CONTEXT-AWARE WORKER INTENT INTERPRETATION FOR HUMAN-ROBOT COLLABORATION IN CONSTRUCTION

by

Xin Wang

A dissertation submitted in partial fulfillment of

the requirements for the degree of

Doctor of Philosophy

(Civil and Environmental Engineering)

at the

UNIVERSITY OF WISCONSIN-MADISON

2024

Date of final oral examination: 05/09/2024

The dissertation is approved by the following members of the Final Examination Committee:

Awad Hanna, Professor, Civil and Environmental Engineering
Dharmaraj Veeramani, Professor, Industrial and Systems Engineering
Jeffrey Russell, Professor, Civil and Environmental Engineering
Yin Li, Assistant Professor, Biostatistics and Medical Informatics
Zhenhua Zhu, Assistant Professor, Civil and Environmental Engineering

ACKNOWLEDGMENTS

Embarking on the journey towards completing a PhD is an endeavor marked by both profound challenges and extraordinary triumphs. As I reflect on this monumental achievement, I find myself humbled by the incredible support and guidance that have propelled me forward. Commencing my doctoral studies just before the tumultuous onset of the Covid-19 pandemic, I encountered a landscape fraught with uncertainties and obstacles. Nonetheless, I made it to the finish line! I am profoundly grateful to the University of Wisconsin-Madison for providing me with an enriching academic environment over the past four and a half years. Madison's serene beauty has been a constant source of inspiration, enhancing my academic journey in countless ways. I am indebted to the faculty, staff, and fellow students for their support and camaraderie through this transformative journey, enabling me to successfully obtain my doctorate despite the multitude of unforeseen obstacles that arose.

First and foremost, I would like to express my sincere thanks to my PhD supervisor, Dr. Zhenhua Zhu. His constant support, guidance, and encouragement have been invaluable throughout the entire process. From the initial stages of refining my research proposal to the final submission of my thesis, his unwavering presence and wealth of wisdom have been instrumental in shaping my academic growth. I highly valued the weekly meetings we held, which not only served as crucial checkpoints to keep me on track academically, but also provided me with plenty of encouragement. I am deeply appreciative of the invaluable contributions Dr. Zhu has made to my personal and academic development.

In addition to my supervisor, I extend my sincerest gratitude to my PhD thesis committee members: Dr. Dharmaraj Veeramani, Dr. Yin Li, Dr. Awad Hanna and Dr. Jeffrey Russell. I am immensely thankful to Dr. Veeramani for his monthly meetings and invaluable inputs. His

insightful questions and constructive suggestions have been instrumental in guiding me through my academic endeavors. Furthermore, I deeply appreciate Dr. Li's invaluable assistance with the technical aspects of my thesis. His profound expertise in computer science has been indispensable in navigating and addressing the complexities of my research. Lastly, I extend my heartfelt thanks to Dr. Hanna and Dr. Russell for their invaluable introductions to construction management practices and for shedding light on how my research could potentially be applied within the construction industry.

I am indebted to my exceptional lab mates Wei and Liqun, whose supports have been a constant source of motivation. Our collaborative writing sessions and informal chats provided a lifeline during the most challenging times. I am proud to say that we became more than just lab partners, but good friends. Wei, thanks for your bad jokes. Liqun, your virtual excavator model looks fantastic although it did not help my research too much. Besides my lab mates, I am also immensely grateful for the friendships I have formed with fellow students throughout my academic journey. Their camaraderie, encouragement, and shared experiences have made the challenges more bearable and the triumphs more joyful.

Finally, I would like to express my deepest gratitude to my family - my father, mother, and sister. Their unwavering love, support, and sacrifices have been the bedrock of my journey through the challenges of pursuing a PhD. Additionally, I extend my heartfelt thanks to my girlfriend, Yijing, for her steadfast love, understanding, and support throughout this journey. Her encouragement and belief in me have been a driving force behind my success. I am truly fortunate to have her by my side.

In closing, as I reflect on the culmination of this remarkable journey, I am reminded of the profound impact of the relationships forged and the personal growth attained. Each individual

mentioned in these acknowledgments has played an integral role in shaping not only my academic pursuits but also my personal development. As I embark on the next chapter of my journey, I carry with me the lessons learned, the memories cherished, and the enduring friendships forged. With heartfelt gratitude, I extend my deepest appreciation to each person who has walked alongside me on this extraordinary path.

“Look at the clouds in the sky, gathering and dispersing, dispersing and gathering again. Such is the way of life, with its comings and goings.”

