



CONTEXT-AWARE HUMAN-ROBOT COLLABORATION IN ASSEMBLY

HONGYI LIU

Doctoral thesis
Royal Institute of Technology
School of Industrial Engineering and Management
Department of Production Engineering
SE-100 44 Stockholm

TRITA-ITM-AVL 2020:34
ISBN 978-91-7873-594-5

TRITA-ITM-AVL 2020:34
ISBN 978-91-7873-594-5

Mr
HONGYI LIU
Doctoral thesis

Academic thesis, which with the approval of the Royal Institute of Technology, will be presented for public review in fulfilment of the requirements for a Doctor of Engineering in Production Engineering. The public review is held online with Zoom on 2020-09-11 at 10am.

“The true delight is in the finding out rather than in the knowing”

— Isaac Asimov

Abstract

The PhD study is aiming to increase the accuracy and efficiency of human-robot collaborative (HRC) assembly systems. To achieve this goal, four main directions are investigated in this research. The first direction is HRC assembly context recognition, which focuses on the identification and recognition of relevant assembly context in the assembly environment. Valuable knowledge can be captured through the assembly context to increase assembly efficiency. The definition of assembly context is given, and recognition algorithms are designed. The second direction is multimodal robot control. Instead of coding, the possibility to control robots with multiple modalities is explored. The algorithm to increase the recognition accuracy of multimodal robot control is developed. The third direction is human motion prediction. Robots can be supported to anticipate and prepare for the human operator's next move with an accurate and timely prediction of the human operator's motion. Two different approaches are explored to predict human motions during the assembly operation. The efficiency of HRC assembly systems can be further boosted. The last direction of the study is remote HRC. A special scenario of HRC is explored where a human operator collaborates with a robot remotely. The scenario is investigated, a possible solution is also provided. Along with the four directions, key algorithms, system designs, and experiments are analysed. Furthermore, the advantages, drawbacks, and future directions of the approaches are given.

Sammanfattning

Doktorand Studien syfte är att öka noggrannheten och effektiviteten i human-robot collaborative (HRC) monteringsystem. För att uppnå detta mål, fyra huvudinriktningar undersöks i denna forskning. Den första riktningen är HRC monterings kontext igenkänning, som fokuserar på identifiering och igenkänning av relevant monterings kontext i monterings miljö. Värdefull information kan fångas genom monterings-kontext för att öka monterings-effektiviteten. Definition av monterings kontext anges och igenkännings-algoritmer utformas. Den andra riktningen är multimodal robotstyrning. Istället för kodning undersöks möjligheten att kontrollera robotar med flera modaliteter. Algoritmen för att öka igenkänningen noggrannheten för multimodal robotstyrning utvecklas. Den tredje riktningen är mänsklig rörelse förutsägelse. Rbotar kan stöttas för att förutse och förbereda för de mänskliga operatörernas nästa drag med en exakt och snabb förutsägelse av mänskliga operatörens rörelse. Två olika tillvägagångssätt utforskas för att förutsäga mänskliga rörelser under montering operationen. Effektiviteten hos HRC-monteringssystem kan förbättras ytterligare. Den sista riktningen för studien är fjärr HRC. Ett speciellt scenario med HRC utforskas där en mänsklig operatör samarbetar med en robot på distans. Scenariot undersöks, en möjlig lösning tillhandahålls också. Tillsammans med de fyra riktningarna analyseras nyckel algoritmer, systemdesign och experiment. Dessutom ges fördelarna, nackdelarna och framtida riktningarna för tillvägagångssätten.

Acknowledgement

First of all, I would like to sincerely thank my supervisor, Prof. Lihui Wang. He is the best supervisor I have ever had. Without his trust, advice, and example, I would not be able to complete this work. During my study, he gave me tremendous patience and trust. His insightful advice and feedback always helped me to find the right direction. His high standard of research ethics and hard-working attitudes also gave me significant influences.

Secondly, I would like to thank the professors that have helped and guided me during different stages of my PhD study: Amir Rashid, Mårten Björkman, Xi Vincent Wang, Antonio Maffei, Mauro Onori, Magnus Boman, Daniel Tesfamariam Semere, Robert X Gao. The help and influence of the professors positively shaped my PhD work.

Thirdly, I own a big thank to everyone worked/working at the production engineering department. During my PhD, I have been given help by many people and shared many joyful memories here. Also, I want to mention everyone worked and collaborated with me at KTH, other universities and several companies. I enjoyed every collaboration and friendship. Research can be difficult sometimes, therefore, many thanks to all my friends at our badminton club, the past five years would be very different without them.

Fourthly, I would like to specially mention the persons that spend time helped and gave me valuable feedback during the writing and revision of this thesis: Xi Vincent Wang, Daniel Tesfamariam Semere, Mo Chen, and Chengqi Li. The thesis would not be finished without their help and feedback.

Lastly, I would like to thank my parents. I would never imagine myself to even pursue a PhD degree. It is their influence and encouragement so I can finish my PhD and start my career in academia.

List of appended publications

Paper 1

H. Liu and L. Wang, "Gesture Recognition for Human-Robot Collaboration: A Review," *International Journal of Industrial Ergonomics*, Vol.68, pp.355-367, 2018. <https://doi.org/10.1016/j.ergon.2017.02.004>

Paper 2

H. Liu and L. Wang, "Human Motion Prediction for Human-Robot Collaboration," *Journal of Manufacturing Systems*, Vol.44, Part 2, pp.287-294, 2017. <https://doi.org/10.1016/j.jmsy.2017.04.009>

Paper 3

P. Wang, H. Liu, L. Wang and R. X. Gao, "Deep Learning-Based Human Motion Recognition for Predictive Context-Aware Human-Robot Collaboration," *CIRP Annals – Manufacturing Technology*, Vol.67, No.1, pp.17-20, 2018. <https://doi.org/10.1016/j.cirp.2018.04.066>

Paper 4

H. Liu, T. Fang, T. Zhou and L. Wang, "Towards Robust Human-Robot Collaborative Manufacturing: Multimodal Fusion," *IEEE Access*, Vol. 6, pp. 74762-74771, 2018. <https://doi.org/10.1109/ACCESS.2018.2884793>

Paper 5

H. Liu and L. Wang, "Remote Human-Robot Collaboration: A Cyber-Physical System Application for Hazard Manufacturing Environment," *Journal of Manufacturing Systems*, Vol.54, pp.24-34, 2020. <https://doi.org/10.1016/j.jmsy.2019.11.001>

Paper 6

J. Zhang, H. Liu, L. Wang and R. X. Gao, "Deep Learning-Based Human Motion Recognition for Predictive Context-Aware Human-Robot Collaboration," *CIRP Annals – Manufacturing Technology*, Vol.69, No.1, pp.9-12, 2020. <https://doi.org/10.1016/j.cirp.2020.04.077>

Other related publications

Journal papers

Paper 7

H. Liu and L. Wang, "Collision-Free Human-Robot Collaboration Based on Context Awareness," *Robotics and Computer-Integrated Manufacturing*, Vol.67, pp.101997, 2021. <https://doi.org/10.1016/j.rcim.2020.101997>

Paper 8

H. Liu, X. V. Wang and L. Wang, "On Machine Learning Driven Human-Robot Collaborative Assembly," submitted

Paper 9

Q. Wang, H. Liu, F. Ore, J. B. Hauge, L. Wang and S. Meijer, "Multi-Actor Perspective on Human Robotic Collaboration implementation in the automotive manufacturing industry, a Swedish case study," submitted

Conference papers

Paper 10

H. Liu and L. Wang, "An AR-based Worker Support System for Human-Robot Collaboration," in *Proceedings of the 27th International Conference on Flexible Automation and Intelligent Manufacturing*, Vol.11, pp.22-30, June 2017.

Paper 11

H. Liu, T. Fang, T. Zhou, Y. Wang and L. Wang, "Deep Learning-based Multimodal Control Interface for Human-Robot Collaboration," *Procedia CIRP of the 51th Conference on Manufacturing Systems*, Vol.72, pp.3-8, May 2018.

Paper 12

H. Liu, Y. Wang, W. Ji and L. Wang, "A Context-Aware Safety System for Human-Robot Collaboration," *Proceedings of the 28th International Conference on Flexible Automation and Intelligent Manufacturing*, Vol.17, pp.238-245, June 2018.

Paper 13

Y. Wang, H. Liu, W. Ji and L. Wang, "Realtime Collaborating With An Industrial Manipulator Using A Constraint-based Programming Approach," *Procedia CIRP* of the 51th Conference on Manufacturing Systems, Vol.72, pp.105–110, May 2018.

Paper 14

W. Ji, Y. Wang, H. Liu and L. Wang, "Interface Architecture Design for Minimum Programming in Human-Robot Collaboration," *Procedia CIRP* of the 51th Conference on Manufacturing Systems, Vol.72, pp.129–134, May 2018.

Paper 15

L. Wang, S. Liu, H. Liu and X. V. Wang, "Overview of human-robot collaboration in manufacturing," *Proceedings of 5th International Conference on the Industry 4.0 Model for Advanced Manufacturing*, pp.15-58, June 2020.

Book chapter

Paper 16

H. Liu and L. Wang, "Latest Developments of Gesture Recognition for Human-Robot Collaboration," submitted.

Contents

1	Introduction	1
1.1	Research background and motivation	1
1.2	Problem formulation	2
1.3	Main objective and research questions	4
1.4	Research methodology	5
1.5	Thesis outline	7
1.6	Research contributions	7
2	Human-robot collaboration	9
2.1	HRC definition	9
2.2	HRC classification	10
2.3	HRC bottlenecks	12
3	Machine learning algorithms	15
3.1	Algorithms classification	15
3.2	Support Vector Machine	18
3.3	Hidden Markov Model	19
3.4	Convolutional Neural Network	20
3.5	Long Short-Term Memory	21
4	HRC context recognition	25
4.1	Assembly context recognition	25
4.2	Transfer learning	26
4.3	Experiment	28
5	Multimodal robot control	31
5.1	Accuracy limitation	32
5.2	Multimodal fusion	33
5.3	Experiment	34

CONTENTS

6	Human motion prediction	41
6.1	Assembly motion prediction	41
6.2	Problem formulation and solution	42
6.3	Experiment	48
7	Remote HRC	55
7.1	Remote HRC	55
7.2	System design	56
7.3	Implementation	60
7.4	Experiment	63
8	Discussion and future works	67
8.1	Discussion on research question 1	67
8.2	Discussion on research question 2	68
8.3	Discussion on research question 3	69
8.4	Discussion on research question 4	71
9	Conclusion	73

List of Figures

1.1	Building blocks for future human-robot collaborative assembly systems	2
1.2	Research questions and related papers	4
1.3	Thesis outline	6
2.1	Illustration of an HRC setup in industrial assembly environment	10
3.1	Comparison of input data for learning models: (a) dataset for supervised learning models, (b) dataset for unsupervised learning models	16
3.2	Comparison of learning processes: (a) normal machine learning models, (b) deep learning models. Adapted from [55] . . .	17
3.3	Example of Support Vector Machine [42].	18
3.4	Example of a Hidden Markov model.	19
3.5	Example of Convolutional Neural Networks.	21
3.6	Illustration of LSTM model: (a) LSTM model example with three cells. (b) forget gate. (c) input gate. (d) output gate. Adapted from [48]	22
4.1	HRC context recognition	25
4.2	Illustration of transfer learning approach for assembly parts and tools recognition	26
4.3	Assembly part (left) and car engine after assembly (right) . .	27
4.4	Example of different assembly motions. (a) grasping. (b) holding. (c) assembling.	28
4.5	Learning curves of the two neural networks	29
4.6	Examples of video frames recognition: motion recognition (top); parts and tools recognition (bottom)	30
5.1	Example flowchart illustration of multimodal HRC	31
5.2	Gesture recognition for HRC	32
5.3	Illustration of multimodal fusion	34

LIST OF FIGURES

5.4	Visualised MFCC representation of speech commands. (a) Left. (b) Right (c) On. (d) Off. (e) Up. (f) Down.	35
5.5	Visualised hand motion commands. (a) Left. (b) Right (c) On. (d) Off. (e) Up. (f) Down.	36
5.6	The multimodal fusion training process comparison with trainable and non-trainable weights. (a)training accuracy for multimodal fusion. (b)training loss for multimodal fusion.	37
5.7	Confusion matrix of the multimodal fusion neural networks.	38
5.8	t-SNE visualisations of the test dataset after classification, the six different colours represent predicted different labels. (a)speech command model. (b)hand motion model. (c)body motion model. (d)fused model.	39
6.1	Human motion prediction	41
6.2	Example of task-level representation in an assembly station	43
6.3	An HMM model representation of a human operator’s assembly motions	43
6.4	The HMM model forward and backward procedure [43]	45
6.5	Skeleton model estimation	46
6.6	RNN for HRC motion prediction	47
6.7	HMM state transition and observation probability graph of the assembly case (a) state transition probability matrix graph; (b) state observation probability graph	49
6.8	Engine assembly setup	50
6.9	Engine assembly workstation	51
6.10	Transitions among <i>handover</i> , <i>standing</i> and <i>installation</i>	52
6.11	Illustrations of future motion trajectory prediction for HRC	52
7.1	Collaborative workstation: a human operator lead-through a collaborative robot with real-time display of the remote assembly parts	55
7.2	Remote workstation: an industrial robot working in dangerous environment with control parameters from the collaborative workstation	56
7.3	Overview of four working modes	57
7.4	Remote robot control system	58
7.5	Model-driven display system	60
7.6	Simplified system overview	60
7.7	ROS Multimaster package	61
7.8	Overview of the final system	62
7.9	Implemented system (a)virtual representation of the environment (b)photo of the physical environment	63

LIST OF FIGURES

7.10 Screenshots of the test result of Mode 4 64
7.11 Test result of Mode 1 65
7.12 Response time comparison of experiment on Modes 1, 2, 3
and 4 65

LIST OF FIGURES

List of Tables

2.1	Technical features in different efficiency levels of HRC assembly systems	11
3.1	Common machine learning models and classification	15
5.1	Test accuracy of unimodal and multimodal neural networks. .	38
6.1	States and observation symbols defined for the assembly task	49
6.2	Prediction MSE for MLP baseline and RNN model	51

LIST OF TABLES

Chapter 1

Introduction

This dissertation studies the realisation of context-aware human-robot collaboration (HRC) in industrial assembly with a focus on the improvement of algorithm accuracy and collaboration efficiency. This chapter starts with an explanation of the research background and motivation. Following the main problem formulation, the main objective and research questions are given, the adapted research methodologies are summarised, thesis outline is provided. Finally, the main research contribution of all the appended papers is outlined.

1.1 Research background and motivation

According to the data from the United Nations, population ageing is already a global phenomenon [1]. In 2019, the number of persons aged 65 years or over is 703 million globally, which is 9% of the global population. It is projected that the number will be double to 1.5 billion in 2050, which is equal to 16% of the global population. In manufacturing industries, consequently, ageing workforce can result in human operator shortage and increasing ergonomic problems [2].

Industrial robots, on the other hand, have been increasingly important in several manufacturing industries [3]. According to the report from the International Federation of Robotics (IFR), the newly installed industrial robots in 2018 reached 400,000 units, a 6% increase compared with last year [4]. In the automotive and electronics industries, industrial robots can work on dangerous, repetitive and ergonomically challenging tasks [5].

Recently, the concept of HRC assembly is a hot topic in robotics research. Different from the conventional industrial robotic assembly systems, HRC assembly systems can utilise the advantages from both human operators and industrial robots [6]–[8]. In HRC assembly, human operators can be the

team leaders and focus on interesting and problem-solving tasks, whereas robots can perform dangerous and physically challenging tasks. However, to achieve smooth collaboration between human operators and robots, there are still technology gaps to be filled. The current HRC assembly systems still suffer from problems such as low algorithm accuracy and low collaboration efficiency [8]. If the gaps are filled, robots and HRC assembly systems can contribute to the fight against the ageing workforce challenges. The human operator shortage problem and operator ergonomic problem can be solved.

1.2 Problem formulation

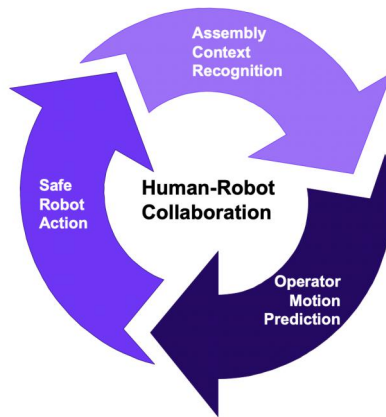


Figure 1.1: Building blocks for future human-robot collaborative assembly systems

To improve current HRC assembly systems, the author proposed a vision of the future HRC assembly systems, illustrated in Figure 1.1. There are three building blocks for an HRC assembly system: assembly context recognition, operator motion prediction, and safe robot action. Assembly context recognition refers to the recognition of relevant assembly context in the assembly environment. Operator motion prediction refers to the reasoning and inference on human operators' possible future moves, based on the recognised assembly context. Safe robot action refers to the triggering and operator-safe control of a robot, based on human operators' possible future moves. In such a vision, human operators can collaborate with robots safely and efficiently, with support from accurate recognition algorithms.

Although the vision of future HRC assembly is clear, the development

of HRC assembly systems is still at the beginning stage. To allow human operators and industrial robots to work together in the same environment at the same time, one of the prerequisites is the guarantee of human operators' safety. There have been many research efforts focused on active collision avoidance between human operators and robots [9]–[13]. With the active collision avoidance system, human operators can work in a collaborative environment safely without jeopardising efficiency. Apart from the development of safety systems, there are many different technological challenges that still need to be solved [8], [14], [15]. In the next sub-section, three of the most important challenges of HRC assembly will be summarised.

1.2.1 Challenges of HRC assembly

1. Methodology of HRC assembly system design: for all research fields, there should be a systematic analytical method that can support the practitioner in overall thinking and study. For instance, design thinking can be the methodology for industrial designers, lean production can be the methodology for production engineers. Due to the fact that the HRC assembly research is still in its initial stage, no mature methodology can be applied yet. HRC assembly systems involve much more human-related uncertainties than traditional assembly systems where everything is pre-programmed and hence under strict control. The integration of human operators and robots in the same environment will be challenging.

In this dissertation, the author will explore a first step towards HRC assembly system methodology.

2. HRC assembly system efficiency: the current HRC assembly systems still suffer from low-efficiency problem. The assembly efficiency can be lower than human assembly teams or robotics assembly teams. There is critical capability missing in the current HRC assembly system. There are very few sensors installed as well as intelligence algorithms applied.

The author will explore context awareness, and human motion prediction to improve the efficiency of HRC assembly systems.

3. HRC algorithms accuracy: similar to many digital systems that are supported by real-time intelligent decision making, HRC assembly systems require the capability of handling sensor data input from different modalities and formats. Specific algorithms need to be designed for timely and accurate intelligent decision making.

In this dissertation, the author will propose and develop different algorithms to improve the accuracy of HRC assembly systems.

1.2.2 Problem statement

As summarised in the previous section, the development of HRC assembly systems is still at its initial phase. The primary problem of HRC assembly system research can be summarised as:

There is a lack of algorithm accuracy and collaboration efficiency in current HRC assembly systems.

1.3 Main objective and research questions

Given the problem statement, the main objective of the paper can be formulated as:

To improve the accuracy and efficiency of HRC assembly systems.

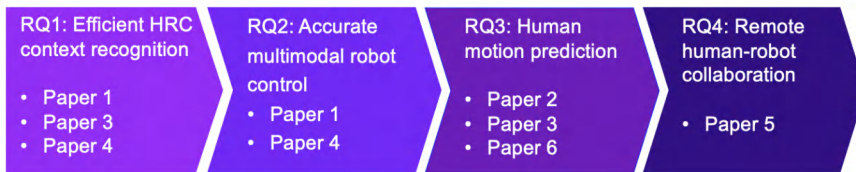


Figure 1.2: Research questions and related papers

Following the main objective, the associated research questions are formulated and presented as:

- Research Question 1 (RQ1): what is HRC context and how to perceive HRC context effectively?
- Research Question 2 (RQ2): how to increase the accuracy of multimodal robot control?
- Research Question 3 (RQ3): how to predict human operators' motion to further improve the assembly efficiency?
- Research Question 4 (RQ4): how to achieve HRC lead-through remotely with efficient response?

