementation of large-scale robot tactile sensors," *IEEE Transactions on Robotics*, vol. 27, no. 3, pp. 389–400, 2011, doi: 10.1109/TRO.2011.2132930.

[153] S. Levine, P. Pastor, A. Krizhevsky, J. Ibarz, and D. Quillen, "Learning hand-eye coordination for robotic grasping with deep learning and large-scale data collection," *International Journal of Robotics Research*, vol. 37, no. 4–5, pp. 421–436, 2018, doi: 10.1177/0278364917710318.

[154] C. Finn and S. Levine, "Deep visual foresight for planning robot motion," *Proc IEEE Int Conf Robot Autom*, pp. 2786–2793, 2017, doi: 10.1109/ICRA.2017.7989324.

[155] R. A. X. S. Fugui Xie, "An NC Code Based Machining Movement Simulation Method for a Parallel Robotic Machine," vol. 10463, Y. Huang, H. Wu, H. Liu, and

Z. Yin, Eds., in Lecture Notes in Computer Science, vol. 10463. , Cham: Springer International Publishing, 2017, pp. 3–13. doi: 10.1007/978-3-319-65292-4_1.

[156] P. Martinez, R. Ahmad, and M. Al-Hussein, "Real-time visual detection and correction of automatic screw operations in dimpled light-gauge steel framing with pre-drilled pilot holes," *Procedia Manuf*, vol. 34, pp. 798–803, 2019, doi: 10.1016/j.promfg.2019.06.204.

[157] E. Matheson, R. Minto, E. G. G. Zampieri, M. Faccio, and G. Rosati, "Human-robot collaboration in manufacturing applications: A review," *Robotics*, vol. 8, no. 4. MDPI AG, 2019. doi: 10.3390/robotics8040100.

[158] M. R. Khosravani, S. Nasiri, and T. Reinicke, "Intelligent knowledge-based system to improve injection molding process," *J Ind Inf Integr*, vol. 25, no. July 2021, p. 100275, 2022, doi: 10.1016/j.jii.2021.100275.

[159] L. Ying, M. Li, and J. Yang, "Agglomeration and driving factors of regional innovation space based on intelligent manufacturing and green economy," *Environ Technol Innov*, vol. 22, p. 101398, 2021, doi: 10.1016/j.eti.2021.101398.

[160] L. Xu, G. Li, P. Song, and W. Shao, "Vision-Based Intelligent Perceiving and Planning System of a 7-DoF Collaborative Robot," *Comput Intell Neurosci*, vol. 2021, 2021, doi: 10.1155/2021/5810371.

[161] I. Ahmed, S. Din, G. Jeon, F. Piccialli, and G. Fortino, "Towards Collaborative Robotics in Top View Surveillance: A Framework for Multiple Object Tracking by Detection Using Deep Learning," *IEEE/CAA Journal of Automatica Sinica*, vol. 8, no. 7, pp. 1253–1270, 2021, doi: 10.1109/JAS.2020.1003453.

[162] N. Dalal and B. Triggs, “Histograms of Oriented Gradients for Human Detection,” in *In Proceedings of the 2008 IEEE Conference on Computer Vision and Pattern Recognition*, 2005, pp. 886–893. [Online]. Available: http://lear.inrialpes.fr

[163] P. Felzenszwalb, D. McAllester, and D. Ramanan, “A Discriminatively Trained, Multiscale, Deformable Part Model.,” in *2008 IEEE Conference on Computer Vision and Pattern Recognition*, 2008, pp. 23–28.

[164] R. Girshick, J. Donahue, T. Darrell, and J. Malik, “Rich feature hierarchies for accurate object detection and semantic segmentation,” *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pp. 580–587, 2014, doi: 10.1109/CVPR.2014.81.

[165] R. Girshick, “Fast R-CNN,” *Proceedings of the IEEE International Conference on Computer Vision*, vol. 2015 Inter, pp. 1440–1448, 2015, doi: 10.1109/ICCV.2015.169.

[166] S. Ren, K. He, R. Girshick, and J. Sun, “Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks,” *IEEE Trans Pattern Anal Mach Intell*, vol. 39, no. 6, pp. 1137–1149, 2017, doi: 10.1109/TPAMI.2016.2577031.

[167] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, “You only look once: Unified, real-time object detection,” *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, vol. 2016-Decem, pp. 779–788, 2016, doi: 10.1109/CVPR.2016.91.

[168] W. Liu and D. Anguelov, “SSD: Single Shot MultiBox Detector,” in *Computer Vision – ECCV 2016*, B. Leibe, J. Matas, N. Sebe, and M. Welling, Eds., in Lecture

Notes in Computer Science, vol. 9905. Cham: Springer International Publishing, 2016, pp. 21–37. doi: 10.1007/978-3-319-46448-0.