ABSTRACT

With years of technical development, construction robots and/or autonomous machines have shown the potential to enhance productivity and safety in the construction industry. However, robots and autonomous machines have not been widely adopted on construction sites. There are various reasons contributing to the low adoption of robots in construction including operational and personnel barriers. An intuitive and accurate human intent representation can help contribute to resolving the above barriers. On one hand, the establishment of such representation can greatly enhance the operability of robots in dynamic construction environments. On the other hand, such representation helps to build a safe environment for worker-robot collaboration.

So far, there are much related work about human intent interpretation, such as recognizing human actions, identifying hand gestures, automating eye tracking, and understanding speech language. These references show that the recent advance in technologies (e.g., computer vision, wearable sensor) has built a solid foundation to interpret and predict the intentions of onsite construction workers and support their collaborations with robotic machines. However, two research questions need to be answered before making such interpretations and predictions work well on construction sites: (1) How to capture and interpret worker intents accurately for worker-robot collaborations in construction? (2) How to extract useful context information to facilitate representation on construction sites?

To answer these two questions, a context-aware human intent representation is proposed in this study to support human-robot collaboration on construction sites. It consists of three components: recognition building, object-enhanced interaction and machine-aware collaboration. In the first component recognition building, a novel vision-based method is developed to achieve gesture recognition. Since computer vision technologies may be easily affected by the construction

environment (e.g., diverse dust and light conditions), a novel wearable sensors-based method is then developed. Through a comparison between these two methods, the sensor-based method is found to have the advantages of early triggering and robust anti-interference capabilities, but may incur higher communication costs in human-robot interactions.

In the second component object-enhanced interaction, a novel object-aware method is proposed for human-robot collaboration in construction, integrating first-person vision and gesture recognition. An end-to-end two-stream network which includes a first-person view-based stream and a motion sensory data-based stream is designed. The first-person view-based stream models the user's gaze using an attention module to concentrate on the important spatiotemporal regions of first-person video for context extraction. The motion sensory data-based stream is used to process the motion sensory data to extract features related to the hand motions. Finally, the feature maps coming from these two streams are fused to achieve the hand gesture recognition.

In the third component machine-aware collaboration, a novel machine-aware hand gesture recognition method is developed as a human-robot interface for use on construction sites having multiple types of machines. The developed method firstly relies on an eye tracker to visually detect and track construction machines in the first-person view. Then, the machine-of-interest is determined based on the bounding boxes of machines and gaze points. Finally, a hand gesture recognition architecture is incorporated with the machine information for conveying messages to the machine-of-interest.

The above methods have been evaluated and tested by experiments on different construction sites. The evaluation results have demonstrated that the proposed methods can capture and interpret the worker intents accurately with context awareness to support human-robot collaborations on construction sites. The expected contributions of the proposed methods include

improving interaction efficiency with construction robots, decreasing onsite safety issues, refining the design and implementation of construction robots, promoting the adoption of robots in construction, etc.

Future work will focus on the following aspects. First, expanding the dataset to encompass a wider range of construction sites, subjects, equipment, tasks, and types of hand gestures will enhance the robustness of the proposed methods. Second, efforts will be made to address the technological challenges associated with automation, including electronics, computation, and communication. Third, the integration of sensor and data fusion techniques will be explored to enhance the reliability of message communication between workers and machines. Fourth, gathering feedback from workers on novel human-robot interfaces will be prioritized to evaluate their trust in construction robots and potential safety concerns during human-robot interactions on construction sites.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER ONE: INTRODUCTION

With years of technical development, construction robots and/or autonomous machines have shown the potential to increase construction productivity as well as solve problems such as safety risks in the construction field [1,2]. However, robots and autonomous machines have not been widely adopted in the construction industry [3]. In the US, only a small percentage of construction companies have effectively integrated robotic technologies in their projects [4]. Figure 1.1 showed a survey result [5], which was conducted by AAB Robotics among 200 US construction companies in 2021 to investigate the implementation degree of robotic process in construction. It reported that 70% of construction companies had not employed any robotic process across their projects while 30% of them already implemented the robotic process. Besides, the adoption of robotics in the construction industry is significantly lower than in other traditionally manual sectors [6]. Based on the World Robotics Report [7] as indicated in Figure 1.2, the top 3 industries which employed the robotic technologies were electrical/electronics, automotive, metal and machinery, separately, in 2023. The operational stock of robots in these 3 industries achieved 28.4%, 24.6% and 11.9%, respectively, of the total stock. The construction industry was categorized as “all others”, which only occupied 11.0% of the whole operation stock.