1.4 Research methodology

In this thesis, mainly two research methodologies are applied: *systematic literature review methodology* [16], [17] and *case study methodology* [18], [19].

1.4.1 Systematic literature review

The definition of systematic literature review methodology is: "A systematic, explicit, comprehensive, and reproducible method for identifying, evaluating, and synthesizing the existing body of completed and recorded work produced by researchers, scholars, and practitioners" [20]. For all research fields, research activities normally starts with the exercise of literature review [16]. By conducting literature review, researchers can gain insight into the research field, consolidate and even solve certain research questions. There are also several different approaches for literature review, such as, systematic literature review, semi-systematic review, and integrative review [20]. In this thesis, the author adapted the systematic literature review method. The systematic literature review method can provide the possibility to identify all empirical evidences in the literature to understand and answer specific research questions.

In this thesis, HRC is a novel topic in the research field of industrial assembly. To understand the state-of-the-art technological approaches, there requires a systematic review effort into the current literature in other research fields such as computer vision and human-computer interaction. This is especially applicable in Paper 1 where systematic literature review method is applied. After a systematic literature review of different sensor technology, gesture identification, gesture tracking, and gesture classification, Paper 1 provides a clear view of the overall process flow of gesture recognition for HRC. The paper can provide a solid foundation for the HRC research in industrial assembly.

1.4.2 Case study methodology

The definition of case study methodology is: "An empirical inquiry that investigates a contemporary phenomenon within its real life context, especially when the boundaries between phenomenon and context are not clearly evident" [21]. Case study methodology can facilitate the exploration and understanding of a certain research question from multiple directions [18]. The case study methodology is especially applicable in new research direction when existing theories and methods are inadequate [21], [22]. There is also an important design choice to make due to the natural focus of the case study. The researcher should know exactly what is interested and design

CHAPTER 1. INTRODUCTION

the case accordingly [23]. The difficult part of a case study is to lift the investigation from a descriptive report into generated knowledge [22].

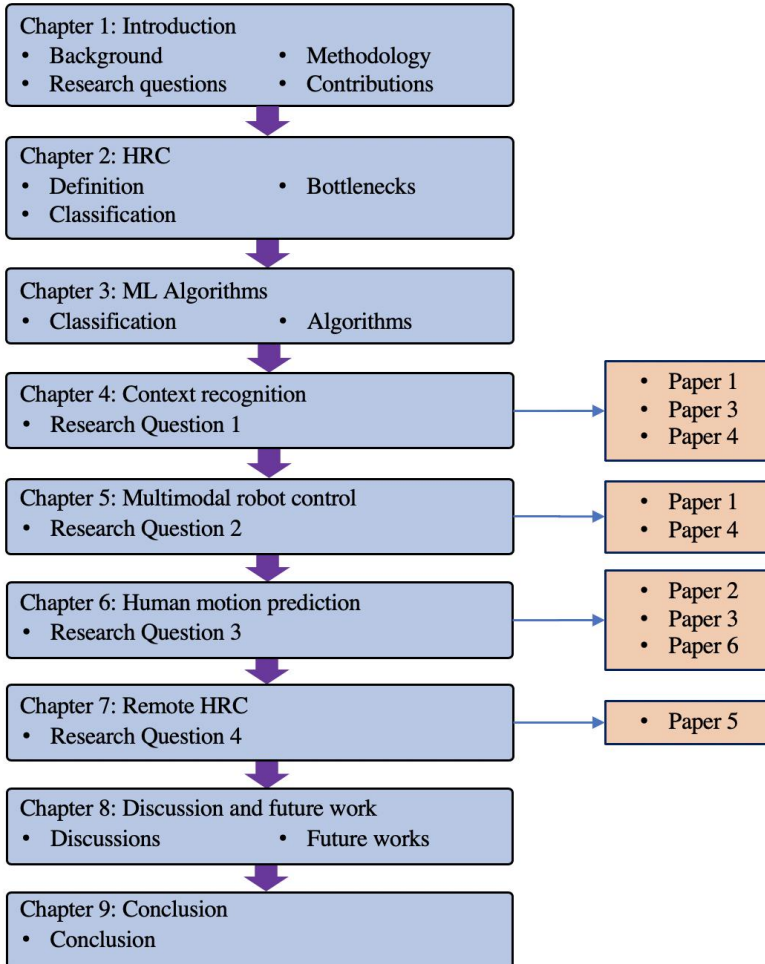


Figure 1.3: Thesis outline

Due to the nature of the thesis, case study methodology is especially applicable, as the HRC assembly is a new concept for industrial assembly, and industrial assembly line setup is also not relevant for long-term academic experiment and study. For most of the papers appended to the thesis, the author designed case studies following the general steps: questions, propositions, units of analysis, the logic linking the data to the propositions,

and finally the criteria for interpreting findings.

1.5 Thesis outline

The outline of the thesis is shown in Figure 1.3. The corresponding chapters, research questions, and the associated papers are summarised. In total, the thesis consists of 8 Chapters. Chapter 1 is the introduction Chapter where background is explained and objectives and research questions raised. Chapter 2 reviews the related literature. Chapter 3 provides a summarisation of the machine learning algorithms that are frequently used in the thesis. Chapter 4 explores the HRC context and context recognition. Chapter 5 introduces the attempt to increase the accuracy of multimodal robot control. Chapter 6 discusses the possibility to improve HRC assembly efficiency by human motion prediction. Chapter 7 analyses HRC in a remote setup. Chapter 8 summarises the thesis and provides future research directions. Chapter 9 gives a final conclusion.

1.6 Research contributions

In this section, the summaries of the contributions for all appended papers are given.

Paper 1: gesture recognition in HRC related literature has been systematically reviewed. The three-steps algorithm process of gesture recognition is identified. Different types of gesture recognition hardware are classified and summarised. The algorithms for gesture recognition are also compared and analysed. The key challenges and trends for gesture recognition in HRC are provided. The paper contributes to the fundamentals for the future development of multimodal HRC and context recognition.

In paper 1, the author mainly responsible for literature searching, literature analysis, data analysis, graph design and paper writing.

Paper 2: the author aims to propose and validate the possibility to improve the efficiency of HRC assembly by predicting human operators' motion during assembly. The problem of human motion prediction in assembly is formulated. A case study is conducted with the adaption of Hidden Markov model (HMM) as the motion prediction algorithm. The result indicates that human operators' motion during assembly is predictable to a certain extent. The paper is the first step in the research direction of human motion prediction in assembly.

In paper 2, the author mainly responsible for literature review, method design, experiment, data analysis and paper writing.

CHAPTER 1. INTRODUCTION

Paper 3: a deep learning-based human motion recognition method is proposed. The proposed method is based on the deep convolutional neural network (DCNN) and transfer learning. The proposed method is tested with a case study of assembly parts recognition and assembly motion recognition. The human motion recognition method proposed in this paper can be used to facilitate the human motion prediction method introduced in Paper 2. The assembly parts recognition method can also be used as a method for HRC context recognition.

In paper 3, the author mainly responsible for literature review, experiment and part of the paper writing.

Paper 4: a multimodal fusion method is proposed to improve the robustness of multimodal robot control. The proposed method is based on the fusion of multimodal neural networks. Data with different modalities can be used as input for different neural networks, and the processed information can be further fused with the proposed method. Result of the experiment shows that the multimodal fusion network outperformed single modality networks. The paper provided the possibility to improve multimodal robot control recognition accuracy by multimodal fusion. Several proposed methods can also be used for HRC context recognition.

In paper 4, the author mainly responsible for literature review, method design, part of the experiment, part of the data analysis and part of the paper writing.

Paper 5: a remote HRC system is proposed for hazard manufacturing environments. The proposed system is following the idea of the cyber-physical system. The system enables a human operator lead-through a remote robot with a local collaborative robot. The system is designed in four different modes to accommodate the different applications' needs. The proposed method is designed, implemented and tested.

In paper 5, the author mainly responsible for literature review, method design, experiment, data analysis and paper writing.

Paper 6: a method for human operators trajectory prediction is proposed. This paper closely followed the idea of Paper 2 and Paper 3. A specially designed deep learning algorithm is developed to provide an accurate prediction of human operators' future motion trajectory. With the knowledge of future human motion trajectory during assembly, the safety and efficiency of HRC assembly systems can be further improved.

In paper 6, the author mainly responsible for method design, experiment and part of the paper writing.

Chapter 2

Human-robot collaboration

This chapter covers an overview of current HRC research for industrial assembly. The definition, classification, and bottlenecks of current HRC research is summarised.

2.1 HRC definition

Traditionally, in robot assembly stations, there is no human operator allowed in the assembly cell. Robots are pre-programmed, assembly parts are precisely placed. Robots should follow the exact plan of execution. The traditional robot assembly stations enjoy advantages such as high efficiency and reliability. On the other hand, in human assembly stations, no robot is deployed. Human operators can collaborate and follow the assembly instructions flexibly to some extent. As a result, the human assembly stations perform better in tasks that require hand-eye coordination and haptic feedback, as problems during assembly can be solved instantly.

While combining the advantages of both human operators and robots, HRC aims to achieve better efficiency by seamless collaboration between human operators and robots. The concept of HRC in industrial assembly can be defined as: human operators and robots share and execute tasks according to their capabilities at the assembly line. The basic setup of HRC for assembly discussed in this thesis is illustrated in Figure 2.1. There is a robot, a human operator, an assembly station, assembly parts and several sensors in the illustration. The human operator can instruct and code the robot intuitively with multi-modalities. The assembly context, such as, assembly parts, human operator's activities, assembly situation is constantly monitored by sensors and understood by the robot. Thus, the assembly context is transparent for both the human operator and the robot. The possibility to predict the human operator's next motion enables the robot

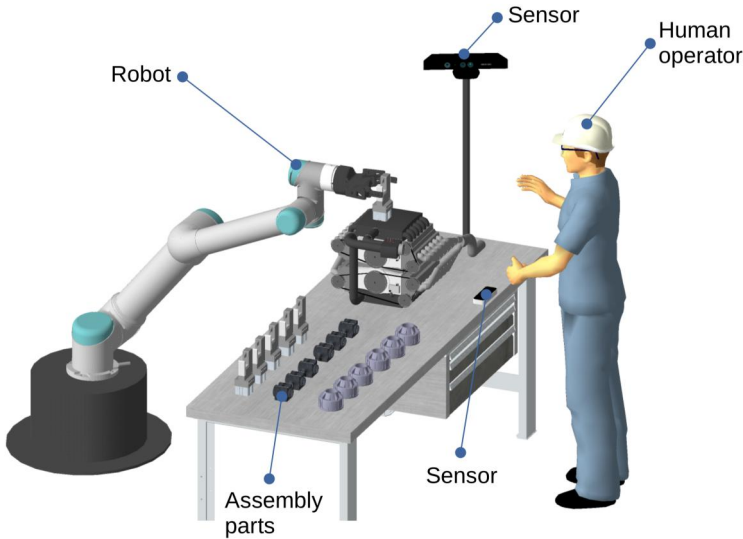


Figure 2.1: Illustration of an HRC setup in industrial assembly environment

to anticipate and better collaborate with human operators, under the same plan of assembly execution. Finally, the above-mentioned hardware and software functionalities should be integrated, data should flow freely and timely, algorithms should be efficient and accurate.

2.2 HRC classification

HRC systems may be classified differently according to different focuses and methods. Many recent papers presented new efforts to classify HRC systems [8], [24]–[27]. The classification efforts can be summarised into two directions: classify on technical aspects and classify on the human-robot team:

- Classify on technical aspects:

One of the classification approaches summarised current HRC systems into three levels: safety, coexistence, and collaboration, where safety level is the fundamental basics, coexistence level provides additional technical features such as task sharing, and collaboration levels involve complex technical features that require coordination and interaction [25], [27]. Another classification approach classified the different types of HRC systems by practical functionalities [24]. The

2.2. HRC CLASSIFICATION

HRC systems are classified into four types: safety-rated monitored stop, hand guiding, speed and separation monitoring, and power and force limiting. The four types of HRC systems require drastically different technical features. The consequences are that different types of HRC systems can provide different functionalities.

- Classify on human-robot team:

Recent literature classified HRC systems with a focus on the HRC team instead of technical aspects. The classification effort included both human operators and robotic systems. One of the approaches classified the HRC teams according to teams' compositions and team roles [26], [28]. The classification is done according to: firstly, the role of leader and follower, and secondly, the size of the team and their working relations (working alone or collaborating) in the team. A recent comprehensive approach also classified the human-robot relationships according to the shared different elements [8]. The shared elements are summarised as: workspace, direct contact, working task, resource, simultaneous process, sequential process. By identifying the shared elements, four different levels of collaboration (coexistence, interaction, cooperation, collaboration) are classified accordingly.

Technical features	Low-level	Medium-level	High-level
Active collision avoidance	✓	✓	✓
Task sharing	✓	✓	✓
Human tracking	✓	✓	✓
Assembly context recognition		✓	✓
Multimodal robot control		✓	✓
Task re-planning		✓	✓
Human motion recognition		✓	✓
Human motion prediction			✓

Table 2.1: Technical features in different efficiency levels of HRC assembly systems

In this thesis, the author will combine the above two approaches and provide a classification by adding a metric for comparison. The classification will still be done to focus on the HRC technical aspects. To be more specific, the author will classify the HRC assembly systems according to the attribute of HRC technical features using the metric of overall *system efficiency*. The author evaluates the HRC systems and their related technical features according to the efficiency of the overall system. The HRC technical features are classified into three types, as shown in Table 2.1:

- Low-level systems: for HRC systems with low-level efficiency, the overall assembly efficiency is expected to be lower than human assembly teams. Technical features such as active collision avoidance, task sharing, and human tracking can track human operators and provide a shared safe working environment. Human operators can work safely alongside robots. Robots can be pre-programmed to share the task of heavy parts handling, human operators' ergonomic load can be released.
- Medium-level systems: for HRC systems with medium-level efficiency, the overall assembly efficiency is expected to be lower or similar to human assembly teams. Technical features such as assembly context recognition, multimodal robot control, task re-planning, and motion recognition can further improve collaboration efficiency. Beyond safety, human operators can instruct robots to conduct simple tasks with intuitive multimodal commands and assembly context recognition. Assembly context such as assembly parts and tools can also be recognised by robots. The recognition of assembly context can enable the capability such as automated assembly parts picking. The possibility to re-plan the tasks can also improve assembly efficiency and flexibility.
- High-level systems: for HRC systems with high-level efficiency, the overall assembly efficiency is expected to be higher than human assembly teams. Technical features such as human motion prediction are provided. With the possibility to predict the human operator's future motion, the robots can further improve assembly efficiency with corresponding reactions. The technical features in high-level HRC systems will be more in the future to further boost the efficiency of HRC assembly systems.

2.3 HRC bottlenecks

In this section, the author will summarise the bottlenecks for the current HRC assembly systems. The bottlenecks will be described with a focus on the following two aspects: accuracy and robustness.

2.3.1 Accuracy

One of the main bottlenecks for current HRC systems is the recognition accuracy of the algorithms [25], [29]–[31]. Due to the industrial robots' extreme accurate mechanical control, in traditional fenced robotics assembly

work stations, there is no accuracy issue. In human assembly work station, the assembly sequences are defined and tested perfectly, the human operators can simply follow the pre-defined routines and execute assembly tasks. Human operators also have the excellent soft touch and hand-eye coordination skills. The neural sensory-motor and the touch feedback loop can greatly increase the required accuracy.

Whereas in HRC assembly systems, human operators and robots are co-located in the same environment. Collaborative tasks are expected to be done together by human operators and robots. Since human operators are now added in the assembly loop, many robot assembly sequences and tasks cannot be predefined as before. Human operators can still perform similar tasks at the same accuracy level as before, while robots are required to execute more collaborative tasks together with human operators. The new requirement is that robots have to improve the sensory-motor feedback capabilities to adapt to the new changes. In order to work alongside human operators, human-level sensory recognition accuracy is required. For example, precise assembly parts recognition and localisation, precise human intention estimation, and precise haptic feedback. Much work has been done on this topic in the relevant research field, the accuracy issues are not yet fully solved. New sensors, algorithms, and methodologies are needed.

2.3.2 Robustness

Another fundamental restriction for current HRC systems is the lack of system robustness [29], [32]–[34]. The robustness problem can be reflected from the following two different directions:

- HRC safety:

It is straightforward that system robustness issues can result in problems such as human safety. The robustness problem of the safety system is the most crucial problem that can happen in HRC assembly systems. The consequences of an unreliable safety system are catastrophic. The potential solution can be the improvement of sensor reliability, and redundancy [25], [35], [36]. For instance, since 2D sensors can be sensitive to light and dust in an industrial setting, researchers have suggested using multiple sensors in the same area to ensure safety with redundancy [36]. It is also important to standardise the safety practice for HRC assembly systems [25], [36]. For instance, one of the recent ISO standard: ISO 10218-1/2 [37], [38] provides clear guidelines for the safety of HRC assembly systems. There are four collaborative operative modes identified by the robot safety standard: *Safety-rated Monitored Stop*, *Hand Guiding*, *Speed and Separation Monitoring*, *Power and Force Limiting* [25].

- Recognition efficiency:

Robustness issues can also contribute to the lower system recognition efficiency for HRC assembly systems. Even if the machine learning algorithms can provide accurate prediction, the prediction still can be affected by sensor reliability, and recognition algorithm limitations [25], [39]. For instance, the recognition model is not well trained with adequate training dataset, where some of the examples used in the real assembly environment are not included in the training dataset. Although the recognition model is perfectly trained, the recognition efficiency can be significantly reduced. For such issues, the development process of HRC assembly systems' recognition models should be further standardised and optimised.

Chapter 3

Machine learning algorithms

The thesis greatly depends on machine learning algorithms. For most of the research questions and in most of the appended papers, machine learning algorithms have been adapted as the algorithm for data-driven decision-making tool. In this chapter, the author will provide an analysis of some of the most commonly used machine learning algorithms.

ML models	Supervised/ Unsupervised	Discriminative/ Generative /	DL/ Non-DL
K-means [40]	Unsupervised	Generative	Non-DL
KNN [41]	Supervised	Discriminative	Non-DL
SVM [42]	Supervised	Discriminative	Non-DL
HMM [43]	Supervised	Discriminative	Non-DL
Random Forest [44]	Supervised	Discriminative	Non-DL
XGBoost [45]	Supervised	Discriminative	Non-DL
CNN [46]	Supervised	Discriminative	DL
RNN [47]	Supervised	Discriminative	DL
LSTM [48]	Supervised	Discriminative	DL
Naive Bayes [49]	Supervised	Generative	Non-DL
GMM [50]	Supervised	Generative	Non-DL
GANs [51]	Semi-Supervised	Generative	DL

Table 3.1: Common machine learning models and classification

3.1 Algorithms classification

Machine learning enables a computer to understand the underlying patterns from a given dataset. In this section, the author starts with a comparison of machine learning models with different classification criteria. Following

the comparison, different machine learning models will be introduced and reviewed in the following sections.

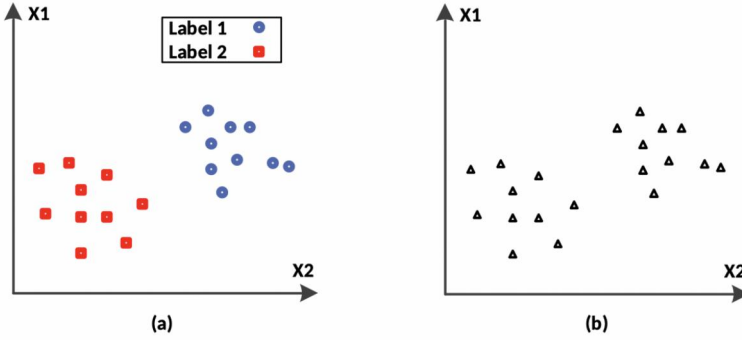


Figure 3.1: Comparison of input data for learning models: (a) dataset for supervised learning models, (b) dataset for unsupervised learning models

Common machine learning models are listed in Table 3.1. The machine learning models can be classified according to different criteria:

- Supervised learning model and unsupervised learning model [52]: supervised learning models are the most common type of machine learning models, which needs training data with corresponding labels. As shown in Figure 3.1(a), the input data of a supervised learning model is similar to:

$$\{(x^1, y^1), (x^2, y^2) \dots (x^n, y^n)\} \quad (3.1)$$

whereas unsupervised learning models can learn the structures or patterns from input data without corresponding labels. As shown in Figure 3.1(b), the input data of unsupervised learning models is similar to:

$$\{x^1, x^2 \dots x^n\} \quad (3.2)$$

In HRC assembly systems, most of the adapted machine learning models belong to supervised learning type. In this chapter, the author will mainly focus on the supervised learning models.