[169] L. Leal-Taixe, C. Canton-Ferrer, and K. Schindler, “Learning by Tracking: Siamese CNN for Robust Target Association,” *IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops*, pp. 418–425, 2016, doi: 10.1109/CVPRW.2016.59.

[170] H. Lin, P. Yang, and F. Zhang, “Review of Scene Text Detection and Recognition,” *Archives of Computational Methods in Engineering*, vol. 27, no. 2, pp. 433–454, 2020, doi: 10.1007/s11831-019-09315-1.

[171] Y. Zhu, C. Yao, and X. Bai, “Scene text detection and recognition: recent advances and future trends,” *Front Comput Sci*, vol. 10, no. 1, pp. 19–36, 2016, doi: 10.1007/s11704-015-4488-0.

[172] M. Brisinello, R. Grbic, M. Vranjes, and D. Vranjes, “Review on text detection methods on scene images,” *Proceedings Elmar - International Symposium Electronics in Marine*, vol. 2019-Septe, no. September, pp. 51–56, 2019, doi: 10.1109/ELMAR.2019.8918680.

[173] M. Jaderberg, K. Simonyan, A. Vedaldi, and A. Zisserman, “Reading Text in the Wild with Convolutional Neural Networks,” *Int J Comput Vis*, vol. 116, no. 1, pp. 1–20, 2016, doi: 10.1007/s11263-015-0823-z.

[174] Z. Zhong, L. Jin, S. Zhang, and Z. Feng, “DeepText: A Unified Framework for Text Proposal Generation and Text Detection in Natural Images,” pp. 1–12, 2016, [Online]. Available: http://arxiv.org/abs/1605.07314

[175] W. He, X. Y. Zhang, F. Yin, and C. L. Liu, "Deep Direct Regression for Multi-oriented Scene Text Detection," *Proceedings of the IEEE International Conference on Computer Vision*, vol. 2017-Octob, pp. 745–753, 2017, doi: 10.1109/ICCV.2017.87.

[176] B. Su and S. Lu, "Accurate scene text recognition based on recurrent neural network," *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 9003, pp. 35–48, 2015, doi: 10.1007/978-3-319-16865-4_3.

[177] B. Shi, X. Bai, and C. Yao, "An End-to-End Trainable Neural Network for Image-Based Sequence Recognition and Its Application to Scene Text Recognition," *IEEE Trans Pattern Anal Mach Intell*, vol. 39, no. 11, pp. 2298–2304, 2017, doi: 10.1109/TPAMI.2016.2646371.

[178] C. Y. Lee and S. Osindero, "Recursive Recurrent Nets with Attention Modeling for OCR in the Wild," *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, vol. 2016-Decem, no. 3, pp. 2231–2239, 2016, doi: 10.1109/CVPR.2016.245.

[179] D. Bahdanau, K. H. Cho, and Y. Bengio, "Neural machine translation by jointly learning to align and translate," *3rd International Conference on Learning Representations, ICLR 2015 - Conference Track Proceedings*, pp. 1–15, 2015.

[180] M. Liao, B. Shi, X. Bai, X. Wang, and W. Liu, "TextBoxes: A fast text detector with a single deep neural network," *31st AAAI Conference on Artificial Intelligence, AAAI 2017*, pp. 4161–4167, 2017.

[181] M. Hradiš, J. Kotera, P. Zemčík, and F. Šroubek, "Convolutional Neural Networks for Direct Text Deblurring," no. 1, pp. 6.1-6.13, 2015, doi: 10.5244/c.29.6.

[182] S. Mahajan and R. Rani, *Text detection and localization in scene images: a broad review*, vol. 54, no. 6. Springer Netherlands, 2021. doi: 10.1007/s10462-021-10000-8.

[183] A. Graves, S. Fernández, M. Liwicki, H. Bunke, and J. Schmidhuber, "Unconstrained online handwriting recognition with recurrent neural networks," *Advances in Neural Information Processing Systems 20 - Proceedings of the 2007 Conference*, pp. 1–8, 2009.

[184] S. Dieleman, K. W. Willett, and J. Dambre, "Rotation-invariant convolutional neural networks for galaxy morphology prediction," *Mon Not R Astron Soc*, vol. 450, no. 2, pp. 1441–1459, 2015, doi: 10.1093/mnras/stv632.

[185] O. Russakovsky *et al.*, "ImageNet Large Scale Visual Recognition Challenge," *Int J Comput Vis*, vol. 115, no. 3, pp. 211–252, 2015, doi: 10.1007/s11263-015-0816-y.

[186] S. Long, X. He, and C. Yao, "Scene Text Detection and Recognition: The Deep Learning Era," *Int J Comput Vis*, vol. 129, no. 1, pp. 161–184, 2021, doi: 10.1007/s11263-020-01369-0.