Figure 1.1 Degree of robotic process in US construction companies

The low adoption of robots and autonomous machines may lead to low productivity and safety issues in construction, since they have technical features to enhance quality and efficiency of the operations and moreover could potentially perform construction tasks in dangerous or challenging environments [8,9]. For example, García de Soto et al. [10] investigated the effects of wall-building robots on productivity by analyzing the cost and time required for the construction of a doubled-curved concrete wall. The results indicated that the cost would increase from 4276 USD/m^3 to 5288 USD/m^3 and the time would rise from 8.30 h/m^3 to 16.93 h/m^3 without the adoption of robots. Another example could be found in the work of Martinez et al. [11]. They employed Unmanned Aerial Vehicles (UAVs) to help the limited number of safety managers to quickly and properly inspect the inaccessible, hard-to-reach, or unsafe locations on the site and further enhance the safety assessments. Their work showed that the safety hazards that which could be identified by managers decreased from #21.2 to #17.8 on average without the help of UAV-based process.

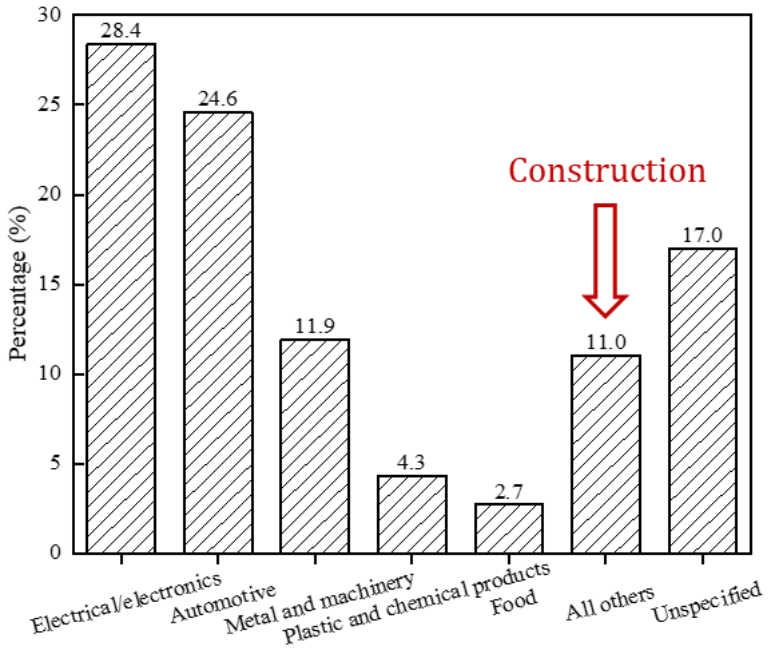


Figure 1.2 World Robotics Report in 2023

There are various reasons which lead to the low adoption of robots in construction. They can be generally classified into five different categories: economic barriers, operational barriers, organizational barriers, personnel barriers, and technology barriers [12]. The economic barriers are obstacles arising from economic factors that pose difficulties to organizations to acquire robots, such as the initial cost of investment. Operational barriers are factors related to the process of incorporating robots into construction processes. The dynamic nature of operations within the environment of the construction workplace is one of the main factors for the operational barriers [13]. Due to programming limitations, the robot can only be used as a pre-coded execution machine [14,15]. This "thinking" incompetency hampers the implementation of robots in the fast-changing working conditions of construction sites. Organizational barriers are hindrances at the company level that might result in a failure to integrate the robots successfully, such as lack of technology culture, minimal collaboration between industry peers, etc. Personnel barriers are the limitations related to technology users and their relationship with the robots. One major challenge is the workers often feel unsafe working around robots [16,17]. They are less willing to work with or alongside robots because of the potential safety concerns (e.g. collisions) raised by the new interactions. Finally, technology barriers are obstacles directly linked to the capabilities of the robots, such as the accuracy, data security and privacy issues.

An intuitive and accurate human intent representation can help contribute to resolving the above barriers, especially operational and personnel barriers [14,18]. This representation requires a reliable message-passing system by which to interact with robots [13] and serves as a safety feature during close human-robot interaction [19]. On one hand, the establishment of such representation can greatly enhance the operability of robots in dynamic construction environments [13]. It allows robotic systems to conduct the multi tasks guided by workers and plan their motions

ahead to suit workers' future needs. In this case, the behaviors of robots can be easily adjusted and planned based on the workers' intentions, which make them adaptable to the surrounding onsite conditions. On the other hand, such representation helps to build a safe environment for worker-robot collaboration [20]. It provides an opportunity for all the onsite workers to stop or move away the robot when he/she feels unsafe around the robot or anticipates that the robot's behavior would be life-threatening to him/herself or other existing workers. Also, the robots would be able to infer the possible movements of the workers and prevent potential path overlaps or collisions [21].