- Generative models and discriminative models [53]: generative models classify dataset based on the relation between features input and the generation of the result. By utilising Bayes rules, generative classifiers learn the model of joint probability $p(x, y)$ and calculate $p(y | x)$ to get the most likely y , where x is the input data and y is the class

label. While discriminative models classify data by just learning the decision boundaries. The input x is mapped to class label y directly. Both generative and discriminative methods are commonly used in HRC assembly systems.

- Deep learning models and non-deep learning models: a recent advancement in the machine learning field is the emerging of deep learning models [54]. Deep learning enables performance improvement by learning through the deep representations of a given dataset. Compared with other machine learning models which require domain-specific expertise to extract features before classification, deep learning models offer flexibility and adaptability to model nonlinear patterns by combining the feature extractor and classifier with deep neural networks [55]. Popular algorithms are Convolutional Neural Networks (CNN), Recurrent Neural Networks (RNN) and Generative Adversarial Nets (GANs). It is worth to notice that deep learning models do not guarantee better performance than other models. However, the manual works required in the learning process are reduced. The comparison of normal machine learning models and deep learning models are shown in Figure 3.2.

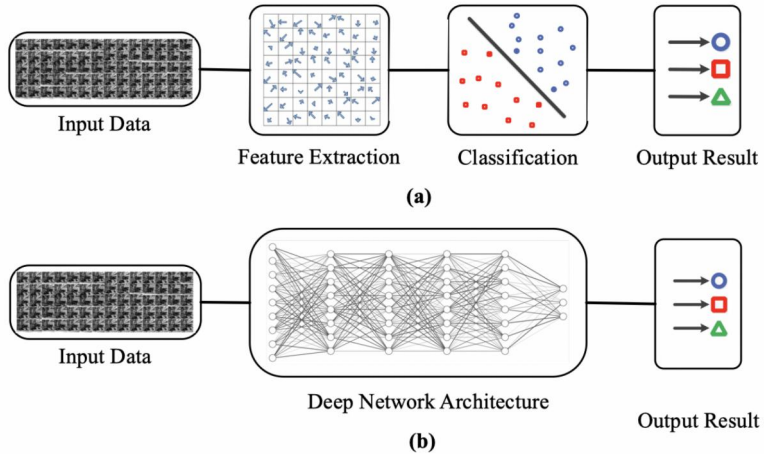


Figure 3.2: Comparison of learning processes: (a) normal machine learning models, (b) deep learning models. Adapted from [55]

Start from the next section, some of the most commonly adapted machine learning models in the thesis are introduced.

3.2 Support Vector Machine

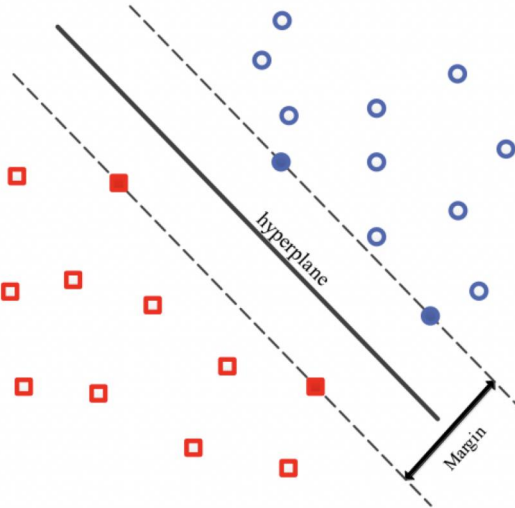


Figure 3.3: Example of Support Vector Machine [42].

As shown in Figure 3.3, Support Vector Machine (SVM) is a discriminative classifier defined by a separating hyperplane [42]. Classification decision boundaries are identified by maximising a margin distance. The optimal separation hyperplane maximises the margin of training data. The data points closest to the optimal hyperplane are called support vectors. The support vectors decide the margin length. As shown in Figure 3.3, the support vectors are solid points filled with colour. Kernel trick was introduced by Scholkopf [56]. Kernel trick enables linear SVM in non-linear situations. Kernel trick can transform low-dimensional training data into high-dimensional feature space with nonlinear methods, as expressed in Equation 3.3:

$$f(x) = \text{sign} \left(\sum_{i=1}^l v_i \cdot k(x, x_i) + b \right) \quad (3.3)$$

where v_i is computed by the solution of a quadratic programming problem, $k(x, x_i)$ is the kernel function. Different classification problems require different kernels. A common problem for SVM is that the number of support vectors grows linearly with the size of the training sets. Some researchers proposed Relevance Vector Machine (RVM) to solve the problem [57]. The

RVM solves the problem within a probabilistic framework. RVM algorithm is proven to be more parsimonious than the SVM algorithm.

3.3 Hidden Markov Model

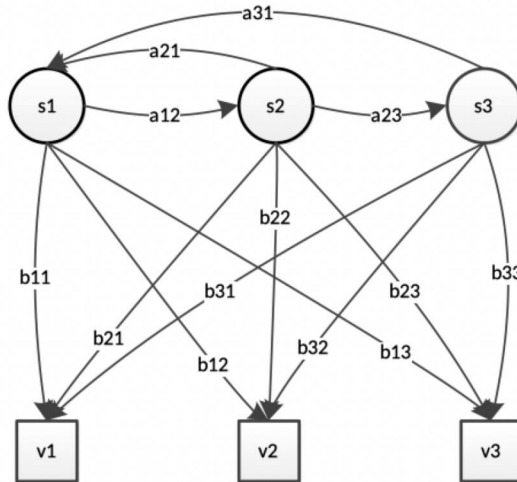


Figure 3.4: Example of a Hidden Markov model.

Hidden Markov model (HMM) is a statistical Markov model with hidden states. It is based on the Markov chain assumption [43]. The states in HMM is not observable. The hidden states have different transition probabilities. The output generated from the states is observable. Each state has a probability distribution of generating different outputs. As the example shown in Figure 3.4, an HMM can be defined from the following elements [43]:

- The states are denoted as $S = \{s_1, s_2, \dots, s_N\}$. N is the number of states in the model. The state sequence is $Q = \{q_1, q_2, \dots, q_t\}$. The state at time t is q_t .
- The observation symbols are denoted as $V = \{v_1, v_2, \dots, v_M\}$. M is the number of distinct observation symbols per state. The observation sequence is $O = \{o_1, o_2, \dots, o_t\}$. o_t is the observation at t .
- The state transition probability distribution is $A = \{a_{ij}\}$, where

$$a_{ij} = P(q_{t+1} = s_j | q_t = s_i), 1 \leq i, j \leq N \quad (3.4)$$

- The observation symbol probability distribution is $B = \{b_j(k)\}$, where

$$b_j(k) = P(o_t = v_k \mid q_t = s_j), 1 \leq j \leq N, 1 \leq k \leq M \quad (3.5)$$

- The initial state distribution is $\pi = \{\pi_i\}$ where

$$\pi_i = P(q_1 = s_i), 1 \leq i \leq N \quad (3.6)$$

It is possible to summarise from the above that a complete HMM requires the specification of parameters N and M , observation symbols, and probability measures A , B , and π . A compact notation is introduced to indicate the complete model parameters:

$$\lambda = (A, B, \pi) \quad (3.7)$$

As introduced by Rabiner [43], three fundamental problems can be solved by HMM in applications:

- Given observation sequence $O = \{o_1, o_2, \dots, o_t\}$ and a model $\lambda = (A, B, \pi)$, how to compute the probability of the observation sequence $P(O \mid \lambda)$;
- Given observation sequence $O = \{o_1, o_2, \dots, o_t\}$ and a model $\lambda = (A, B, \pi)$, how to choose the optimal state sequence $Q = \{q_1, q_2, \dots, q_t\}$; and
- How to adjust model parameters $\lambda = (A, B, \pi)$ to maximise $P(O \mid \lambda)$.

3.4 Convolutional Neural Network

Convolutional Neural Network (CNN) is an emerging and fast-growing branch of deep learning [46], [58]. As introduced at the beginning of this chapter, CNN provides convolutional layers that replace the feature extraction process. Fully connected dense layers provide classification capabilities that further classify the features extracted by convolutional layers into target categories.

An example of the CNN network is shown in Figure 3.5. After receiving input data, a max-pooling layer is applied as the first layer to down-sampling the input data by selecting the maximum value of a given data subset. The second layer takes the output of the first layer as input. The second layer is a convolutional layer, which filters the input data by convolutional kernels and outputs the extracted features. After another max-pooling layer as the third layer, two fully connected dense layers are applied as the fourth and

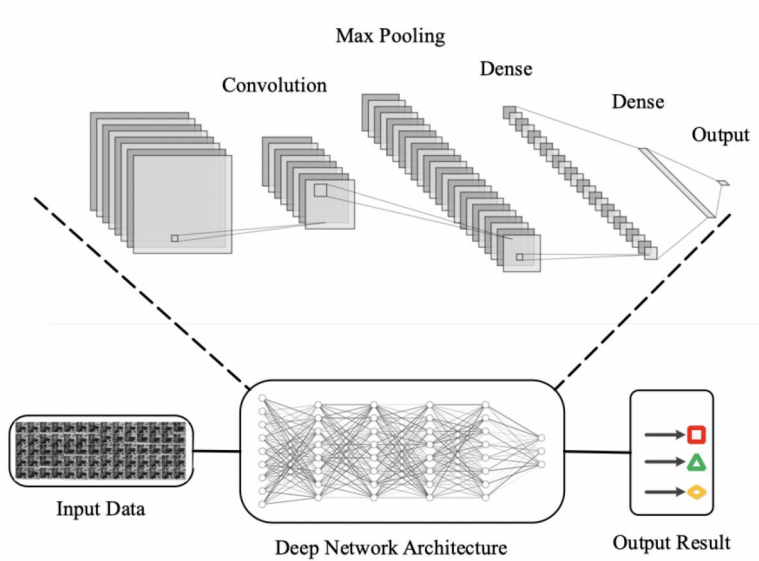


Figure 3.5: Example of Convolutional Neural Networks.

fifth layer of the network. The fully connected dense layers receive and map the extracted features to the output categorical labels. In summary, compared with other models, CNN is a supervised learning model that is capable of achieving acceptable results without careful features design and engineering [46].

3.5 Long Short-Term Memory

Long Short-Term Memory (LSTM) is a special type of Recurrent Neural Networks (RNN) which is suitable for learning long sequential data. Compared with basic RNN, LSTM is capable of learning the long-term dependencies from input data [48]. LSTM provides cell states and gates mechanism. With the input gate, the forget gate and the output gate, LSTM allows recurrent networks to selectively remember or forget information. This design increases the possibilities to remotely link the causes and effects.

As shown in Figure 3.6(a), LSTM model is formed by cells. Within a cell, several different mechanisms are designed to achieve the LSTM functionalities:

- Forget gate:

$$f_t = \sigma(W_f \cdot [h_{t-1}, x_t] + b_f) \quad (3.8)$$

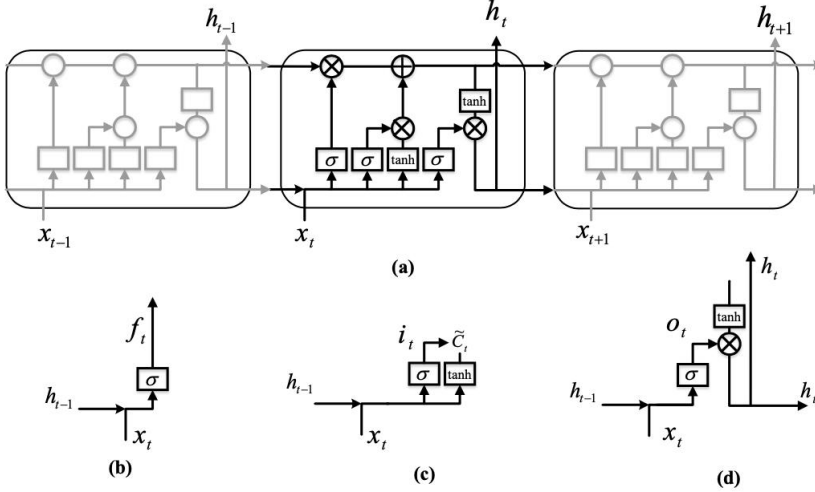


Figure 3.6: Illustration of LSTM model: (a) LSTM model example with three cells. (b) forget gate. (c) input gate. (d) output gate. Adapted from [48]

Forget gate is used for deleting useless knowledge from the previous outputted cell state. In equation 3.8, h_{t-1} is the hidden state from the previous cell. x_t is the input to the system at time t . Forget gate architecture is also shown in Figure 3.6(b).

- Input gate:

$$i_t = \sigma(W_i \cdot [h_{t-1}, x_t] + b_i) \quad (3.9)$$

$$\tilde{C}_t = \tanh(W_c \cdot [h_{t-1}, x_t] + b_c) \quad (3.10)$$

Input gate is used for adding useful knowledge into the current cell state. Equation 3.9 outputs i_t that applies a sigmoid function to regulate the values to be added to the current cell state. Equation 3.10 provides all possible values that can be added to the current cell state by a \tanh function. Input gate architecture is also shown in Figure 3.6(c)

- Output gate:

$$o_t = \sigma(W_o \cdot [h_{t-1}, x_t] + b_o) \quad (3.11)$$

$$h_t = o_t * \tanh(C_t) \quad (3.12)$$

Output gate is used for generating output that selects from useful information in the current cell state. o_t is generated by equation 3.11

3.5. LONG SHORT-TERM MEMORY

that utilises the previous hidden state h_{t-1} and the current input x_t . Then, an output is generated by equation 3.12 which multiplies a regulatory with o_t . Output gate architecture is also shown in Figure 3.6(d).

Finally, the new cell state is calculated by:

$$C_t = f_t * C_{t-1} + i_t * \tilde{C}_t \quad (3.13)$$

where C_t is the new cell state, C_{t-1} is the old cell state, f_t is the output from forget gate, i_t and \tilde{C}_t are the output from input gate.

Chapter 4

HRC context recognition

In this chapter, the author will explain the HRC context, and the related recognition method. This chapter forms the response to RQ1.

4.1 Assembly context recognition

As mentioned in Figure 1.1, context recognition is one of the building blocks of future HRC assembly systems. The environments that human operators and robots work together provides a special co-working context. The context can include different information such as: assembly parts, assembly tools, human motions, and assembly sequences, etc. In a teamwork environment, a human operator can perceive all the mentioned context information easily by visual observation, while robots could not understand any of the mentioned context information without sensors and recognition algorithms. If the robots can perceive the collaboration context effectively, the robots will be supported with knowledge such as: timely locate the right tools or assembly parts. The overall HRC team efficiency can be improved, and teamwork between human and robot can be smoother.

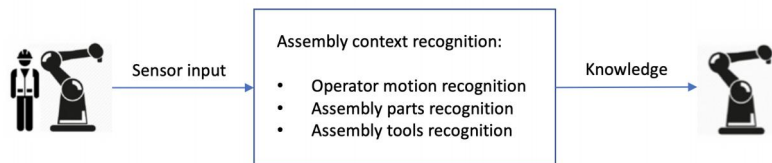


Figure 4.1: HRC context recognition

In this chapter, the author will explore the possibility to effectively recognise HRC context information. As shown in Figure4.1, the input for the as-

sembly context recognition is sensor data. The output of assembly context recognition is knowledge for HRC assembly system. The human operator is, without doubt, the most important part of an HRC assembly system. Even performing the same task, the motions of different human operators can be significantly different. Therefore, the human operator’s motion is one of the most important assembly context, as the human operator’s motion indicates the stages of assembly. Moreover, the different assembly scenarios also introduce different ergonomic challenges such as assembly parts and tools pickup. For the possibility of automatic assembly parts and tools pickup, the assembly parts and tools are also considered as part of the assembly context. In summary, this chapter will explore assembly context recognition of operator motion recognition, assembly parts recognition and assembly tools recognition.

4.2 Transfer learning

Since the sensor applied in this research is a normal video camera, the dataset is in the format of video clips. There are several different sequence recognition models mentioned in the previous chapter that can be deployed to recognise body motions from the video clips. An efficient approach is to process images instead of videos, to slice the video clips into images to down-sample the dataset. The author will eventually need to train two specific neural networks for the two recognition tasks: human motion recognition, and assembly parts and tools recognition.

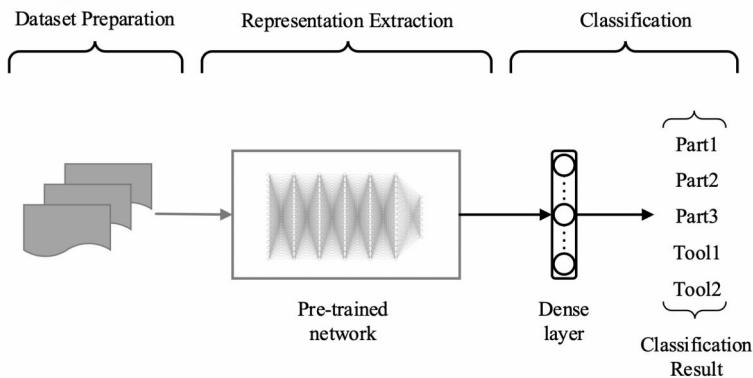


Figure 4.2: Illustration of transfer learning approach for assembly parts and tools recognition

To maximise efficiency, transfer learning approach is applied in this chapter. Transfer learning can reuse knowledge from other models. The training cost on a similar dataset can be greatly reduced. As mentioned, video clips can be sampled into image sequences, it would be possible to utilise transfer learning to recognise body motion, assembly parts and assembly tools from video clips. The basic idea of transfer learning is shown in Figure 4.2.

Transfer learning approach, in general, can be defined by four fundamental elements: a domain \mathcal{I} , a task \mathcal{T} , a learning source S , and a target source T [59], [60]. The domain \mathcal{I} consists of feature space \mathcal{X} and marginal probability distribution $P(\mathbf{X})$. Given a domain $\mathcal{I} = \{\mathcal{X}, P(\mathbf{X})\}$, the task is $\mathcal{T} = \{\mathcal{C}, f(\cdot)\}$, where \mathcal{C} is a label space, and $f(\cdot)$ is a predictive function. The predictive function can be trained from a training dataset $\{(\mathbf{X}_i, c_i)\}_{i=1}^n$, where n is the total number of training samples. The learning effort for the target predictive function $f_T(\cdot)$ applied in the target domain \mathcal{I}_T can be greatly reduced with the utilisation of the knowledge transferred from the source domain \mathcal{I}_S and the source task \mathcal{T}_S .

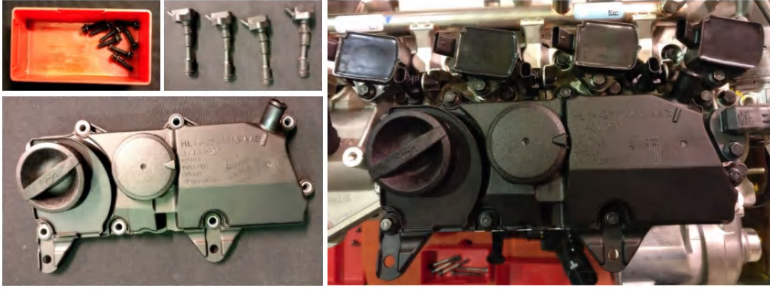


Figure 4.3: Assembly part (left) and car engine after assembly (right)

In the specific case of body motion recognition, assembly parts and tools recognition, the author adapts AlexNet. AlexNet is developed by Alex Krizhevsky and trained with 1.2 million images of 1,000 different categories in the ImageNet dataset [46]. AlexNet has demonstrated superior performance in image-related tasks by greatly improve classification accuracy. The AlexNet $f_A(\cdot)$ is trained from \mathcal{I}_S and \mathcal{T}_S . In both cases, the sequences of pre-sampled images can be sent through the pre-trained AlexNet $f_A(\cdot)$, to extract the knowledge represented. After the feature extraction process, the representation of the training dataset can be denoted as $r^{B_1}(X_i^B; \theta^{B_1})$ where θ^{B_1} is the parameters transferred from $f_S(\cdot)$ and X_i^B denotes the relevant training samples. Then, two fully connected layers are added to further train the target function $f_T(\cdot)$. To minimize overfitting, a 'Dropout' function is employed. The 'Dropout' function can randomly drop neurons

out of the neural network. After the 'Dropout' function, the network is further classified by minimising the cross-entropy loss, before being fed into a 'Softmax' layer for normalisation [55]:

$$\min_{W^f, \theta^{B_1}} \sum_{i=1}^n \mathcal{L} \left(\text{softmax} (W^f r^{B_1} (X_i^B; \theta^{B_1})), c_i \right) \quad (4.1)$$

where W^f is the weights, c_i is the labels. Detailed expiation of AlexNet can be found in Paper 3.