[187] Z. Wojna *et al.*, "Attention-Based Extraction of Structured Information from Street View Imagery," *Proceedings of the International Conference on Document Analysis and Recognition, ICDAR*, vol. 1, pp. 844–850, 2017, doi: 10.1109/ICDAR.2017.143.

[188] M. S. Nadeem, "Textual Deblurring using Convolutional Neural Network Textual Deblurring using Convolutional Neural Network," 2021.

[189] C. Li, Y. Yang, K. He, S. Lin, and J. E. Hopcroft, “Single Image Reflection Removal through Cascaded Refinement,” *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pp. 3565–3574, 2020.

[190] Y. Li, M. Liu, Y. Yi, Q. Li, D. Ren, and W. Zuo, “Two-Stage Single Image Reflection Removal with Reflection-Aware Guidance,” vol. 1, 2020, [Online]. Available: http://arxiv.org/abs/2012.00945

[191] D. G. Lowe, “Distinctive image features from scale-invariant keypoints,” *Int J Comput Vis*, vol. 60, no. 2, pp. 91–110, 2004, doi: 10.1023/B:VISI.0000029664.99615.94.

[192] H. Bay, T. Tuytelaars, and L. Van Gool, “SURF: Speeded Up Robust Features,” *Computer Vision–ECCV 2006*, pp. 404–417, 2006, [Online]. Available: http://link.springer.com/chapter/10.1007/11744023_32

[193] D. Ciregan, U. Meier, and J. Schmidhuber, “Multi-column deep neural networks for image classification,” *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, no. February, pp. 3642–3649, 2012, doi: 10.1109/CVPR.2012.6248110.

[194] P. Luc, C. Couprie, S. Chintala, and J. Verbeek, “Semantic Segmentation using Adversarial Networks,” Nov. 2016, [Online]. Available: http://arxiv.org/abs/1611.08408

[195] E. Shelhamer, J. Long, and T. Darrell, “Fully Convolutional Networks for Semantic Segmentation,” *IEEE Trans Pattern Anal Mach Intell*, vol. 39, no. 4, pp. 640–651, 2017, doi: 10.1109/TPAMI.2016.2572683.

[196] B. Romera-Paredes and P. H. S. Torr, "Recurrent instance segmentation," *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 9910 LNCS, pp. 312–329, 2016, doi: 10.1007/978-3-319-46466-4_19.

[197] M. Ren and R. S. Zemel, "End-to-end instance segmentation with recurrent attention," *Proceedings - 30th IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017*, vol. 2017-Janua, pp. 293–301, 2017, doi: 10.1109/CVPR.2017.39.

[198] K. He, G. Gkioxari, P. Dollár, and R. Girshick, "Mask R-CNN," *IEEE Trans Pattern Anal Mach Intell*, vol. 42, no. 2, pp. 386–397, 2020, doi: 10.1109/TPAMI.2018.2844175.

[199] H. Z. Imam, H. Al-Musaibeli, Y. Zheng, P. Martinez, and R. Ahmad, "Vision-based spatial damage localization method for autonomous robotic laser cladding repair processes," *Robot Comput Integr Manuf*, vol. 80, no. September 2022, p. 102452, 2023, doi: 10.1016/j.rcim.2022.102452.

[200] P. He, W. Huang, Y. Qiao, C. C. Loy, and X. Tang, "Reading scene text in deep convolutional sequences," *30th AAAI Conference on Artificial Intelligence, AAAI 2016*, pp. 3501–3508, 2016, doi: 10.1609/aaai.v30i1.10465.

[201] Z. Tian, W. Huang, T. He, P. He, and Y. Qiao, "Detecting text in natural image with connectionist text proposal network," *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 9912 LNCS, pp. 56–72, 2016, doi: 10.1007/978-3-319-46484-8_4.

[202] X. Zhou *et al.*, “EAST: An efficient and accurate scene text detector,” *Proceedings - 30th IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017*, vol. 2017-Janua, pp. 2642–2651, 2017, doi: 10.1109/CVPR.2017.283.

[203] A. Graves, S. Fernández, F. Gomez, and J. Schmidhuber, “Connectionist temporal classification,” in *Proceedings of the 23rd international conference on Machine learning - ICML ’06*, New York, New York, USA: ACM Press, Feb. 2006, pp. 369–376. doi: 10.1145/1143844.1143891.

[204] Y. Gao, Y. Chen, J. Wang, M. Tang, and H. Lu, “Reading scene text with fully convolutional sequence modeling,” *Neurocomputing*, vol. 339, pp. 161–170, 2019, doi: 10.1016/j.neucom.2019.01.094.