CHAPTER TWO: RELATED WORK

Robotics in Construction

With the evolution of Industry 4.0, the construction industry has started to adopt the robotic technologies to transfer the traditional and hazardous sites to automated and safer ones. The types of intelligent robotic systems in construction can be generally grouped into four categories: industrial robots, drones, autonomous vehicles and humanoid laborers [22]. Table 1.1 summarized the above four categories. More details of the four categories are discussed in the following sections. It should be noted that there is no consensus regarding a defined categorization and the lines between categories are constantly moved or blurred by new developments in technologies [3].

Table 1.1 Summary of robotic technologies in construction

Category	Typical applications	References
Industrial robots	Articulated robots for manufacturing	[23,24]
	Cartesian printers for 3D printing	[25–30]
	Cobots to work alongside humans	[31,32]
Drones	Topographic mapping	[33]
	Equipment tracking	[34]
	Remote monitoring and inspection of worksites	[35]
Autonomous vehicles	Automated drilling	[36]
	Automated excavation	[37,38]
	Automated earth moving	[39,40]
Humanoid laborers	HRP-5P	[41]
	NASA’s Valkyrie	[42,43]

Industrial Robots

Industrial robots are robotic arms that can move in several directions and can be programmed to carry out many different types of tasks in construction environments [44]. They are automated, programmable and capable of movement on three or more axes. Given their higher levels of accuracy, industrial robots can be used to produce higher quality products which results in the reduction of time required for quality control [45]. Also, consider that they can complete tasks without requiring stoppages or breaks. This ability to continuously operate without fatigue makes robots highly productive and more efficiently than humans [46].

So far, there are three common uses of industrial robots in construction, which are articulated robots for manufacturing, cartesian printers for 3D printing and collaborative robots to work alongside humans, separately. Articulated robots resemble a human arm closely and can be used for a variety of manufacturing tasks and applications such as simple assembling [23], welding [24], etc. For example, Greaves and Jenkins [24] depicted a steel-welding robot which employed a laser sensor to detect the configuration of a joint and then performed the welding in an optimized computer-controlled way. Cartesian 3D printers refer to the robots that use the three-axis system of cartesian coordinates to determine the positions and direction of the print head precisely [25]. Most of the research [26–28] has been conducted for concrete printing, in particular for concrete extrusion and powder bed processes. Others focused on developing technologies to build infrastructure components made of metal [29] or polymers [30]. Besides, collaborative robots, also known as cobots, are the robots working alongside humans and performing tasks that would otherwise be too difficult for the human or robot to do alone [31]. Gautam et al. [32] presented a cobot which was dedicated to screwing gypsum board panels to the ceiling of a room. In their work, the cobot conducted the repetitive screwing work to benefit workers by removing

ergonomical strain while the worker was in charge of inspecting wood hardness to adjust the screwing force.

Drones

Drones refer to unmanned aircraft vehicles (UAVs) that can be remotely controlled or fly autonomously to conduct various tasks on construction sites. They are usually equipped with downward-facing sensors, such as RGB, multispectral, thermal or LIDAR and thus can capture a great deal of aerial data in a short time [47]. The information they collect can be sent to a computer via drone software, allowing users to analyze and interpret data. These features allow construction projects to be more efficient, building projects to be better managed, and inspections to be more thorough [48].

Currently, there are various applications of drones in the construction field. These applications include topographic mapping, equipment tracking, remote monitoring and inspection of worksites, etc. For topographic mapping, drones can exponentially cut down on the time to visualize a site's topography due to their ability to map vast quantities of land. Further, the high-resolution images produced by drones can be manipulated into 3D models to pinpoint challenges and plan the layout during pre-construction [22]. For example, Jiang et al. [33] improved image-based modeling techniques to reconstruct the 3D construction site model in real-time. These improved techniques were validated effective for the rapid layout planning in hoisting projects. Also, drones are capable of keeping track of each equipment's location on a job site, recording errors and malfunctions to control the construction quality. Sun [34] proposed a UAV-based system for tracking the road construction machinery to enhance the paving quality. Besides, by sending drones in the air, companies can visually inspect large infrastructures or those located in hard-to-reach areas more quickly and cost effectively. For instance, the work of de Melo et al. [35]

verified that the visual assets collected by UAV could improve the safety inspection on jobsites by means of a better visualization of working conditions.