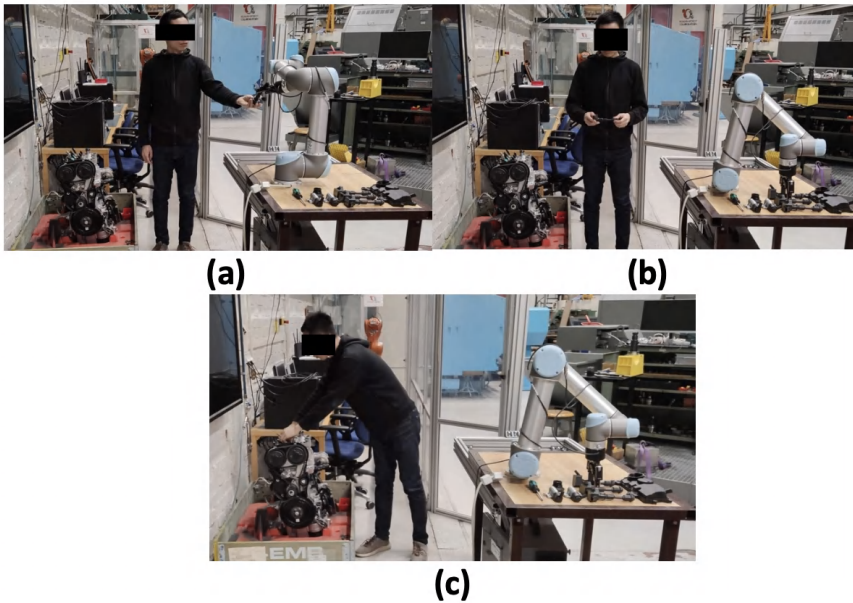


Figure 4.4: Example of different assembly motions. (a)grasping. (b)holding. (c)assembling.

4.3 Experiment

To evaluate the proposed context recognition approach, a case study using a car engine is designed. A large plastic part and four small control plugs will be assembled and fastened by screws on a car engine, shown in Figure 4.3. The assembly process was performed by an operator and recorded by a video camera. The video will then be processed and feed to the trained

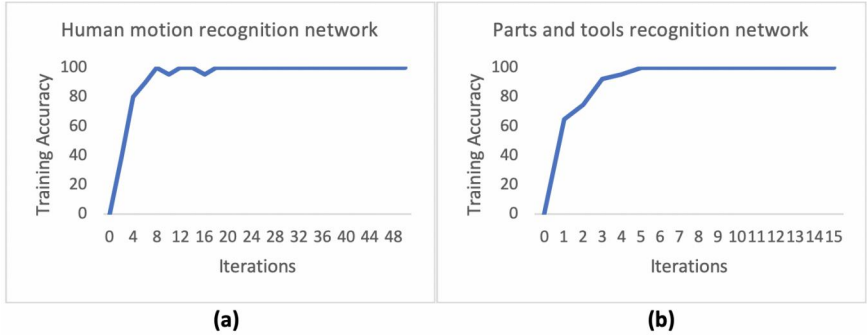


Figure 4.5: Learning curves of the two neural networks

neural networks to recognise the designed assembly motion and assembly parts and tools.

To collect training dataset, images are taken for different motions and assembly parts and tools, an example of the assembly motions are shown in Figure 4.4. There are three categories of assembly motions: grasping (Figure 4.4a), holding (Figure 4.4b), and assembling (Figure 4.4c). The assembly tools and parts have three categories as well: large part, small part and screwdriver. To make sure the dataset provide enough variety, 10 images were taken under different angles and conditions for each part or tool. 10 different images were also taken with different combinations of motions and parts/tools. The neural network trained for parts and tools will be utilised to identify the exact part or tool after the assembly motion recognised.

The author applied 80%-20% split for the images captured, 80% were used for training, 20% for testing. The curves of learning for both human motion network and parts/tools network are shown in Figure 4.5. It can be observed that both of the proposed neural networks reached 100% after only a few iterations. The two neural networks were then used to process video images after the training. In total, there were 4485 frames (25 frames per second, 180s in total) of videos captured, 20% of the frames are sampled for evaluation. The result of operator motion recognition is shown in the top row of Figure 4.6. Within the 897 test frames, 30 frames were misclassified (shown as the peaks in Figure 4.6), leading to a classification accuracy of 96.6%. It is worth to notice that most of the misclassified samples clustered during the transitions stages, which might be the causes for uncertainty. It is further noticed that the motion of grasping is always followed by holding and assembling. There is a typical pattern here in the assembly process. In the bottom row of Figure 4.6, the result of parts and tools recognition is

CHAPTER 4. HRC CONTEXT RECOGNITION

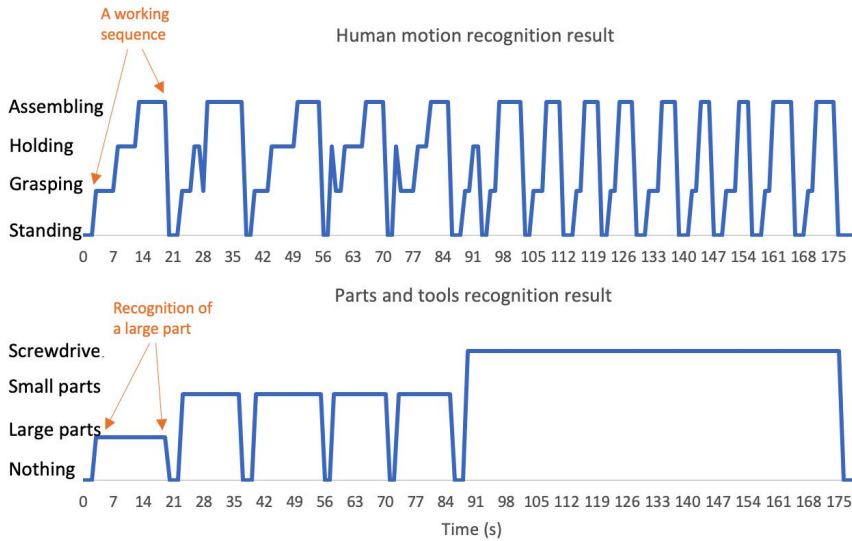


Figure 4.6: Examples of video frames recognition: motion recognition (top); parts and tools recognition (bottom)

shown. It can be observed that large or small parts are normally recognised for 8-15s, when the human operator is also recognised as working with the large part for the entire sequence of grasping, holding and assembling, in around 3-19s. After around 85s, only one holding position was captured during the period of time, when the operator is using the screwdriver.

Chapter 5

Multimodal robot control

In this chapter, the author will explain the improvement of accuracy for multimodal robot control. This chapter provides the response to RQ2.

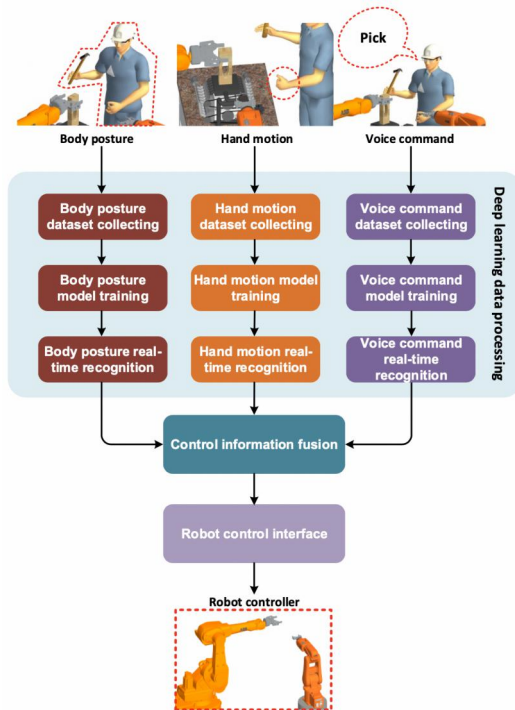


Figure 5.1: Example flowchart illustration of multimodal HRC

5.1 Accuracy limitation

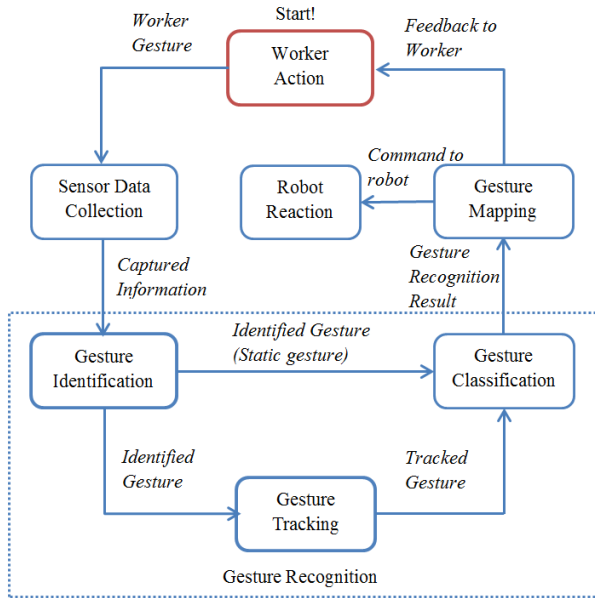


Figure 5.2: Gesture recognition for HRC

Normally, human operators can use control codes to command an industrial robot with a programming-based user interface. However, future Industry 4.0 and HRC workstation requires a more flexible and intuitive way to control an industrial robot. A recent development in this direction is multimodal HRC: multimodal control interface towards intuitive control of an industrial robot. Multimodal robot control can provide human operators with intuitive ways to control the industrial robot. Thus, the industrial robot can dynamically adapt its task plan to collaborate with human operators in the same collaborative environment. The basic idea of multimodal robot control is illustrated in Figure 5.1. An example of multimodal HRC: the process flow of gesture recognition for HRC is shown in Figure 5.2

However, the quality of sensors for multimodal robot control still can not satisfy the industrial standards, and the recognition accuracy of the algorithms are also limited [39], [61]. Hence, multimodal HRC systems are still in the laboratory stage and cannot be directly adapted in the manufacturing industry. A simple solution is to add more modalities into the system and use the combined results to increase accuracy. However, information fusion

is a challenge. If the information is well fused from different modalities' sensors, the accuracy of the system can be further improved.

5.2 Multimodal fusion

In this section, the author will provide a solution to solve the limitations. As introduced in previous chapters, deep learning algorithms have demonstrated great potential to improve accuracy in pattern recognition related tasks. Recent deep learning research also demonstrated great potential in multimodal fusion [62]–[65]. If the recognition algorithm for individual modalities is also embedded in neural networks, the multimodal fusion then can further identify the hidden patterns within the high dimensional multimodal dataset. The result will be more accurate than the result of any single modalities. Thus, comprehensive decisions for accurate multimodal HRC can be possible.

The multimodal HRC problem can be formulated as a multi-class classification problem. \mathcal{D} represents the dataset collected from sensors. The dataset provide m samples from the feature space and class label space $\mathcal{X} \times \mathcal{C}$, represented by $\mathcal{D} = \{(\mathbf{X}_i, c_i)\}_{i=1}^m$. \mathbf{X} is the feature vectors where $\mathbf{X}_i = \{x_1, x_2, \dots, x_m\}$ and c is the categorical labels with k different categories. \mathbf{X}_i is connected to a certain categorical label c_i according to the underlying function $f : \mathbf{X}_i \rightarrow c_i$.

The goal is to find a classifier g from the hypothesis space \mathcal{H} that can approximate function f . The error \mathcal{J} can be measured by repeatedly applying the classifier g on the dataset \mathcal{D} . Thus, the optimisation problem can be formulated:

$$\min_{g \in \mathcal{H}} \mathcal{J}_{\mathcal{D}}(g) \quad (5.1)$$

where

$$\mathcal{J}_{\mathcal{D}}(g) = \frac{1}{m} \sum_{i=1}^m \mathbb{I}\{g(\mathbf{X}_i) \neq c_i\} \quad (5.2)$$

where \mathbb{I} is characteristic function.

The brief process of multimodal fusion is illustrated in Figure 5.3. The process starts with corresponding unimodal datasets collected from sensors, following unimodal representation extraction, at last, multimodal representation concatenation and classification. The three unimodal representations are: $r^S(X_i^S; \theta^S)$, $r^H(X_i^H; \theta^H)$, and $r^{B_2}(X_i^B; \theta^{B_2})$, where X_i^S , X_i^H , and X_i^B represent the three different training samples from different sensors. θ^S , θ^H , and θ^{B_2} are the parameters from three unimodal neural networks. In the concatenation step, the three unimodal neural networks are fused by

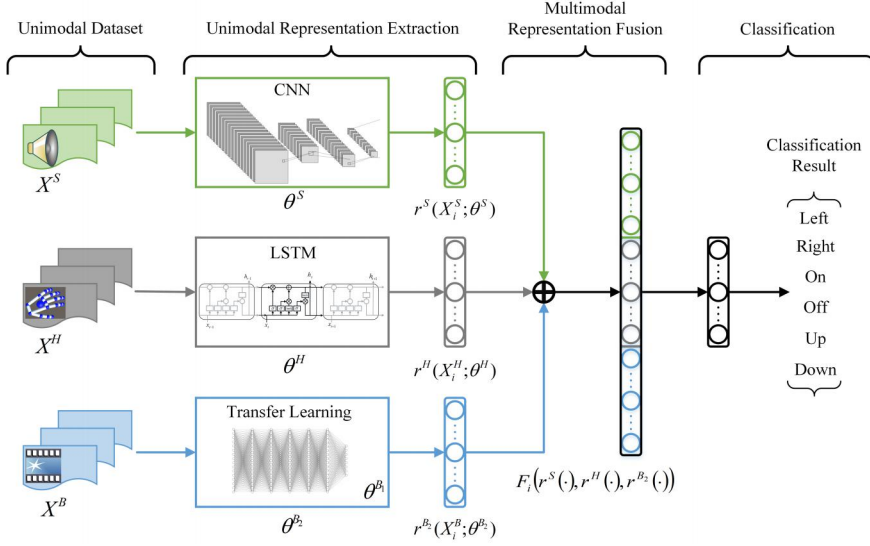


Figure 5.3: Illustration of multimodal fusion

a concatenate function F :

$$\mathcal{G} = F\left(r^S(X_i^S; \theta^S), r^H(X_i^H; \theta^H), r^{B_2}(X_i^{B_2}; \theta^{B_2})\right) \quad (5.3)$$

where \mathcal{G} is the overall representation after multimodal fusion. The fused representation is trained through another layer of fully connected neural networks. Finally, the network is connected to a 'Softmax' function:

$$\min_{W^F, \theta^S, \theta^H, \theta^{B_2}} \sum_{i=1}^n \mathcal{L}\left(\text{softmax}(W^F \mathcal{G}), c_i\right) \quad (5.4)$$

where W^F is the weights. Further details of the multimodal fusion can be found in Paper 4.

5.3 Experiment

In this section, the accuracy improvement for multimodal robot control is implemented and tested.

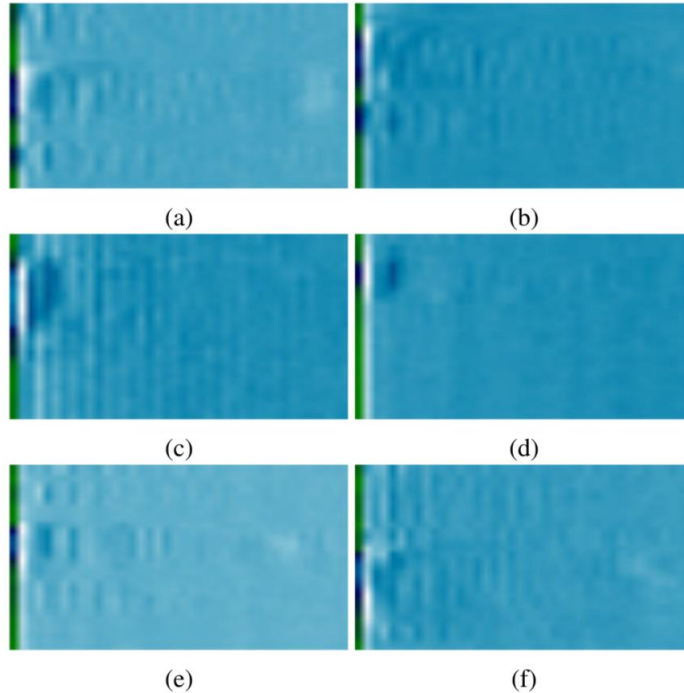


Figure 5.4: Visualised MFCC representation of speech commands. (a) Left. (b) Right (c) On. (d) Off. (e) Up. (f) Down.

5.3.1 Multimodal datasets and pre-processing

To test the proposed multimodal fusion method, the author constructed a multimodal dataset that consist of 3 different modalities: speech command, hand motion recognition, and body motion recognition. For each of the 3 modalities, 6 different label categories are defined: *left*, *right*, *on*, *off*, *up*, and *down*. The visualised speech commands and hand motion commands are shown in Figure 5.4 and Figure 5.5.

- Speech command dataset: the author selects the speech command dataset from a commonly used public speech dataset [66]. The public dataset provides over 65000 audio recordings of more than 30 words. The total number of the selected dataset is 14178. In the audio waveform, the audio data is a 1-D vector of signals. The author preprocessed the audio data into a 2-D matrix of mel-frequency cepstral coefficient (MFCC) vector [67]. Therefore an audio data can be treated

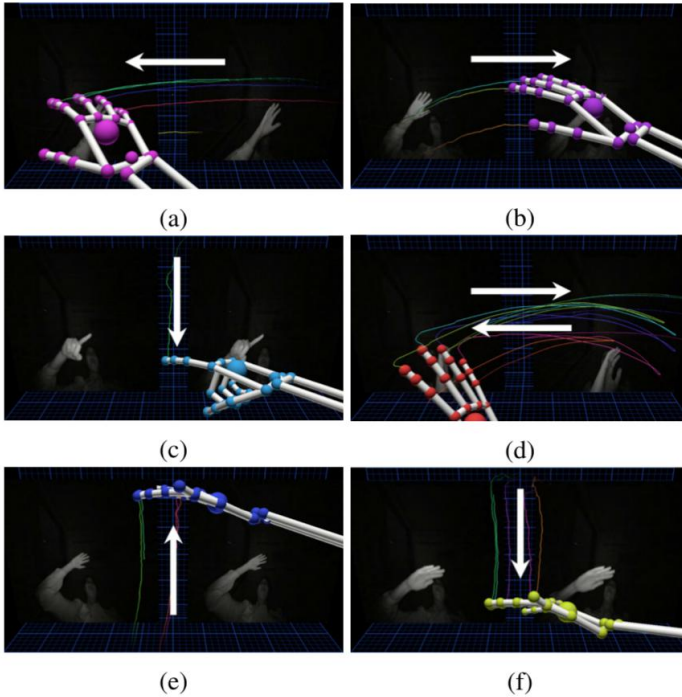


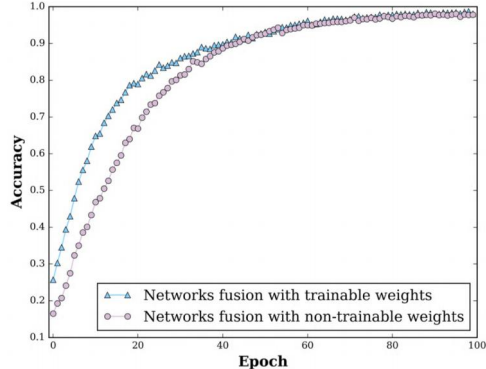
Figure 5.5: Visualised hand motion commands. (a) Left. (b) Right (c) On. (d) Off. (e) Up. (f) Down.

as a single-channel image, as shown in Figure 5.4.