[205] F. Yin, Y.-C. Wu, X.-Y. Zhang, and C.-L. Liu, “Scene Text Recognition with Sliding Convolutional Character Models,” 2017, [Online]. Available: http://arxiv.org/abs/1709.01727

[206] M. Liukkonen and T. N. Tsai, “Toward decentralized intelligence in manufacturing: recent trends in automatic identification of things,” *International Journal of Advanced Manufacturing Technology*, vol. 87, no. 9–12, pp. 2509–2531, 2016, doi: 10.1007/s00170-016-8628-y.

[207] E. Tekin, J. M. Coughlan, and H. Shen, “Real-time detection and reading of LED/LCD displays for visually impaired persons,” *2011 IEEE Workshop on Applications of Computer Vision, WACV 2011*, pp. 491–496, 2011, doi: 10.1109/WACV.2011.5711544.

[208] A. Cheng, S. Lu, and F. Gao, "Anomaly Detection of Tire Tiny Text: Mechanism and Method," *IEEE Transactions on Automation Science and Engineering*, vol. PP, pp. 1–18, 2023, doi: 10.1109/TASE.2023.3257831.

[209] G. Monteiro *et al.*, "A Comprehensive Framework for Industrial Sticker Information Recognition Using Advanced OCR and Object Detection Techniques," *Applied Sciences*, vol. 13, no. 12, p. 7320, 2023, doi: 10.3390/app13127320.

[210] M. Liao, Z. Zou, Z. Wan, C. Yao, and X. Bai, "Real-Time Scene Text Detection With Differentiable Binarization and Adaptive Scale Fusion," *IEEE Trans Pattern Anal Mach Intell*, vol. 45, no. 1, pp. 919–931, Jan. 2023, doi: 10.1109/TPAMI.2022.3155612.

[211] M. Liao, Z. Zhu, B. Shi, G. S. Xia, and X. Bai, "Rotation-Sensitive Regression for Oriented Scene Text Detection," *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pp. 5909–5918, 2018, doi: 10.1109/CVPR.2018.00619.

[212] F. Wang, L. Zhao, X. Li, X. Wang, and D. Tao, "Geometry-Aware Scene Text Detection with Instance Transformation Network," *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pp. 1381–1389, 2018, doi: 10.1109/CVPR.2018.00150.

[213] M. Chen, X. Zhang, M. Feng, and W. Xing, "Image deblurring method based on fourier transform," *2019 6th International Conference on Systems and Informatics, ICSAI 2019*, no. Icsai, pp. 1285–1289, 2019, doi: 10.1109/ICSAI48974.2019.9010430.

[214] N. Arvanitopoulos, R. Achanta, and S. Sabine, “Single Image Reflection Suppression,” *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017, doi: 10.1109/CVPR.2017.190.

[215] R. Y. O. Abiko, S. Member, and M. Ikehara, “Single Image Reflection Removal Based on GAN With Gradient Constraint,” *IEEE Access*, vol. 7, pp. 148790–148799, 2019, doi: 10.1109/ACCESS.2019.2947266.

[216] K. Simonyan and A. Zisserman, “Very deep convolutional networks for large-scale image recognition,” *3rd International Conference on Learning Representations, ICLR 2015 - Conference Track Proceedings*, pp. 1–14, 2015.

[217] A. Graves and J. Schmidhuber, “Framewise phoneme classification with bidirectional LSTM and other neural network architectures,” *Neural Networks*, vol. 18, no. 5–6, pp. 602–610, 2005, doi: 10.1016/j.neunet.2005.06.042.

[218] M. Liao, B. Shi, and X. Bai, “TextBoxes++: A Single-Shot Oriented Scene Text Detector,” *IEEE Transactions on Image Processing*, vol. 27, no. 8, pp. 3676–3690, 2018, doi: 10.1109/TIP.2018.2825107.

[219] E. Rainarli and K. E. Dewi, “The Implementation of Cosine Similarity to Calculate Text Relevance between Two Documents,” in *2nd International Conference on Computing and Applied Informatics 2017*, 2018, pp. 1–6.

[220] E. Sari, M. Belbahri, and V. P. Nia, “How Does Batch Normalization Help Binary Training?,” 2019, [Online]. Available: http://arxiv.org/abs/1909.09139

[221] X. Zhao, H. Jia, and Y. Ni, “A novel three-dimensional object detection with the modified you only look once method,” *Int J Adv Robot Syst*, vol. 15, no. 2, pp. 1–13, 2018, doi: 10.1177/1729881418765507.

[222] H. H. Asada, “Motion Control,” in *Robotics. Advanced Textbooks in Control and Signal Processing*, Springer, London, 2010, pp. 303–361. doi: 10.1007/978-1-84628-642-1_8.