Autonomous Vehicles

Autonomous vehicles (AVs) are defined as terrestrial equipment that can be piloted remotely, or operate autonomously to conduct various construction jobs automatically. These vehicles include autonomous dozers, excavators, haul trucks, rovers, etc. The current technology of AVs uses sensors (e.g. LiDAR, GPS, camera vision systems [49]) and high precision maps to identify obstacles, paths, etc., which provide a "sense-plan-act" tool for self-driving to reach the given destination and conduct the corresponding jobs [50]. The use of these vehicles could contribute to reducing human operator errors and thus increasing onsite safety [49], as well as offering operators more free time that could be allocated to other tasks [51].

The main applications of autonomous vehicles include automated drilling, excavation, earth moving, etc. For example, Srnoyachki [36] designed an automated drilling system to allow the robotic all-terrain vehicles to perform expedient airfield pavement evaluations on hard paved or concrete surfaces. For automated excavation, Thangavelautham et al. [37] developed a control approach called Artificial Neural Tissue to conduct the multirobot excavation tasks. Fernando et al. [38] presented the development and field validation of an iterative learning-based admittance control algorithm for autonomous excavation in fragmented rock using robotic wheel loaders. To automate the earth moving process, the core technologies including machine perception, localization, navigation and control, communication technologies were discussed and analyzed in the work of Dadhich et al. [39] and Ha et al. [40].

Humanoid Laborers

Humanoid laborers are human body shaped robots that are capable of replacing human workers to perform simple construction tasks. They can generally conduct environmental measurement and object recognition, full-body motion planning and control, task description and execution management, as well as systematize their tasks [52]. These abilities allow them to carry out heavy or dangerous work, thus easing the strain and danger faced by human workers. The employment of humanoid laborers could also help to solve the labor shortage problem in construction [22].

There are two typical examples of humanoid laborers in construction, which are HRP-5P and NASA's Valkyrie, respectively. HRP-5P is developed by National Institute of Advanced Industrial Science and Technology (AIST) to take on often dangerous and backbreaking work for human workers [41]. It uses head-mounted sensors to take 3D measurements of the surrounding environment and own enhanced freedom of movement (37 degrees of freedom in total) of the waist, arms, legs and hands to emulate human movement. Housing the intelligence in HRP-5P has enabled autonomous gypsum board installation, which is a typical example of heavy labor on construction sites [41]. In addition, NASA's Valkyrie is designed and built by the Johnson Space Center (JSC) Engineering Directorate to operate in degraded or damaged human-engineered environments, such as space construction [42]. Valkyrie has four body cameras, 28 torque-controlled joints, and 44 degrees of freedom, making it incredibly dexterous [43]. These technologies make the robot able to use human tools and plan its own path safely across difficult terrain to a location picked by its operator.

Human Intent Interpretation and Prediction

Many efforts have been dedicated to the design of methods for human intent interpretation and prediction. In these methods, multiple cues are utilized to interpret and predict human intents

and support the communications between humans and robots. The employed cues generally include human actions, hand gestures, eye tracking, speech language, etc. [21,53–55] They could be adopted individually or in combination depending on the task contexts and requirements [55]. The following sections introduce the details of the methods based on their employed cues.

Human Actions

Human actions could be one common and favored way to predict human intents. The prediction methods typically focused on discovering and utilizing the spatial-temporal patterns behind human actions recorded in videos for making predictions in a short or long-term. The short-term prediction infers actions based upon temporally incomplete action video sequence in seconds. The long-term prediction infers the future actions based on current observed human actions to understand what is going to happen next.

Many studies have been developed to predict human intents based on human actions. For short-term prediction, Ryoo et al. [56] introduced the concept of "onset" that summarizes pre-activity observations. The "onsets" were combined with visual features from a robot's viewpoint to enable its early recognition of human activities during human-robot interactions [56]. Kong et al. [57] utilized deep neural networks to learn and model sequential context information from fully observed videos; and this information could be transferred to make action predictions from partially observed videos. Compared with the short-term action prediction, the long-term prediction is more challenging due to large uncertainty [58]. The contexts (e.g., interacting objects) play an important role on achieving accurate prediction results. Such context information could be embodied using Stochastic Context Sensitive Grammars [59], Probabilistic Suffix Tree [60], Predictive Accumulative Function [61], Anticipatory Temporal Conditional Random Field [62], etc.

Hand Gestures

Hand gestures, as a common way to express intents, have various application possibilities in human machine interaction due to their simple but effective nature [63–66]. They generally consist of two different categories: subtle motion gestures and large motion gestures, based on their motion degree [67]. Subtle motion gestures only concentrate on hand movements while large motion gestures involve the swing of both hands and arms. Gestures can make the information to be presented easily without the effects of the noisy environments [55]. They also help humans from different backgrounds and cultures to express their thoughts using a standard mode.