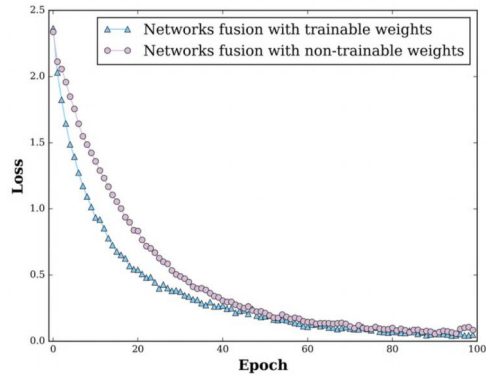
- **Hand motion dataset:** the hand motion dataset is collected by an HRC sensor, Leap Motion Controller [68]. The Leap Motion Controller can track the direction and orientation of hand joints (metacarpal, proximal, intermediate, distal and the tips of five fingers) and bones in 100Hz frequency. In total, 64 hand motion features are collected in each time step. The dataset includes 1183 sequences of hand motions with the six different categorical labels. The collected hand motion data is showed in Figure 5.5.
- **Body motion dataset:** the body motion dataset is collected by normal video cameras. The collected video clips are further processed into sequences of images, which consists of 1379 body motion data with six categorical labels. The sequences of images are further processed by Inception-v3 [69] pre-trained model. The representation after the

process is used as the input data for multimodal fusion.

5.3.2 Multimodal fusion



(a)



(b)

Figure 5.6: The multimodal fusion training process comparison with trainable and non-trainable weights. (a)training accuracy for multimodal fusion. (b)training loss for multimodal fusion.

In the multimodal fusion, the three modalities, speech command, hand motion and body motion are first trained with unimodal neural networks. Representations can be then extracted with the second-last layer of each of the three unimodal neural networks. The representations are further concatenated and fed into another layer of fully connected neural network, resulting in a neural network shown in Figure 5.3.

It is worth to notice that the multimodal fusion process can be trained in two different ways: to mark the weights of the network "trainable" (can change network weights) or "non-trainable" (cannot change network weights). If the network weights are trainable, the weights of the whole network can be changed. If the network weights are non-trainable, the network weights in unimodal models will be fixed and only the weights in the multimodal fusion layer can be updated during the multimodal fusion training process. The training process with trainable and non-trainable weights are shown in Figure 5.6.

5.3.3 Result comparison

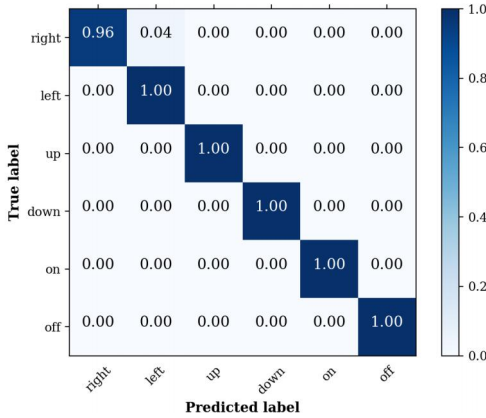


Figure 5.7: Confusion matrix of the multimodal fusion neural networks.

Model	Speech command	Hand motion	Body motion	Fused
Accuracy	93.83%	98.24%	95.95%	99.58%

Table 5.1: Test accuracy of unimodal and multimodal neural networks.

Figure 5.6 presents the multimodal fusion training process with trainable and non-trainable weights. As can be seen from Figure 5.6(a) and Figure 5.6(b), the accuracy and the loss of the multimodal fusion with trainable weights converge much faster than with non-trainable weights. Both neural networks eventually converged to around 98.07% in training accuracy and 0.0496 in loss. As shown in 5.1, the multimodal fusion neural network provides 99.58% accuracy on the test dataset, which is a significant improvement compared with the three unimodal neural networks. The confusion

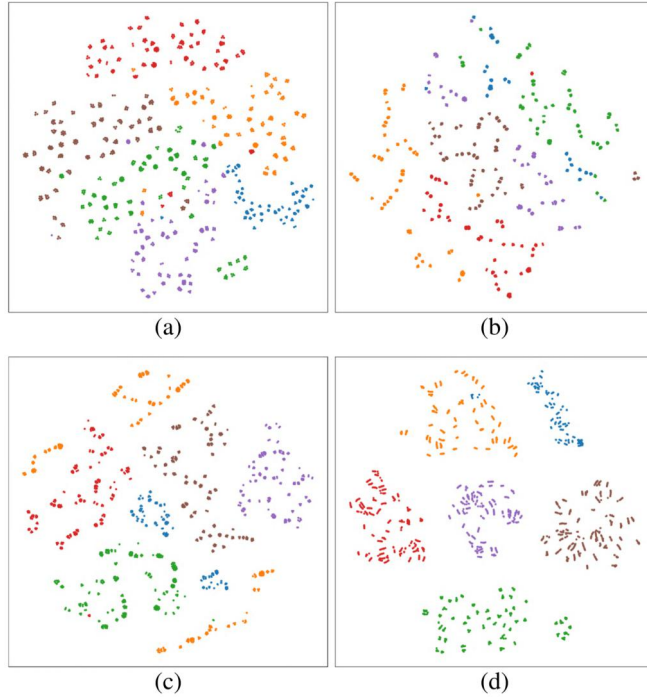


Figure 5.8: t-SNE visualisations of the test dataset after classification, the six different colours represent predicted different labels. (a) speech command model. (b) hand motion model. (c) body motion model. (d) fused model.

matrix of the multimodal fusion neural network is shown in Figure 5.7. The multimodal fusion neural network provides an accuracy of 100% in almost all labels except that in the label *right*. The same misclassification can also be found in Figure 5.8. The quality of test dataset might contribute to the miss-classification. Further analysis can be found in Paper 4 and Section 8.2.

In Figure 5.8, the author visualised the hidden representations of the test datasets after classification by the four neural networks using t-SNE [70] method. The hidden representations can reflect the hidden distributions of the test dataset. The trained neural network can extract the hidden representations if it is well-trained with knowledge. Each plotted test dataset corresponds to its trained neural network. For example, the speech command neural network (Figure 5.8a) is applied to the speech command dataset to extract the hidden representations. In Figure 5.8, each plotted point rep-

CHAPTER 5. MULTIMODAL ROBOT CONTROL

resents a data point from its corresponding test dataset, the six different colours represent six categorical labels. It can be confirmed that the multimodal fusion representations are better separated, compared with the result of the other three neural networks. The multimodal fusion neural network outperforms the three unimodal neural networks.

Chapter 6

Human motion prediction

In this chapter, the author will explain the concept of human motion prediction to improve efficiency in HRC assembly systems. This chapter introduces the response to RQ3.

6.1 Assembly motion prediction

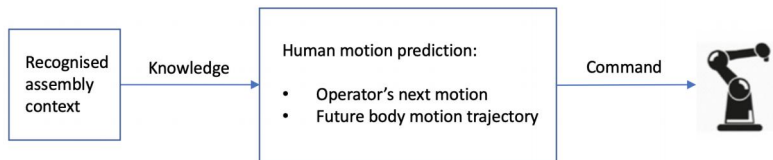


Figure 6.1: Human motion prediction

As illustrated in Figure 1.1, operator motion prediction is an important building block for future HRC assembly systems. The recognised assembly context can be used as the knowledge input for operator motion prediction. The output of operator motion prediction can be a direct command to the robot. In general, HRC assembly systems are more customised and flexible than conventional assembly systems. However, since a human operator's (work-related) motions are repetitive and can be recognised, it is possible to model the human operator's behaviour during assembly, so the operator's assembly motions can be better anticipated and prepared accordingly. The author explored human motion prediction from two directions:

- To model the assembly process as a sequence of discrete human motions, and predict the next motion.

- To forecast the human operator's future motion trajectory for online robot action planning.

6.1.1 Operator's next motion

To recognise a human operator's next motion, existing human motion recognition techniques can be applied. The recognised human motions are further modelled by Hidden Markov model (HMM). The operator's motion transition and observation probability matrices are then generated. Based on the trained HMM model, the operator's motion prediction becomes possible. The predicted motion can be used for assistive robot control.

6.1.2 Future motion trajectory

Utilising the recognition capabilities of machine learning models, the author developed a recurrent neural network (RNN) model for operator motion trajectory prediction. The developed RNN model learns from the interactions data among the human body and provides an estimation for body trajectory prediction. A robust prediction of the operator's future motion trajectory can be utilised to instruct the robot to collaborate with human operator proactively, and further enhance the efficiency and safety in HRC assembly systems.

6.2 Problem formulation and solution

In this section, the problem formulations and solutions for the two different approaches will be introduced.

6.2.1 Operator's next motion

In this subsection, the first approach will be presented: to model the assembly process as a sequence of human motions, and predict the next motion.

Task-level assembly

In assembly, operator instruction sheet (OIS) can provide detailed instructions of the given assembly tasks in the specific assembly station. The task sequence is pre-defined and fixed. However, in an HRC assembly system, the human operator's motion can be flexible. Operators may prefer to perform the same task in a variety of ways. Since human operator's motions can be captured as part of assembly context, it is possible to generalise task-level human motions into a discrete model, as shown in Figure 6.2.

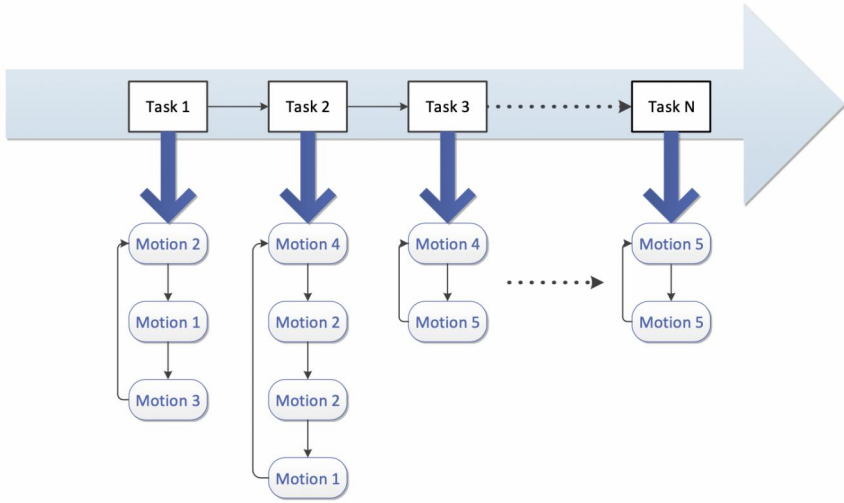


Figure 6.2: Example of task-level representation in an assembly station

Task representation

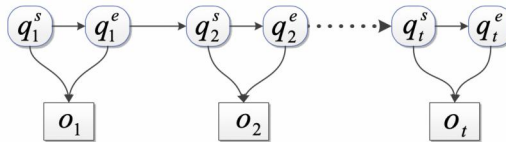


Figure 6.3: An HMM model representation of a human operator's assembly motions

As Figure 6.3 shows, the representation of a human operator's motions can be modelled as a linear sequence. It can be seen as a Markov process that each motion starts after the end of the previous motion. q_t^s is the start of a motion. q_t^e is the end of the motion. The motion o_t is presented between q_t^s and q_t^e . For the same operator, although different q_t^s and q_t^e can be observed, similar sequence of $\{o_1, o_2 \cdots o_t\}$ can be captured. In such way, the robotics system can take the advantage of discretised response. Although robot need to be controlled continuously, the clear boundary of q_t^s and q_t^e still can provide a clear guidance for industrial robot control.

Given the observation of motion o_t , the observation probability can be

described, given the start and finish of the motion:

$$P(o_{s:e}^t | q_t^s, q_t^e) \quad (6.1)$$

The observation probability of motion recognition can be described considering HMM:

$$P_h(o_{s:e}^t | q_{s:e}^t) = \begin{cases} 1 & \text{if detected} \\ 0 & \text{otherwise} \end{cases} \quad (6.2)$$

where $P_h(o_{s:e}^t | q_{s:e}^t)$ represents a reliable motion recognition technologies, that can detect the human motion accurately. If the motion detection technology contains uncertainty, the following equation can be used:

$$P_l(o_{s:e}^t | q_{s:e}^t) = \begin{cases} L_h & \text{if } L_h \leq P(o_{s:e}) \leq 1 \\ L_l & \text{if } L_l < P(o_{s:e}) < L_h \\ 0 & \text{if } 0 \leq P(o_{s:e}) \leq L_l \end{cases} \quad (6.3)$$

where $P_l(o_{s:e}^t | q_{s:e}^t)$ is the probability if a motion is recognised. $P(o_{s:e})$ represents the observation probability. L_h and L_l are parameters that represent the limits of high and low detection probabilities. The parameters can be adjusted based on the different recognition algorithms.

HMM solution

As introduced in Section 3.3, HMM can solve three fundamental problems. For our specific case, the observation sequence $\{o_1, o_2 \cdots o_t\}$ is already known, whereas A and B need to be learned. The prediction of a human worker's motion mainly relies on A . Therefore, the third problem needs to be solved: how to adjust model parameters $\lambda = (A, B, \pi)$ to maximise $P(O | \lambda)$.

To solve the problem, EM (expectation-modification) method [43] can be applied. $\xi_t(i, j)$ is defined as the probability of being in state s_i at time t , and state s_j at time $t + 1$, given the model and the observation sequence:

$$\xi_t(i, j) = P(q_t = s_i, q_{t+1} = s_j | O, \lambda) \quad (6.4)$$

By applying the forward and backward procedures shown in Figure 6.4 [43]. The notation $\xi_t(i, j)$ can be presented as:

$$\xi_t(i, j) = \frac{\alpha_t a_{ij} b_j o_{t+1} \beta_{t+1}(j)}{P(O | \lambda)} = \frac{\alpha_t a_{ij} b_j o_{t+1} \beta_{t+1}(j)}{\sum_{i=1}^N \sum_{j=1}^N \alpha_t a_{ij} b_j o_{t+1} \beta_{t+1}(j)} \quad (6.5)$$

where $\alpha_t(i)$ is the forward variable:

$$\alpha_t(i) = P(o_1, o_2 \cdots o_t, q_t = s_i | \lambda) \quad (6.6)$$

6.2. PROBLEM FORMULATION AND SOLUTION

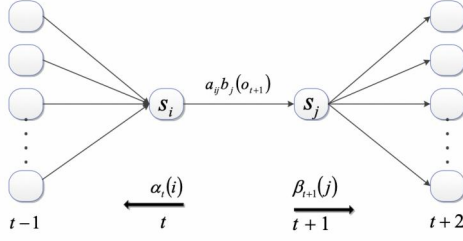


Figure 6.4: The HMM model forward and backward procedure [43]

where $\beta_t(i)$ is the backward variable:

$$\beta_t(i) = P(o_{t+1}, o_{t+2} \cdots o_T | q_t = s_i, \lambda) \quad (6.7)$$

$\gamma_t(i)$ is defined as the probability of being in state s_i at time t , given the observation sequence $\{o_1, o_2 \cdots o_t\}$ and model $\lambda = (A, B, \pi)$. Therefore, we find:

$$\gamma_t(i) = \sum_{j=1}^N \xi_t(i, j) \quad (6.8)$$

$$\bar{a}_{ij} = \frac{\sum_{t=1}^{T-1} \xi_t(i, j)}{\sum_{t=1}^{T-1} \gamma_t(i)} \quad (6.9)$$

$$\bar{b}_j = \frac{\sum_{t=1, s.t. o_t=v_t}^T \gamma_t(j)}{\sum_{t=1}^T \gamma_t(j)} \quad (6.10)$$

given:

$$\sum_{j=1}^N \bar{a}_{ij} = 1, 1 \leq i \leq N \quad (6.11)$$

$$\sum_{k=1}^M \bar{b}_j(k) = 1, 1 \leq j \leq N \quad (6.12)$$

After A and B are given, $\lambda = (A, B, \pi)$ is known as well. Since A indicates the state transition probability distribution, it is possible to predict the state.

6.2.2 Future motion trajectory

In this subsection, the second approach will be presented: to forecast human operator's future motion trajectory for online robot action planning

Human skeleton model

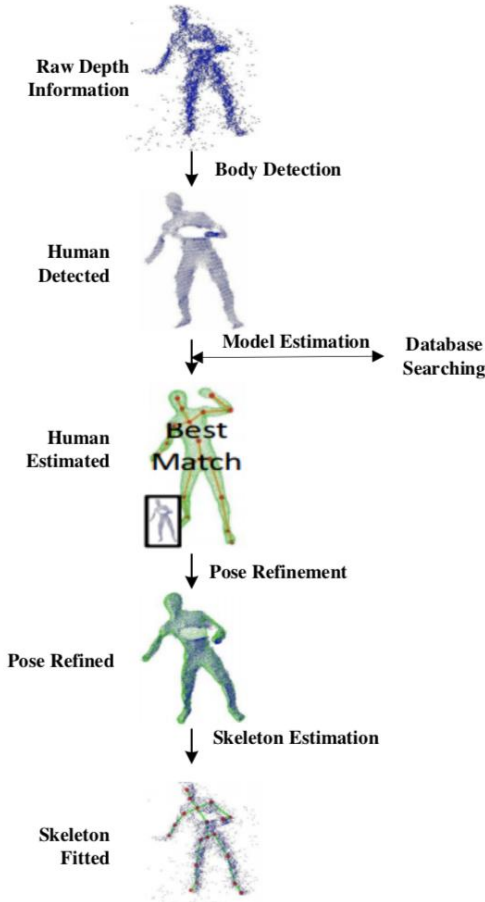


Figure 6.5: Skeleton model estimation

As shown in Figure 6.5, the body skeleton model is a simplified human body model that preserves the most important human body information. The skeleton representation can reduce a huge amount of data compare with point cloud representation, which is used extensively in HRC. The volume information is skipped, while the body joints and bones positions and orientations are well kept.

RNN modelling for HRC

It has been long investigated that recurrent neural network (RNN) and LSTM similar networks can capture motion pattern in sequence dataset [48], [71]. The similar approach can also be applied to HRC. Since the human body parts can be well simplified by skeleton model, the RNN neural network can capture the underlying recurrent patterns among the sequence of motion states.

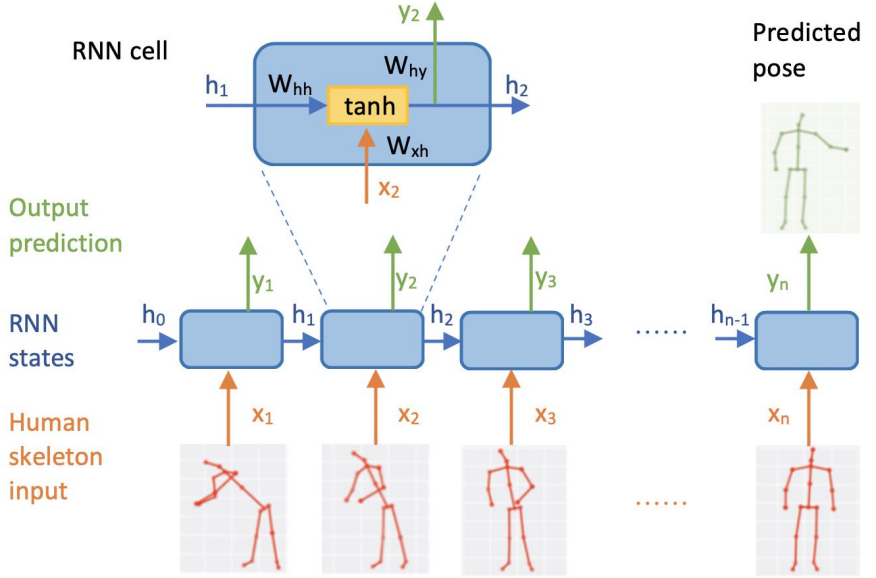


Figure 6.6: RNN for HRC motion prediction

The RNN modelling for HRC can be presented as:

$$h_n = \varphi(W_{hh}h_{n-1} + W_{xh}x_n + b) \quad (6.13)$$

As shown in Figure 6.6 The output pose prediction at time step n is y_n . The associated observation is x_n , the previous state is h_{n-1} , weights controlling the influence of the previous state on the current state is W_{hh} , the weights of the state is W_{xh} , the weights for output is W_{hy} . \tanh is a nonlinear function that can combine the inputs and provide outputs. The prediction of y_n will require all the observations until state x_n . To customise the RNN into HRC motion prediction problem, the author designed a neural network architecture that includes two arms, two legs, one spine and four links between the joints. With such a neural network architecture, the body

motions can be predicted more accurately. Further details of the design can be found in Paper 6.

6.3 Experiment

In this section, experiments are designed to test the implemented human motion prediction systems with the two different approaches.

6.3.1 Operator's next motion

The author designed a car engine assembly case to test the operator's next motion approach.

Car engine assembly task

The author designed a car engine assembly task to demonstrate the potential to predict operator's next move as an HRC application. The parts before assembly are shown in the left part of Figure 4.3. Three are four electric control plugs. All the plugs need to be plugged in the engine as shown in the left part of Figure 4.3 and fastened with a screw. There is also a plastic cover that needs to be placed on top of the engine and fastened with eight screws. The right of Figure 4.3 shows the car engine after assembly.

Experiment result

The author defined five different motions for the task:

1. Take screwdriver
2. Take plastic part (take big part)
3. Take electric control plug (take small part)
4. Take screw
5. Use screwdriver to assembly screw (assembly)

The first four motions: take screwdriver, take plastic part, take electric control plug, take screw, can be directly detected by RFID tags, which is a reliable motion recognition method. The last motion: assembly, can be detected by vision-based motion algorithm, which is a detection method with uncertainty. Hence, the five different states are: $S = \{s_1, s_2, s_3, s_4, s_5\}$. According to Equations 6.2 and 6.3, the six different observation symbols are: $V = \{v_1, v_2, v_3, v_4, v_5, v_6\}$. The states and observation symbols are shown in Table 6.1.

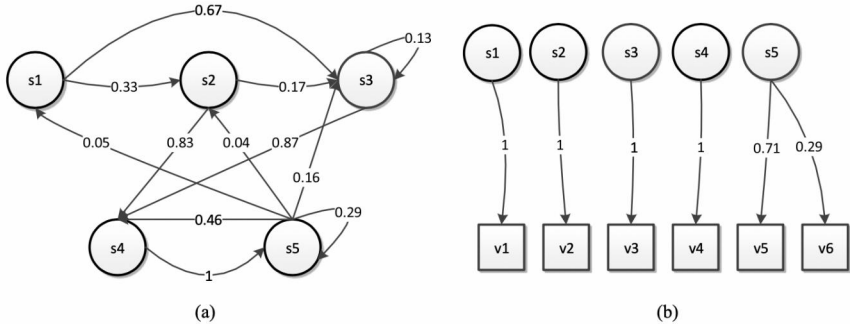


Figure 6.7: HMM state transition and observation probability graph of the assembly case (a) state transition probability matrix graph; (b) state observation probability graph

States	Meaning	Observation	Meaning
s_1	take screwdriver	v_1	observed
s_2	take big part	v_2	observed
s_3	take small part	v_3	observed
s_4	take screw	v_4	observed
s_5	assembly	v_5	observed (high prob)
		v_6	observed (low prob)

Table 6.1: States and observation symbols defined for the assembly task

During the experiment, a human operator is invited to perform the assembly task ten times. The initial motion state is s_1 : take screwdriver. The sequence of the operator assembly is used for HMM model training. The visualisation graph of the trained state transition probability distribution matrix is shown in Figure 6.7 (a). The visualisation graph of the state observation probability graph is shown in Figure 6.7 (b). It can be observed that there are differences between reliable and uncertain motion recognition technologies. It can be seen from the state transition probability that the worker explored many different assembly sequences. State s_2 , s_3 , and s_4 have relatively certain next state, whereas s_1 and s_5 have less uncertain next state. s_5 has many different next states due to the end of a ‘sub-sequence’. It can also be noticed that sometimes v_6 appears after v_5 .

The experiment showcased the possibility of human motion prediction for HRC. The operator explored different assembly sequences, but there are still patterns captured. For instance, since states s_2 , s_3 , and s_4 have comparatively certain next states, it is then possible to control a robot to help

the human operator. s_5 also has quite certain next state. In the experiment, the assembly parts are not considered. The limited number of assembly parts can also be used to improve motion prediction uncertainty. The vision-based motion recognition technology also introduced uncertainty. By improving the motion detection algorithm, it is possible to greatly improve the reliability of the result, hence, the system robustness can be further improved.

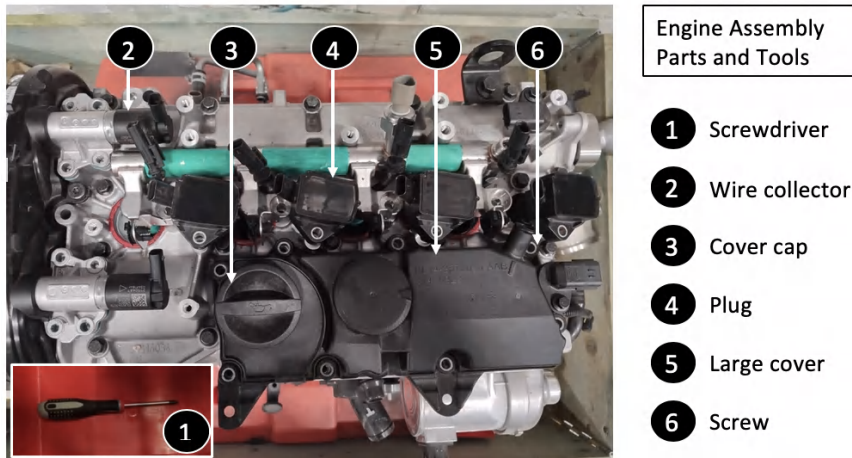


Figure 6.8: Engine assembly setup

6.3.2 Future motion trajectory

To test the future motion trajectory approach, a similar but more complex experiment is proposed in this case study, shown in Figure 6.8. The engine assembly case consists of a plastic part, a cover cap, two wire collectors, four electric control plug, a screwdriver, and eleven screws. The assembly workstation is shown in Figure 6.9. In the assembly workstation, a UR5 robot is installed to the left of the operator, an engine block is placed on the right of the operator, the assembly parts and tools are stored in the colour-coded containers to the left of the UR5 robot. A Kinect sensor is installed facing the assembly workstation. The Kinect sensor can provide the tracking of human skeleton model at 30Hz.

The final goal of the test is to timely predict the human motion trajectory. To achieve the goal, the author defined three poses that serve as the triggering point for the motion trajectory prediction: *handover* (x_1), *standing* (x_2), and *installation* (x_3), as shown in Figure 6.10. The human

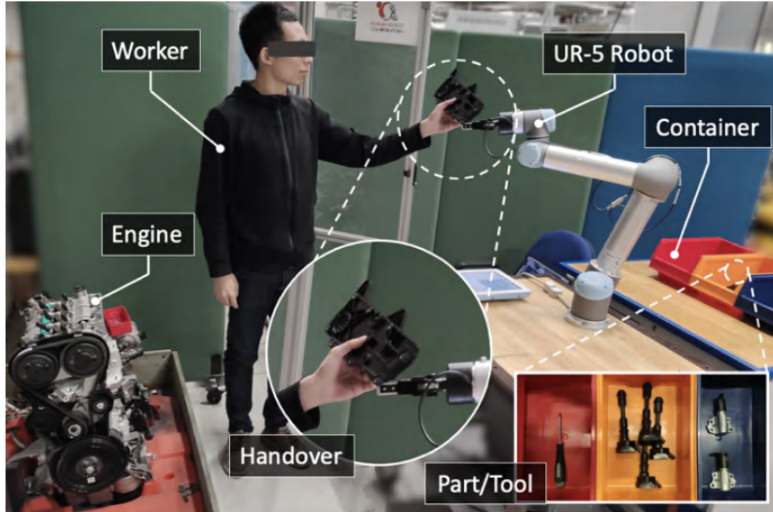


Figure 6.9: Engine assembly workstation

MSE (mm)	MLP baseline	RNN model
Training	27.1	17
Validation	27.0	16.5

Table 6.2: Prediction MSE for MLP baseline and RNN model

operator can start with any of the poses, and the human operator can directly transit from one pose to another following the arrows sequences. For each of the pose, the operator can stay as long as needed.

To collect training dataset, ten different human operator assembly sequences are recorded by the Kinect sensor. The author adapts relative coordinates to record the joint dataset. For instance, a joint (x,y,z) coordinate locations is recorded with respect to its parent joint. The network input length is 30 frames. The final prediction result will include the contribution of all the 30 frames, which guarantees a robust prediction. The size of the training dataset is 8,206 training samples, which is randomly split into 70%-30% training and validation sets. The author adapts mean squared error as the evaluation metrics. The author compared the implemented approach with a simple MLP baseline approach, the result of which is shown in Table 6.2. It can be seen that the MSE is greatly reduced.

An example of parts handover motion trajectory prediction for HRC assembly is shown in Figure 6.11. The yellow line is the predicted end target

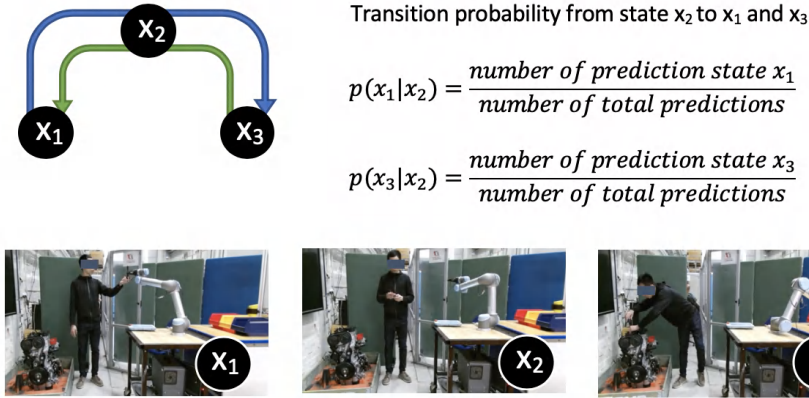


Figure 6.10: Transitions among *handover*, *standing* and *installation*

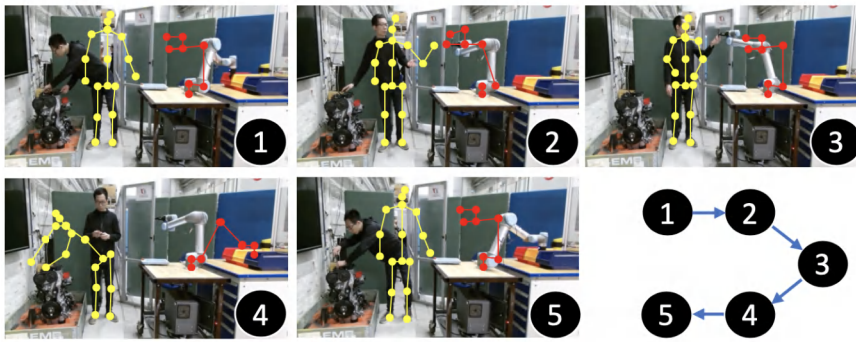


Figure 6.11: Illustrations of future motion trajectory prediction for HRC

location of the operator skeleton. The red line is the corresponding robot target position given the operator motion prediction. In Figure 6.11(1), *installation* pose is detected, therefore, the robot is triggered to pick up an assembly part. The predicted next human operator pose will be *standing*. The corresponding next robot target position is standby. In Figure 6.11(2), the operator starts to move the left arm, the prediction is moving to *handover* pose. Thereafter, the robot will be triggered to move to the predicted *handover* location. With further arm movements, more accurate target locations will be updated. Finally, the robot will be stopped in the final *handover* location and open the gripper, as shown in Figure 6.11(3). After *handover*, the robot will return to the standby position in Figure 6.11(4). With the observation of operator *installation* pose, the robot will pick up

6.3. EXPERIMENT

the next assembly part to prepare for the next *handover*, as shown in Figure 6.11(5). Further experiment analysis can be found in Paper 6 and Section 8.3.

Chapter 7

Remote HRC

In this chapter, the author will explain the concept of remote HRC. This chapter offers the response for RQ4.

7.1 Remote HRC

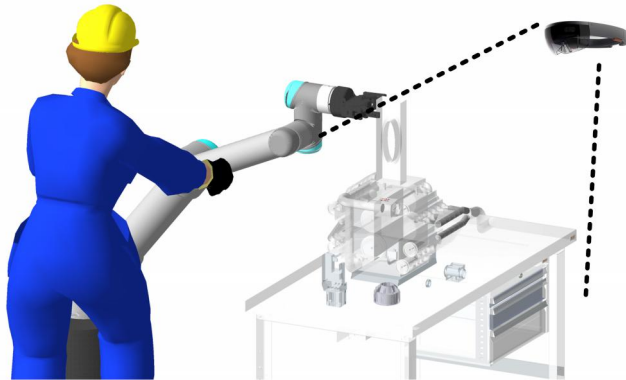


Figure 7.1: Collaborative workstation: a human operator lead-through a collaborative robot with real-time display of the remote assembly parts

The haptic lead-through feature of the collaborative robot is convenient in many different assembly scenarios. With the lead-through feature, human operators can be released from complex robot control codes. However, in a hazard manufacturing environment, where human operators are not allowed to enter, the desired lead-through feature will not be possible to use. As a solution, the author proposed in this chapter, a remote human-robot

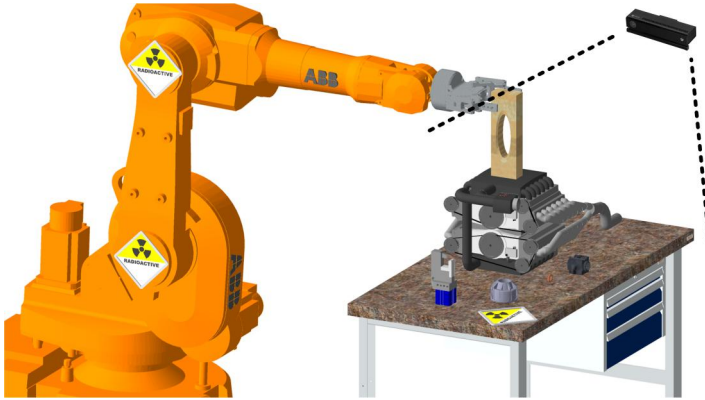


Figure 7.2: Remote workstation: an industrial robot working in dangerous environment with control parameters from the collaborative workstation

collaboration system that closely follows the concept of cyber-physical systems (CPS). The proposed system can provide another possibility for HRC to be able to work in Hazard manufacturing environment.

Figure 7.1 and Figure 7.2 provide an example for the appearance of the proposed remote HRC system. Figure 7.1 shows the collaborative workstation, where an operator is guiding a collaborative robot following the models of assembly part transmitted from the remote workstation. Figure 7.2 shows the remote workstation, where the collaborative robot's end-effector position at the collaborative workstation are transferred to the industrial robot in the remote workstation. The industrial robot is controlled according to the motion of the collaborative workstation. In the remote workstation, there are also sensors to capture the assembly parts. The assembly parts are further recognised and displayed in the collaborative workstation.

7.2 System design

The overall system function design will be introduced in this section. The system is designed to work in four different modes to accommodate different operation scenarios. Following the high-level introduction, the author will further introduce the designed remote robot control system and model-driven display system.

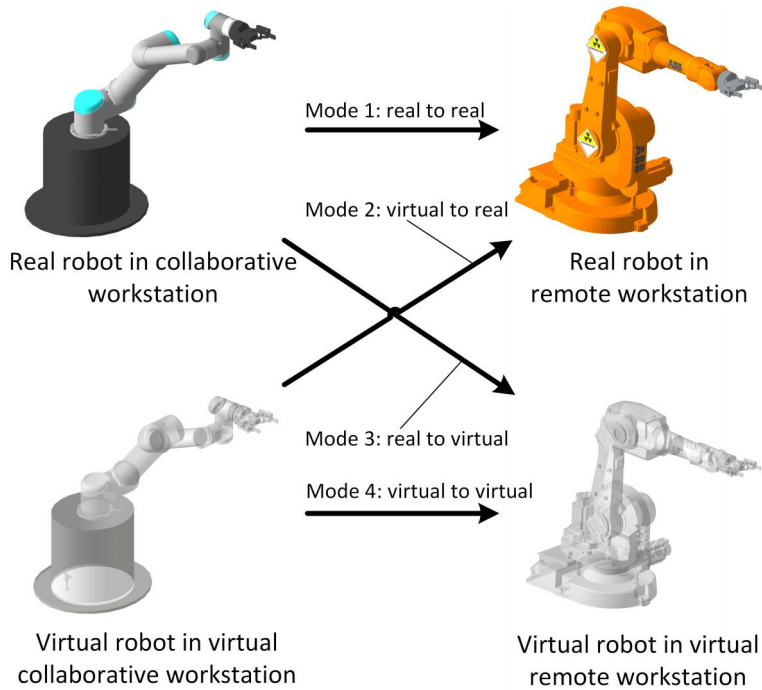


Figure 7.3: Overview of four working modes

7.2.1 Four operational modes

The high-level system design is shown in Figure 7.3. The system is designed to be able to work in four different operational modes for different needs:

1. Real robot controls a real robot:

To be used in the hazard manufacturing environment. In the collaborative workstation, the collaborative robot is guided by a human operator. In the remote workstation, the real industrial robot is controlled with the same motion.
2. Virtual robot controls real robot:

To be utilised in the hazard manufacturing environment where no collaborative robot is available to use. The human operator can manipulate a virtual collaborative robot. The motion will be repeated by the real industrial robot at the remote workstation.
3. Real robot controls virtual robot:

To be used for system testing and virtual commissioning, where the remote workstation does not need to be accessed. The real collaborative robot can be guided by a human operator, a virtual industrial robot can be controlled. Any wrong command from the collaborative workstation will not result in damage in the remote workstation.

4. Virtual robot controls virtual robot:

To be adapted for operator training, where no real robot is required. In this mode, everything is virtual representation. The human operator can manipulate a virtual representation of a collaborative robot to control a virtual industrial robot.

7.2.2 Remote robot control system

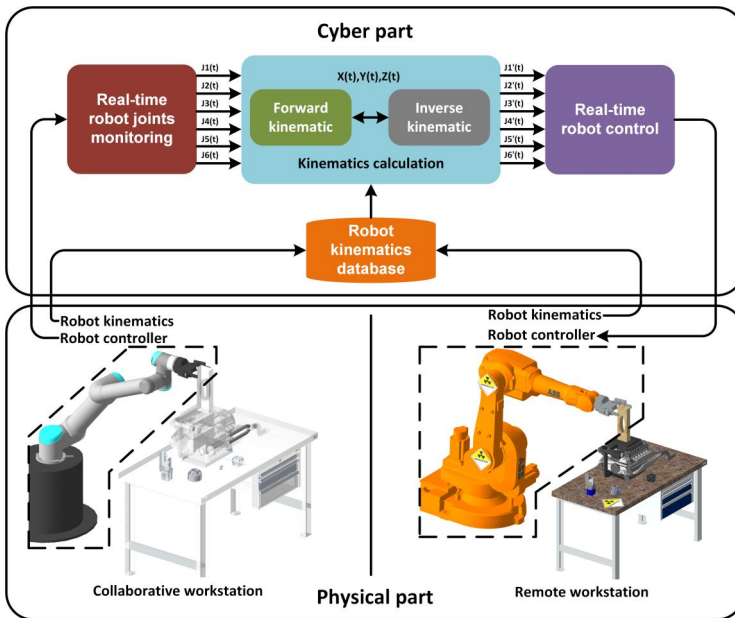


Figure 7.4: Remote robot control system

The centre of remote HRC is the remote robot control system. The remote robot control system needs to make sure that the exact joints positions are reflected in real-time on the industrial robot in the remote workstation. As shown in Figure 7.4, closely following the concept of CPS, the remote robot control system includes a cyber part and a physical part. The

physical part is shown at the lower part of Figure 7.4, surrounded by the broken lines. The cyber part is shown in the upper part of the Figure 7.4. The real-time robot joints monitoring module is always connected with the robot controller of the collaborative robot, where the robot joints positions are monitored in real-time. All the robot kinematics are stored in the robot kinematics database. The kinematics calculation module can calculate the forward kinematics and inverse kinematics, to provide the right command to the real-time robot command module, where the remote industrial robot is controlled.