To collaborate with humans, robots need to understand human gestures correctly and act efficiently based on the gestures, which makes the automatic capture and interpretation of hand gestures a hot research topic. Various research studies have been developed to achieve hand gesture recognition. They could be classified into two categories, vision-based methods [65,68] and wearable sensors-based methods [69,70], depending on the type of data source they relied on. Using hand-crafted features and deep learning are two kinds of vision-based methods. Traditional methods are generally based on hand-crafted features, such as HOG [71], iDT [72] and MFSK [73]. Besides, more studies were proposed to derive new sophisticated features which could represent the appearance, shape and/or motion changes of a gesture. For instance, Almeida et al. [74] presented a methodology for extracting hand gesture recognition features, such as hand area and hand movement velocity. These features were then fed into a SVM classifier to understand Brazilian sign language. Ahmed et al. [75] developed an integrated statistical algorithm which consisted of three modules: real-time detection of hand regions, hands trajectory tracking, and gesture recognition through the analysis of hand location variations. Memo and Zanuttigh [76]

relied on the local curvature of a hand contour as feature descriptors and input them into a SVM classifier to achieve reliable, real-time hand gesture recognition.

So far, the deep learning technologies have been widely used for vision-based methods. Typically, they can be achieved through 3D-CNN based methods or two-stream CNN architecture. For 3D-CNN based methods, one example can be found in the work of Miao et al. [77]. They firstly extracted spatiotemporal features using Res3D network and combined the extracted features through a canonical correlation analysis. The final recognition was made by a linear SVM classifier. Similarly, Liao et al. [78] relied on the combination of a deep residual 3D-ConvNet and a bi-directional LSTM network to extract the spatiotemporal features of hand gestures from video sequences and score them accordingly for the automatic recognition of the sign language. Wang and Zhu [79] investigated and compared the performances of two 3D-CNN based methods including ResNeXt-101 and Res3D+ConvLSTM+MobileNet on hand gesture recognition. Further, they proposed a vision-based framework to capture and interpret the hand gestures of construction workers [80]. The classification decision was made approximately 4 s later after the start of the hand gesture. In addition, a two-stream CNN architecture where two CNNs are adopted to model spatial and temporal information of sequences, separately, provides another technique for hand gesture recognition. For example, Huang et al. [81] developed C3D network into a two-stream 3D-CNN architecture where one stream focused on the local, detailed hand gestures while the other stream was designed to extract global hand motions.

Motion sensory data provide an alternative data source to achieve hand gesture recognition. They usually can be collected by wearable sensors attached on human bodies or placed near hands, such as surface electromyography (sEMG) sensors, Inertial Measurement Units (IMU), radar sensors, etc. The sensor-based methods could also be classified as two categories: traditional

methods and deep learning. Traditional methods generally relied on hand-crafted features, such as time domain [82] and frequency domain features [67]. These features would be fed into a machine learning classifier to obtain the final classification result. Currently, deep learning has become the mainstream. For example, Côté-Allard et al. [83] applied CNNs on aggregated data from multiple users to identify hand gestures. In their work, CNNs were combined with transfer learning to decrease the data requirement of the training model. Fang et al. [70] designed a new CNN architecture named SLRNet to achieve dynamic gesture recognition. The CNN architecture extracted the features of two hands and fused the features into the fully connected layer. Kim et al. [84] presented an efficient hand gesture recognition algorithm based on a restricted column energy (RCE) neural network. DTW distance was employed as the metric calculation of the RCE neural network to cope with time-dependent data from sensors. Jirak et al. [85] introduced an echo state network (ESN) framework for continuous gesture recognition. The framework included LSTM layers to achieve the automatic detection of the start and end phase of a gesture. Yuan et al. [86] proposed an improved deep feature fusion network to detect long distance dependency in complex hand gestures. In their work, a LSTM model with fused feature vectors was introduced to classify complex hand motions into corresponding categories. Wang and Zhu [87] proposed a system for recognition of construction workers' hand gestures using wearable sensors on fingers. The classification decision was made approximately 2 s later after the start of the hand gesture.

Eye Tracking

Conventionally, eye tracking has been regarded as one of the most visible cues for user behavior/intention recognition [88]. It helps observe and measure eye movements, pupil dilation, gaze point, and blinking to see where subjects of a study focus their visual attention, what they engage with, and what they ignore [89]. Eye tracking could provide unbiased, objective, and

quantifiable data because it intuitively reflects where humans are currently interested. It is also a critical and efficient interface for disabled people (e.g. those with motor disabilities) or humans with their hands occupied due to its hands-free and human-dominated communication nature [90].