The authors also designed a virtual haptic effect to avoid collision with objects in the remote workstation. If the sensor detects collision objects in the path of the robot in remote workstation, the robot will be commanded to stop, and respectively, the collaborative robot at the collaborative workstation will also be stopped. The human operator will also be notified and change the lead-through operation accordingly.

The proposed remote robot control system also needs to guarantee the same robot motions between the collaborative robot and the remote robot during operator lead-through. There might be kinematic differences between the two robots, hence, the proposed interface needs to calculate the forward and inverse kinematics to adapt to the different robots. Detailed method and approach can be found in Paper 5.

7.2.3 Model-driven display system

Another important part of the proposed remote HRC system is the model-driven display system. The model-driven display system can provide timely cognitive feedback to the human operator. Since the operator and the remote workstation are in different locations, the assembly environment of the remote workstation should be made transparent for the operator. Model display system and object recognition algorithm is the key component in a model-driven display system.

Figure 7.5 is the design of the model-driven display system, which also includes a cyber part and a physical part. In the physical part of the system (shown in the lower part of Figure 7.5), the remote workstation can provide video stream for the cyber part of the system. In the cyber part of the system (shown in the upper part of Figure 7.5), the assembly objects can be recognised according to the assembly models database and object recognition algorithm. Thereafter, the recognised parts can be further translated, rotated, and registered in the right positions. Finally, the relevant virtual models are displayed for the operator. The detailed algorithm for object recognition can be found in Paper 5.

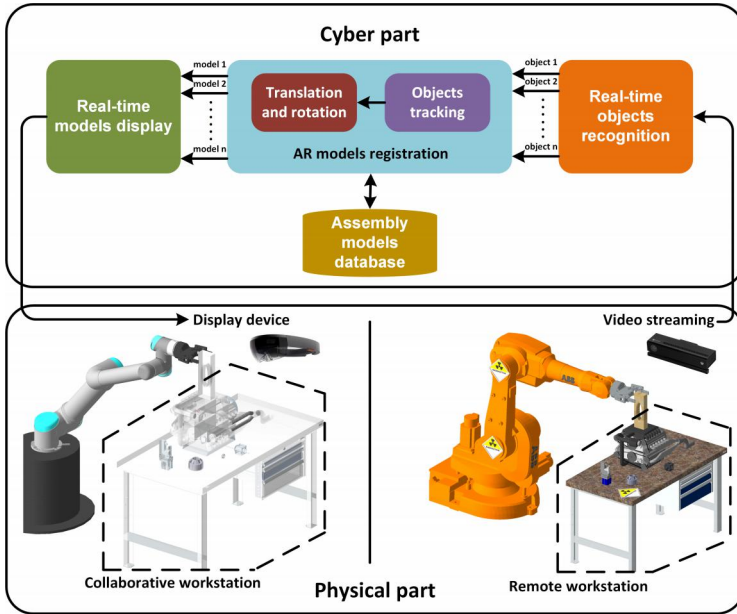


Figure 7.5: Model-driven display system

7.3 Implementation

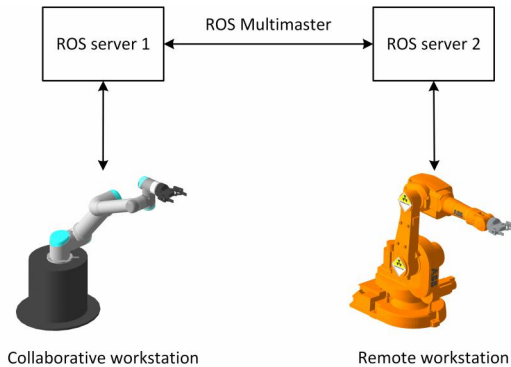


Figure 7.6: Simplified system overview

Following the high-level system design, the detailed implementation of the remote HRC final system (shown in Figure 7.6) is explained in this

section. The final implementation is inspired by the concept of CPS and adapted the Robot Operating System (ROS) framework [72] and the ROS Multimaster software package [73].

7.3.1 ROS Multimaster Package

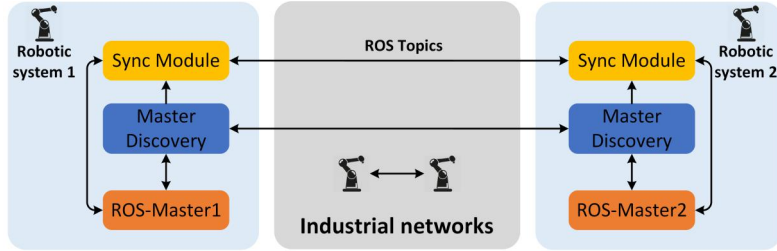


Figure 7.7: ROS Multimaster package

There are two different robots in the designed remote HRC system. However, normally, each ROS system can only provide control towards one robot. To accommodate the needs, the author adopts the ROS Multimaster package [73]. Multimaster is an open-source software package that allows ROS topics exchange between multiple ROS-based robotic systems, as shown in Figure 7.7. The Multimaster package provides: firstly, master discovery nodes that can detect other master discovery nodes. Secondly, synchronise modules that can provide connectivity with local nodes and exchange data. In the following section, the implemented system is presented.

7.3.2 Final System

The overall system is shown in Figure 7.8. The system is an implementation of Mode 1: a real robot controls a real robot. There is also a cyber environment and a physical environment included in the final system.

In the physical environment, there are two workstations: remote workstation and collaborative workstation (upper corner and lower corner of the Figure 7.8). The collaborative workstation provides a collaborative robot and display device for the human operator. The remote workstation provides an industrial robot and the related working environment.

Two different servers support the robotic systems in the cyber part of the system: a collaborative ROS server and a remote ROS server. The two servers are connected by industrial networks. The servers are both supported by ROS and its related systems to monitor and control the robot. In the collaborative ROS server, relevant robot monitoring modes can provide

CHAPTER 7. REMOTE HRC

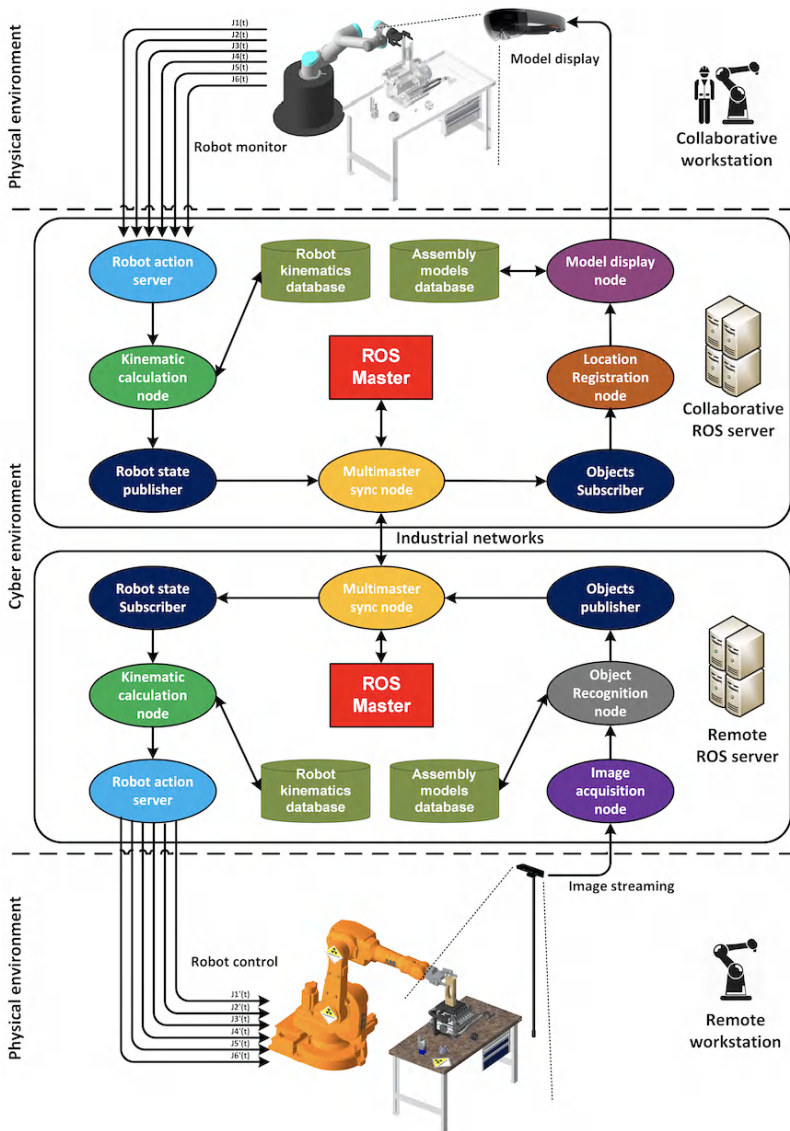


Figure 7.8: Overview of the final system

real-time joint values of the robot in the collaborative workstation. The kinematic calculation node can convert the end-effector positions from joint space into Cartesian space. The end-effector positions are therefore sent to

the remote ROS server for further calculations. Finally, a similar kinematic calculation node in the remote ROS server will translate the joint commands and send to the robot at the remote workstation. In the remote ROS server, there are also ROS nodes that can recognise the assembly parts. The recognised assembly parts will be further sent to the collaborative ROS server. With the right database and object location registration algorithm, the assembly parts can be displayed at the right locations for human operators via display system.

7.4 Experiment

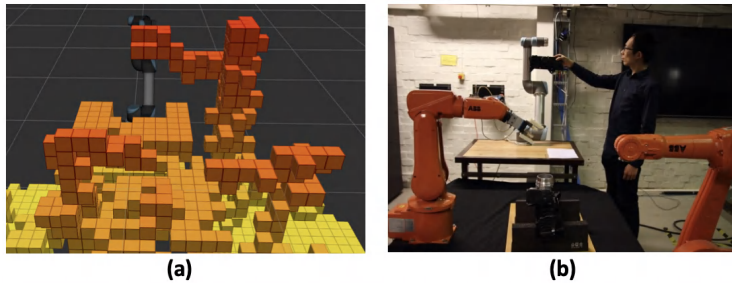


Figure 7.9: Implemented system (a)virtual representation of the environment (b)photo of the physical environment

In this section, the author implemented the proposed system, shown in Figure 7.9. Two test cases are designed to demonstrate the implemented final system:

1. Mode 4: a virtual robot controls a virtual robot
2. Mode 1: a real robot controls a real robot.

7.4.1 Test case Mode 4

The first test case is a scenario of Mode 4: a virtual robot controls a virtual robot. The full system is implemented and connected virtually in the cyber environment. As shown in Figure 7.10, a UR5 robot is simulated in the virtual collaborative workstation. An interactive handler is designed at the end-effector of the UR5 robot for haptic lead-through. In the lower part of Figure 7.10, another UR5 robot is simulated in the virtual remote workstation. Five different end-effector locations are designed as the target of the test. The human operator is expected to lead-through the virtual robot

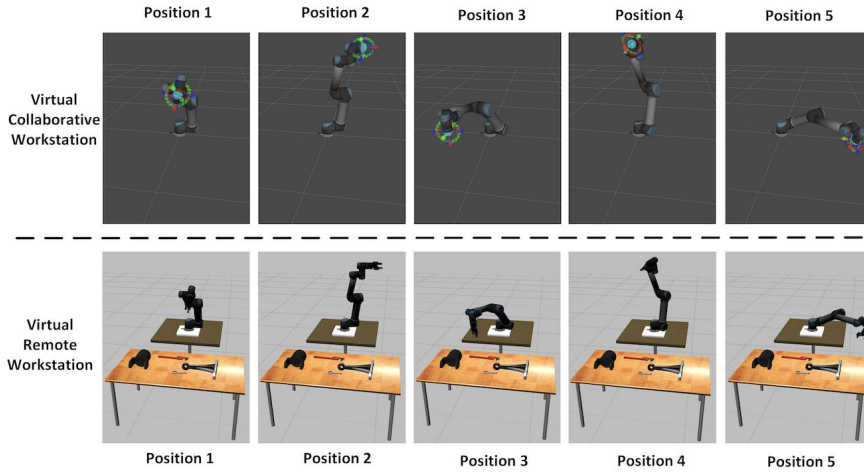


Figure 7.10: Screenshots of the test result of Mode 4

in the collaborative workstation into the desired positions. The response time result is shown in Figure 7.12, the screenshots of the test are shown in Figure 7.10.

7.4.2 Test case Mode 1

The second test case is Mode 1: a real robot controls a real robot, the same architecture explained in Figure 7.8. As shown in Figure 7.11, a UR5 robot is installed in the collaborative workstation, an ABB IRB1600 robot is installed in the remote workstation. The designed test case simulates a production task, where the human operator is expected to lead-through the UR5 robot into four desired positions, ABB IRB1600 robot should follow the motions of the UR5 robot. The response time result is shown in Figure 7.12, the photos of the test are shown in Figure 7.11.

7.4.3 Response Time Comparison

In this section, the author implemented all 4 modes and designed an experiment to compare the response times. The author repeats 60 tests for each of the modes. During the experiment, a human operator is working in the collaborative workstation. The human operator is expected to lead-through the collaborative robot to a random location. The process is repeated for multiple times and the response time is calculated. The time is counted when the new end-effector target sent from the collaborative ROS server

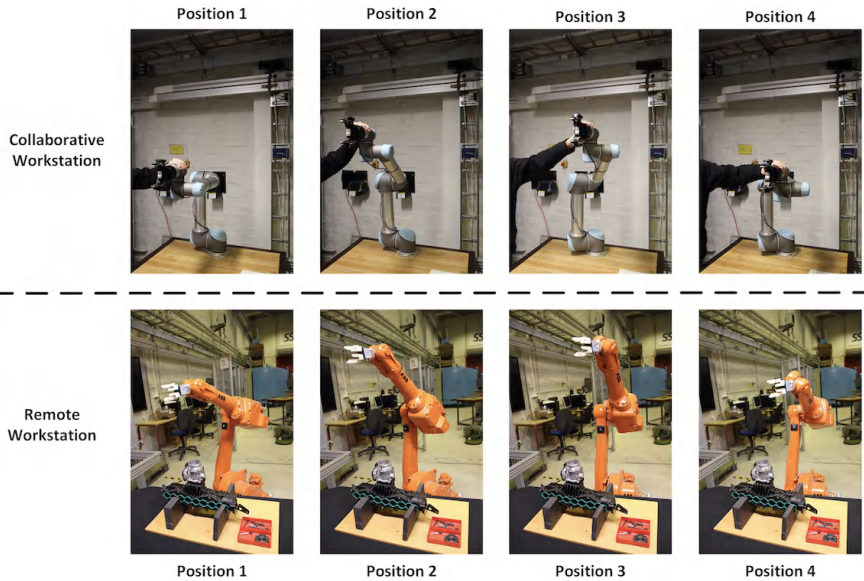


Figure 7.11: Test result of Mode 1

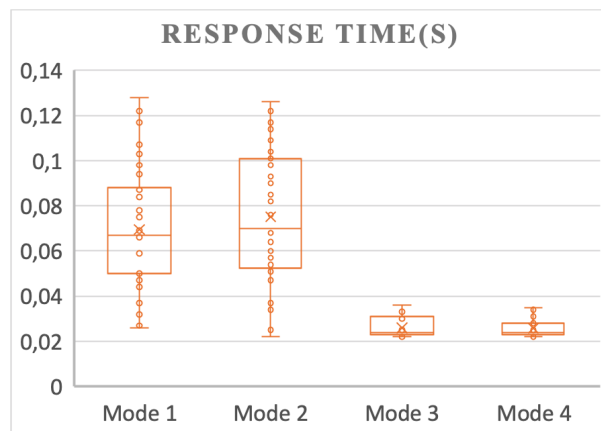


Figure 7.12: Response time comparison of experiment on Modes 1, 2, 3 and 4

until the collaborative ROS server receives the reply that the other robot started to move.

As shown in Figure 7.12, Modes 3 and 4 provide average response time

CHAPTER 7. REMOTE HRC

of 0.025 seconds. Modes 1 and 2 provide average response time 0.067-0.070 seconds. It is obvious that Modes 3 and 4 responded quicker than Modes 1 and 2. The response time of Modes 3 and 4 also has a smaller standard deviation than that of Modes 1 and 2. The reason for the differences can be the different control mechanism. Modes 3 and 4 have the remote workstation in the virtual environment, where a virtual robot can be controlled instantly. However, in the case of Modes 1 and 2, the remote workstation has a physical robot. The control parameters need to be sent to the real robot controller to activate the robot. Delay due to mechanical control is expected. Further analysis can be found in Paper 5 and Section 8.4.

Chapter 8

Discussion and future works

In this chapter, the author will discuss the proposed research questions, with remarks, self-critiques, and possible future directions.

8.1 Discussion on research question 1

RQ1: what is HRC context and how to perceive HRC context effectively?

The first research question is about assembly context, and the effective recognition of assembly context.

Firstly, as can be seen from the thesis, in HRC assembly, no systematic guidance and methodology are available yet. Context awareness can potentially be used as a methodology and guidance for the recognition tasks in HRC assembly, if appropriate development and adjustment for different manufacturing scenarios can be introduced in the future.

Secondly, the definition of assembly context can be challenging and ever-evolving. Of course, it would be a perfect scenario to include as much information as possible. In reality, it is only possible to include the most critical information as the assembly context, for instance, operator-related and assembly efficiency-related information [11]. In this thesis, the author decided to explore operator motion, assembly parts and tools as the assembly context. However, in real assembly lines, different industries and different assembly lines might have very different assembly context. For instance, in an automobile engine assembly line, the quality of an assembled engine can be very essential assembly context, as the assembly quality checking (normally done by human operators) is a key step. Whereas in other assembly lines where the automation level is much lower, assembly sequence is more important, since more human operators are working at

the same assembly station, the coordination between human operators can be critical. As a contrast, in a single operator assembly station, the assembly parts and tools recognition becomes more important, as the robots are expected to help the human operators in parts and tools handling.

Thirdly, regarding the recognition algorithm for assembly context. Due to the computational expenses of object recognition and current computational power development trend, it might be possible that in the future, end-to-end (direct large dataset from the industrial environment, a specific recognition algorithm trained for the specific application) machine learning training approach can be possible. However, the dataset collected must be significantly large to be able to support a meaningful and representative recognition algorithm. In contrast, the author adapted transfer learning to simplify the task. Computational expenses are reduced, and the task of dataset collection can be simplified as well. Due to the cost advantages during model training, transfer learning approach will continue to be a preferred approach in the near future. In different applications, there are different task-specific transfer learning algorithms. For instance, in human pose recognition, there is a pre-trained model PoseNet [74] and in sequence recognition, there is a pre-trained model BERT [75].

Lastly, the future research direction can be more industrial-level assembly context usability study. Another future direction can be the industrial-level assembly context requirements specification and validation.

8.2 Discussion on research question 2

RQ2: how to increase the accuracy of multimodal robot control?

The second research question discusses the improvement of accuracy for the recognition algorithm in multimodal robot control.

Multimodal robot control, recently, started to be tested in different HRC applications in industry. The potential to release human operators from the robot control code preparation is already a clear trend in the industry.

Firstly, there are actually different approaches towards multimodal fusion. The multimodal fusion can be done in different network layers. The result can be extremely different in terms of accuracy. Unfortunately, the reason for the differences is still unclear according to the best knowledge of the research community. In the thesis, the fusion at the second-last layer will result in an efficient and accurate model. It can be easily identified that the fused model outperformed any model trained with single modality dataset. With further analysis of Figure 5.7 and Figure 5.8(d), it can be

discovered that the 4% wrongly classified *right* data points can be found at the top middle part of Figure 5.8(d). The misclassification might be result of data quality issues in testset. The author can provide a more fundamental thought on the reason for the accuracy improvement. It might be the case that compared with the unimodal dataset, neural networks can learn richer knowledge representations from the multimodal dataset. Hence, a more accurate inference can be made considering all the possible embedded representations collectively. If we consider the result of the thesis in the application level, it suggests that the sensor instability is possible to be compensated by adding more sensors or data modalities. Although current sensors are not robust enough for industrial applications, there are possibilities to build a robust HRC assembly system with current sensors and the multimodal fusion algorithm.

Secondly, regarding the fusion process with trainable and non-trainable weights. The neural network accuracy improves faster with trainable weights, as can be observed in Figure 5.6. The trainable mode allows the unimodal models' parameters to change during the fusion training process. However, the two modes eventually converged to the same accuracy level, since the fusion model also provided a sufficiently large search space.

Finally, there is the potential that multimodal fusion approach to be adapted in industry. As discussed in the above sections, safety and reliability are the highest priorities in the manufacturing industries. The multimodal fusion method can serve as an algorithm to improve the data captured from sensors, and provide a robust multimodal HRC assembly system. Towards the goal, a run-time system for real-time multimodal inference is still needed. The run-time system can be developed as part of the future works.

8.3 Discussion on research question 3

RQ3: how to predict human operators' motion to further improve the efficiency of assembly?

The third research question discusses the possibility to predict the human operator's motion during assembly as a measure to improve assembly efficiency.

In previous research of HRC assembly, there is always a gap between the recognised human motions and the robot action control. The human motion perdition research in the thesis is aiming to fill the gap and close the loop of HRC assembly.

Firstly, from the experiment of the first approach, it is possible to con-

CHAPTER 8. DISCUSSION AND FUTURE WORKS

clude: there is a pattern to predict a human operator's motion. Some of the motions can provide quite certain clues about the operator's next motion. Therefore, if similar motion is detected, it is possible to control the robot to facilitate the human operator's next motion. Some other motions cannot generate a certain next move. However, the transition probabilities still can be mapped according to the observed examples. In such cases, the robot can be prepared and anticipate the human operator's most likely next motion. Also, several other sources of information can be used to increase the prediction accuracy of human motion prediction, for instance: assembly sequences and left-over assembly parts. In the study, different kinds of motion recognition methods are also considered, whereas, in industrial applications, the sensor and algorithm robustness problem can be solved by improved sensor reliability and algorithm accuracy. The motion recognition problem in future HRC assembly systems can provide a more accurate result. For instance, the multimodal fusion method can also be used in human motion recognition to improve the detection accuracy of the human operator's motions.

Secondly, the second approach clearly shows the possibility to predict the human operator's motion trajectory during assembly. Building upon the result of the first approach, the second approach explores the possibility to predict the human operator's motion during the ongoing assembly motion. During the experiment, one of the important take away is the importance of uncertainty handling. Since the algorithm will always predict the final handover location according to the dataset, the current state will be a crucial point of reference. The author adapted the approach called MC dropout [76], to randomly drop some neurons during the training process, so the trained model will always provide certain randomness. During inference, the author collectively applied 10 different trained models. The human operator's handover motion target will be confirmed only when the majority of the models agree on the same target. Thus, the uncertainty of the prediction will be greatly reduced. Overall, with an accurate prediction on the human operator's assembly motion trajectory, the assembly efficiency can be improved.

Lastly, the future direction of human motion prediction can be the prediction of handover strategy, as the handover process still can provide very different scenarios due to the different shapes of assembly parts. If the problem is solved, the efficiency of HRC assembly can be further improved. Another future direction is the applicability study of human motion prediction in a real industrial environment.

8.4 Discussion on research question 4

RQ4: How to achieve HRC lead-through remotely with efficient response?

The last research question discusses the possibility for efficient remote HRC lead-through. The target use case is a hazard production environment where accessibility is limited. More application scenarios can be found, with the development and advancement of the cyber-physical system and Industry 4.0. The same concept can be applied in future fully automated human-free factories. The proposed system can be used for remote commissioning by a robot controlled remotely.

Firstly, the author provides a special design for the system to work in four different modes. Such a flexible design can be well adapted in the current production and manufacturing industry. In a real production environment, Mode 1 can be adapted to allow a real robot controls another real robot. During operator training, the system can be switched to Mode 4, where everything is virtual. The initial adaption of a new assembly sequence can be tested in Mode 3 before integrated into the production line. During remote commissioning, Mode 2 can be utilised to manipulate a virtual robot to control a real robot remotely.

Secondly, the adaption of the ROS system. The advantages and disadvantages of the ROS system are very clear. ROS system can provide excellent connectivity to different robots, sensors, and algorithms. The whole system can be built fast and easy. However, some robot manufacturers cannot provide reliable connectivity to ROS, since fewer integration efforts were made. Therefore, it is only possible to use robot manufacturers' official communication protocols. Normally, the official communication protocols can provide more reliable control towards the robot.

Lastly, the response time of the system is also an important aspect. The response time of such a system is a vital indicator of the system's performance. The author has measured different delays of the system. The delay in communication is normally below 0.01 seconds. The path planning time delay is normally around 0.015 seconds. Since the architectures of the 4 modes are very similar, the communication and the path planning time of the 4 Modes should be the same. However, there are significantly different response time between Modes 1, 2 and Modes 3, 4. The differences can be introduced by the controller of the physical robot. Modes 3 and 4 only need to control a virtual robot, whereas, in Modes 1 and 2, a real robot need to be controlled. Overall, the proposed architecture already saved the computational bandwidth by the model-based approach. Further control mechanisms can be explored to reduce the system latency.

Chapter 9

Conclusion

The thesis studied the possibilities to increase efficiency and accuracy of HRC assembly systems from four main directions. The first direction is HRC assembly context recognition, which focuses on the identification and recognition of relevant assembly context in the assembly environment. The definition of assembly context is given, and algorithms are designed to recognise the assembly context. The second direction is accurate multimodal robot control. The algorithm to increase the recognition accuracy is developed. The third direction is human motion prediction. Two different approaches towards accurate and timely prediction of human operators' motions are provided. The efficiency of HRC assembly systems is further boosted. The last direction of the study is remote HRC. The remote HRC scenario is analysed, possible solution is also provided. Along the four directions, key algorithms, system designs, and experiments are investigated. Furthermore, the advantages, drawbacks, and future directions of the approaches are explored.

Bibliography

- [1] U. N. Publications, *World Population Ageing 2019 Highlights*, ser. Economic & social affairs. UN, 2020, ISBN: 9789211483253. [Online]. Available: <https://books.google.se/books?id=2SjAzAEACAAJ>.
- [2] J. L. Vandergrift, J. E. Gold, A. Hanlon, and L. Punnett, “Physical and psychosocial ergonomic risk factors for low back pain in automobile manufacturing workers,” *Occupational and environmental medicine*, vol. 69, no. 1, pp. 29–34, 2012.
- [3] T. Brogårdh, “Present and future robot control development—an industrial perspective,” *Annual Reviews in Control*, vol. 31, no. 1, pp. 69–79, 2007.
- [4] IFR, “Executive Summary World Robotics 2019 Industrial Robots,” International Federation of Robotics, Tech. Rep., 2019.
- [5] A. Sauppé and B. Mutlu, “The social impact of a robot co-worker in industrial settings,” in *Proceedings of the 33rd annual ACM conference on human factors in computing systems*, 2015, pp. 3613–3622.
- [6] J. Zhang, H. Liu, Q. Chang, L. Wang, and R. X. Gao, “Recurrent neural network for motion trajectory prediction in human-robot collaborative assembly,” *CIRP Annals*, 2020.
- [7] R. X. Gao, L. Wang, M. Helu, and R. Teti, “Big data analytics for smart factories of the future,” *CIRP Annals*, 2020.
- [8] L. Wang, R. Gao, J. Vánca, J. Krüger, X. V. Wang, S. Makris, and G. Chryssolouris, “Symbiotic human-robot collaborative assembly,” *CIRP annals*, vol. 68, no. 2, pp. 701–726, 2019.
- [9] A. Mohammed, B. Schmidt, and L. Wang, “Active collision avoidance for human–robot collaboration driven by vision sensors,” *International Journal of Computer Integrated Manufacturing*, vol. 30, no. 9, pp. 970–980, 2017.
- [10] B. Schmidt and L. Wang, “Depth camera based collision avoidance via active robot control,” *Journal of manufacturing systems*, vol. 33, no. 4, pp. 711–718, 2014.

BIBLIOGRAPHY

- [11] H. Liu, Y. Wang, W. Ji, and L. Wang, "A context-aware safety system for human-robot collaboration," *Procedia Manufacturing*, vol. 17, pp. 238–245, 2018.
- [12] H. Liu and L. Wang, "Collision-free human-robot collaboration based on context awareness," *Robotics and Computer-Integrated Manufacturing*, vol. 67, p. 101997,
- [13] L. Wang, B. Schmidt, and A. Y. Nee, "Vision-guided active collision avoidance for human-robot collaborations," *Manufacturing Letters*, vol. 1, no. 1, pp. 5–8, 2013.
- [14] J. Krüger, T. K. Lien, and A. Verl, "Cooperation of human and machines in assembly lines," *CIRP annals*, vol. 58, no. 2, pp. 628–646, 2009.
- [15] L. Wang, "From intelligence science to intelligent manufacturing," *Engineering*, vol. 5, no. 4, pp. 615–618, 2019.
- [16] H. Snyder, "Literature review as a research methodology: An overview and guidelines," *Journal of Business Research*, vol. 104, pp. 333–339, 2019.
- [17] C. Okoli and K. Schabram, "A guide to conducting a systematic literature review of information systems research," 2010.
- [18] P. Baxter, S. Jack, *et al.*, "Qualitative case study methodology: Study design and implementation for novice researchers," *The qualitative report*, vol. 13, no. 4, pp. 544–559, 2008.
- [19] R. Johansson, "On case study methodology," *Open house international*, vol. 32, no. 3, p. 48, 2007.
- [20] A. Fink, *Conducting research literature reviews: From the internet to paper*. Sage publications, 2019.
- [21] R. K. Yin, *Case study research and applications: Design and methods*. Sage publications, 2017.
- [22] J. Rowley, "Using case studies in research," *Management research news*, 2002.
- [23] M. B. Miles, A. M. Huberman, M. A. Huberman, and M. Huberman, *Qualitative data analysis: An expanded sourcebook*. sage, 1994.
- [24] A. Vysocky and P. Novak, "Human-robot collaboration in industry," *MM Science Journal*, vol. 9, no. 2, pp. 903–906, 2016.
- [25] V. Villani, F. Pini, F. Leali, and C. Secchi, "Survey on human-robot collaboration in industrial settings: Safety, intuitive interfaces and applications," *Mechatronics*, vol. 55, pp. 248–266, 2018.

- [26] X. V. Wang, Z. Kemény, J. Váncza, and L. Wang, “Human–robot collaborative assembly in cyber-physical production: Classification framework and implementation,” *CIRP annals*, vol. 66, no. 1, pp. 5–8, 2017.
- [27] A. De Luca and F. Flacco, “Integrated control for phri: Collision avoidance, detection, reaction and collaboration,” in *2012 4th IEEE RAS & EMBS International Conference on Biomedical Robotics and Biomechatronics (BioRob)*, IEEE, 2012, pp. 288–295.
- [28] H. A. Yanco and J. Drury, “Classifying human-robot interaction: An updated taxonomy,” in *2004 IEEE International Conference on Systems, Man and Cybernetics (IEEE Cat. No. 04CH37583)*, IEEE, vol. 3, 2004, pp. 2841–2846.
- [29] H. Liu and L. Wang, “Gesture recognition for human-robot collaboration: A review,” *International Journal of Industrial Ergonomics*, vol. 68, pp. 355–367, 2018.
- [30] A. I. Guha and S. Tellex, “Towards meaningful human-robot collaboration on object placement,” in *Proc. RSS Workshop on Planning for Human-Robot Interaction: Shared Autonomy and Collaborative Robotics*, 2016.
- [31] W. Wang, R. Li, Y. Chen, and Y. Jia, “Human intention prediction in human-robot collaborative tasks,” in *Companion of the 2018 ACM/IEEE international conference on human-robot interaction*, 2018, pp. 279–280.
- [32] P. Tsarouchi, A.-S. Matthaiakis, S. Makris, and G. Chryssolouris, “On a human-robot collaboration in an assembly cell,” *International Journal of Computer Integrated Manufacturing*, vol. 30, no. 6, pp. 580–589, 2017.
- [33] E. Lakomkin, M. A. Zamani, C. Weber, S. Magg, and S. Wermter, “On the robustness of speech emotion recognition for human-robot interaction with deep neural networks,” in *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, IEEE, 2018, pp. 854–860.
- [34] C. Breazeal, C. D. Kidd, A. L. Thomaz, G. Hoffman, and M. Berlin, “Effects of nonverbal communication on efficiency and robustness in human-robot teamwork,” in *2005 IEEE/RSJ international conference on intelligent robots and systems*, IEEE, 2005, pp. 708–713.
- [35] M. Askarpour, D. Mandrioli, M. Rossi, and F. Vicentini, “Safer-hrc: Safety analysis through formal verification in human-robot collaboration,” in *International Conference on Computer Safety, Reliability, and Security*, Springer, 2016, pp. 283–295.

BIBLIOGRAPHY

- [36] R.-J. Halme, M. Lanz, J. Kämäräinen, R. Pieters, J. Latokartano, and A. Hietanen, “Review of vision-based safety systems for human-robot collaboration,” *Procedia CIRP*, vol. 72, pp. 111–116, 2018.
- [37] I. 1.-1. 2011, *Robots and robotic devices—safety requirements for industrial robots—part 1: Robots*, 2011.
- [38] I. ISO, “10218-2: 2011: Robots and robotic devices—safety requirements for industrial robots—part 2: Robot systems and integration,” *Geneva, Switzerland: International Organization for Standardization*, 2011.
- [39] H. Liu, T. Fang, T. Zhou, and L. Wang, “Towards robust human-robot collaborative manufacturing: Multimodal fusion,” *IEEE Access*, vol. 6, pp. 74 762–74 771, 2018.
- [40] J. A. Hartigan and M. A. Wong, “Algorithm as 136: A k-means clustering algorithm,” *Journal of the Royal Statistical Society. Series C (Applied Statistics)*, vol. 28, no. 1, pp. 100–108, 1979.
- [41] L. E. Peterson, “K-nearest neighbor,” *Scholarpedia*, vol. 4, no. 2, p. 1883, 2009.
- [42] I. Steinwart and A. Christmann, *Support vector machines*. Springer Science & Business Media, 2008.
- [43] L. R. Rabiner, “A tutorial on hidden markov models and selected applications in speech recognition,” *Proceedings of the IEEE*, vol. 77, no. 2, pp. 257–286, 1989.
- [44] A. Liaw, M. Wiener, *et al.*, “Classification and regression by random-forest,” *R news*, vol. 2, no. 3, pp. 18–22, 2002.
- [45] T. Chen and C. Guestrin, “Xgboost: A scalable tree boosting system,” in *Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining*, 2016, pp. 785–794.
- [46] A. Krizhevsky, I. Sutskever, and G. E. Hinton, “Imagenet classification with deep convolutional neural networks,” in *Advances in neural information processing systems*, 2012, pp. 1097–1105.
- [47] A. Graves, A.-r. Mohamed, and G. Hinton, “Speech recognition with deep recurrent neural networks,” in *2013 IEEE international conference on acoustics, speech and signal processing*, IEEE, 2013, pp. 6645–6649.
- [48] S. Hochreiter and J. Schmidhuber, “Long short-term memory,” *Neural computation*, vol. 9, no. 8, pp. 1735–1780, 1997.
- [49] A. McCallum, K. Nigam, *et al.*, “A comparison of event models for naive bayes text classification,” in *AAAI-98 workshop on learning for text categorization*, Citeseer, vol. 752, 1998, pp. 41–48.

- [50] D. A. Reynolds, T. F. Quatieri, and R. B. Dunn, "Speaker verification using adapted gaussian mixture models," *Digital signal processing*, vol. 10, no. 1-3, pp. 19–41, 2000.
- [51] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets," in *Advances in neural information processing systems*, 2014, pp. 2672–2680.
- [52] S. B. Kotsiantis, I. Zaharakis, and P. Pintelas, "Supervised machine learning: A review of classification techniques," *Emerging artificial intelligence applications in computer engineering*, vol. 160, pp. 3–24, 2007.
- [53] A. Y. Ng and M. I. Jordan, "On discriminative vs. generative classifiers: A comparison of logistic regression and naive bayes," in *Advances in neural information processing systems*, 2002, pp. 841–848.
- [54] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *nature*, vol. 521, no. 7553, pp. 436–444, 2015.
- [55] J. Wang, Y. Ma, L. Zhang, R. X. Gao, and D. Wu, "Deep learning for smart manufacturing: Methods and applications," *Journal of Manufacturing Systems*, vol. 48, pp. 144–156, 2018.
- [56] B. Schölkopf, "The kernel trick for distances," in *Advances in neural information processing systems*, 2001, pp. 301–307.
- [57] M. E. Tipping, "Sparse bayesian learning and the relevance vector machine," *Journal of machine learning research*, vol. 1, no. Jun, pp. 211–244, 2001.
- [58] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proceedings of the IEEE*, vol. 86, no. 11, pp. 2278–2324, 1998.
- [59] Y. Yao and G. Doretto, "Boosting for transfer learning with multiple sources," in *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2010, pp. 1855–1862. DOI: 10.1109/CVPR.2010.5539857.
- [60] S. J. Pan, Q. Yang, *et al.*, "A survey on transfer learning," *IEEE Transactions on knowledge and data engineering*, vol. 22, no. 10, pp. 1345–1359, 2010.
- [61] H. Liu, T. Fang, T. Zhou, Y. Wang, and L. Wang, "Deep learning-based multimodal control interface for human-robot collaboration," *Procedia CIRP*, vol. 72, pp. 3–8, 2018.
- [62] J. Ngiam, A. Khosla, M. Kim, J. Nam, H. Lee, and A. Y. Ng, "Multimodal deep learning," 2011.

BIBLIOGRAPHY

- [63] N. Srivastava and R. R. Salakhutdinov, “Multimodal learning with deep boltzmann machines,” in *Advances in neural information processing systems*, 2012, pp. 2222–2230.
- [64] A. Eitel, J. T. Springenberg, L. Spinello, M. Riedmiller, and W. Burgard, “Multimodal deep learning for robust rgb-d object recognition,” in *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, IEEE, 2015, pp. 681–687.
- [65] Q. Xiong, J. Zhang, P. Wang, D. Liu, and R. X. Gao, “Transferable two-stream convolutional neural network for human action recognition,” *Journal of Manufacturing Systems*, 2020.
- [66] P. Warden, “Speech commands: A dataset for limited-vocabulary speech recognition,” *arXiv preprint arXiv:1804.03209*, 2018.
- [67] F. Zheng, G. Zhang, and Z. Song, “Comparison of different implementations of mfcc,” *Journal of Computer science and Technology*, vol. 16, no. 6, pp. 582–589, 2001.
- [68] F. Weichert, D. Bachmann, B. Rudak, and D. Fisseler, “Analysis of the accuracy and robustness of the leap motion controller,” *Sensors*, vol. 13, no. 5, pp. 6380–6393, 2013.
- [69] J. Donahue, Y. Jia, O. Vinyals, J. Hoffman, N. Zhang, E. Tzeng, and T. Darrell, “Decaf: A deep convolutional activation feature for generic visual recognition,” in *International conference on machine learning*, 2014, pp. 647–655.
- [70] L. v. d. Maaten and G. Hinton, “Visualizing data using t-sne,” *Journal of machine learning research*, vol. 9, no. Nov, pp. 2579–2605, 2008.
- [71] T. Mikolov, M. Karafiát, L. Burget, J. Černocký, and S. Khudanpur, “Recurrent neural network based language model,” in *Eleventh annual conference of the international speech communication association*, 2010.
- [72] M. Quigley, K. Conley, B. Gerkey, J. Faust, T. Foote, J. Leibs, R. Wheeler, and A. Y. Ng, “Ros: An open-source robot operating system,” in *ICRA workshop on open source software*, Kobe, Japan, vol. 3, 2009, p. 5.
- [73] S. H. Juan and F. H. Cotarelo, “Multi-master ros systems,” *Institut de Robotics and Industrial Informatics*, pp. 1–18, 2015.
- [74] G. Papandreou, T. Zhu, N. Kanazawa, A. Toshev, J. Tompson, C. Bregler, and K. Murphy, “Towards accurate multi-person pose estimation in the wild,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 4903–4911.

BIBLIOGRAPHY

- [75] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, “Bert: Pre-training of deep bidirectional transformers for language understanding,” *arXiv preprint arXiv:1810.04805*, 2018.
- [76] Y. Gal and Z. Ghahramani, “Dropout as a bayesian approximation: Representing model uncertainty in deep learning,” in *international conference on machine learning*, 2016, pp. 1050–1059.