There are many efforts dedicated to developing reliable eye tracking-based methods. Zhang et al. [91] presented a novel eye tracking-learning-detection algorithm with tracking feedback. The detection area was adjusted adaptively and narrowed by the tracking feedback to adapt to the situations where human eye was partially blocked or got morphological changes. Santini et al. [92] introduced a novel method named Pupil Reconstructor with Subsequent Tracking (PuReST) for fast and robust pupil tracking. The PuReST consisted of three distinct parts: initial pupil detection, shared tracking preamble, outline and greedy tracker. Bozomitu et al. [93] developed an eye-tracking-based human computer interface for real-time applications. The performance of eight algorithms was analyzed with the results identifying the circular Hough transform approach as the most appropriate pupil detection algorithm for the developed interface. Laddi and Prakash [94] proposed an unobtrusive and calibration-free framework towards eye gaze tracking based interface for desktop environment. The proposed eye gaze tracking involved hybrid approach wherein the unsupervised image gradients method computed the iris centers over the eye regions extracted by the supervised regression-based algorithm. As the technology advances and matures, commercial eye tracking products such as Tobbi Glasses 3 [95], Pupil Core [96] become available in the market and have various potential fields of application.

Moreover, the performance could be improved by the integration with the environment recognition process. Deep learning-based computer vision methodologies such as image segmentation, scene understanding, and object detection have been actively introduced because they provide abundant clues to understand a given environment [88]. For example, Barz and

Sonntag [97] introduced a guided-gaze object classification method for constructing episodic memories of egocentric events. They estimated the gaze points by using Pupil-Labs Eye Tracker [98] and then selected the objects of interest from the gaze estimation with GoogLeNet as the object detector. Kim et al. [99] presented a method that selected a device of interest by linking gaze estimation and object detection technology for user interaction in a smart IoT system. In their work, DeepGaze was employed for gaze estimation while YOLO was used to detect objects. Then, a target IoT device desired for user control was determined by mappings between the gaze and the detected objects. Khosravan et al. [100] developed a novel system, called Gaze2Segment, to support radiologists' reading experience with an automatic image segmentation task. Firstly, a visual attention map was created using the radiologists' gaze information and a computer-derived saliency map was extracted from the gray-scale CT images. The visual attention map and saliency map were further integrated to segment CT images automatically. Cho and Kang [88] proposed a novel human gaze-aware attentive object detection framework to understand human behavior and the surrounding environment. The proposed framework detected users' attentive objects and showed more precise and robust performance against object-scale variations.

Speech Language

Speech language is another common and favored communication channel in many human-human interaction scenarios. Microphones capture raw audio data, and the system interprets it as commands to instruct robots [55]. It is easy for humans to learn, remember and use because its structure and vocabulary are already rooted in humans' heads. Also, speech language allows considerable flexibility in executing the steps of a task.

Most of the current studies relied on natural language processing. For example, Cantrell et al. [101] and Shen and Inkpen [102] demonstrated how robots used human-robot dialogs to learn

the meanings of action verbs and acquire knowledge about new actions from humans. Deuerlein et al. [103] designed a cloud-based speech processing system to recognize commands and convert them to machine-readable codes. The natural language processing typically requires pre-coded domain knowledge and large training datasets. Shivakumar et al. [104] addressed the spoken language intent detection under noisy conditions. In their work, a word feature representation was proposed to compensate for the errors made by recognition systems and to increase the robustness of the spoken language understanding. Brawler et al. [105] presented a framework that integrated the speech processing with contextual information, so that both "speech" and "context" models could be maintained and incrementally updated to jointly classify a human's intent. Huang et al. [106] leveraged unpaired text data for training end-to-end speech-to-intent systems. The systems could directly extract the intent label associated with a spoken utterance without explicitly transcribing the utterance.

Context Information Expression

Recently, one of the emerging directions of human-robot collaboration (HRC) research is context awareness [20,107,108]. The situation where human operators and robots collaboratively work together in the same workstation, creates a unique environmental context [20]. The context expression of collaboration generally involves identification of the objects (parts or tools) that the human worker is handling, the sequences of performing actions during the tasks, and the work space environment [108]. With context awareness, the robots will be able to know how they can effectively assist the human operator to improve productivity of the collaboration system while maintaining safety [109]. The current applications are mostly focused on manufacturing tasks [108,110–114]. Others are developed to support human robot collaboration in home settings [115,116], simulated environments [117], etc.

Objects Identification

Consider that human body motions associated with certain tasks may be similar regardless of the context of the tasks. The identification of objects context can help the robot understand what specific tasks the human operator intends to perform so that the robot can assist correspondingly. For example, Wang et al. [108] combined part/tool identification into human motion recognition to assist in the interpretation of the operator's intention. The combined system consisted of two steps: categorizing the human motions into representative categories, specifying what parts/tools the human operator was holding from the identified category. Liu et al. [110] employed an improved CNN+LSTM method to categorize the human motions based on a combination of objects and actions. Bruckschen et al. [115] presented a framework that utilized a transition model as well as observations of the human's location and pose for the prediction of their movement goal. The transition model learned information about previous object interactions and could be used to infer possible objects the human would interact with. Corona et al. [118] developed a novel context-aware motion prediction architecture. In their architecture, a semantic-graph model was used to parameterize the human and objects in the scene and their mutual interactions. These learnt interactions were fed into RNN to predict future movements of the humans and objects.

Working Sequences Understanding

The working sequence is a sequence of human motions, which could be used to represent a certain working task. The working sequences understanding provides the past experiences about how humans performed working tasks and could be combined with human motion recognition for predicting future motions. Mainprice and Berenson [117] categorized human actions based on the probabilistic models which learned how the human moved to perform the set of tasks. During task execution, the category that best fitted the real movements of the human was selected and used as

a predictor of the human future movements. This prediction was finally considered in order to generate the optimal robot trajectory. Liu and Wang [119] used a motion transition probability matrix to model human motion sequence. In online testing, the results of human motion recognition were obtained and combined with the transition probability matrix to predict future motions. Further, they [20] proposed a context awareness-based collision-free HRC system. Based on the pre-defined operational sequence, the system could monitor and predict human operator's poses to avoid potential collisions. Tang et al. [120] proposed a motion context modeling by summarizing the historical human motion with respect to the current skeleton. Such motion context could help to capture the human motion patterns and ease the motion uncertainties, thus benefiting the long term predictions.

Work Space Environment Parsing

The work space environment parsing is intended to construct a virtual and comprehensive representation for the physical working environment. The environment parsing formats include scene graph [112], 2D map [116], 3D expression [111,113], etc. They provide informative context for safe robot movements [114]. For instance, Hata et al. [112] modeled scene graph for safe HRC in a warehouse navigation case. In their work, Mask R-CNN was utilized to segment scene objects from images and subsequently encoded the extracted object information into a scene graph for further fuzzy logic-based risk management. In the framework of Hu et al. [116], 2D semantic map generated via simultaneous localization and mapping (SLAM) technique was leveraged to represent the global map of the environment, Mask R-CNN was employed to detect scene objects, and LSTM was utilized to parse human instructions. The framework enabled a robot to safely walk through a changing environment by following human instructions. Liu et al. [111] aimed at collision-free robot planning for HRC manufacturing tasks. In their work, OctMap was leveraged

to represent the 3D occupancy status of an HRC working space, while Markov Decision Process (MDP) and reinforcement learning (RL) techniques were adopted for collision avoidance. Choi et al. [113] proposed a safety measurement method for HRC system, utilizing 3D point cloud representation for the physical environment and synchronizing with a digital twin model in real-time for further distance measurement in the virtual space.

CHAPTER THREE: RESEARCH GAPS, OBJECTIVE AND METHODOLOGY OVERVIEW

The recent advance in technologies (e.g. computer vision, wearable sensor) has built a solid foundation to interpret and predict the intentions of onsite construction workers and support their collaborations with robotic machines. However, several challenges need to be addressed before making such interpretations and predictions work well on construction sites. First, most of the current methods for human intent interpretations were established in more controlled environments (e.g., indoor settings [66], human sitting or standing still [121], camera close to humans [122]). However, the construction environment is characterized by its dynamic and complex nature, filled with various tools, materials, and machinery. Also, the human intent interpretation in the construction fields may be impacted by environmental conditions and the existence of other workers. Consequently, accurately capturing and interpreting worker intents within the construction field presents considerable challenges. Second, timely context awareness has been proven effective to support human robot collaboration in home settings [115,116], manufacturing tasks [108,110], and simulated environments [117]. However, it is still not clear what context information should be extracted and how they could facilitate human intent representation in construction.

The main objective of this study is to build a context-aware human intent representation that supports human-robot collaboration on construction sites. The representation tries to answer the following questions: (1) How to capture and interpret worker intents accurately for worker-robot collaborations in construction? (2) How to extract useful context information to facilitate representation on construction sites? Among the various types of human-robot interfaces, non-verbal communication is deemed to be well-suited for noisy work environments [55]. Therefore, hand gestures and first-person vision are employed as recognition cues for interpreting and

predicting workers' intents due to their natural and intuitive nature as well as strong anti-interference to noisy environments. The built representation is expected to provide a unified and generalized approach for interpreting and predicting human's intent to interact with robotic machines on real construction sites.

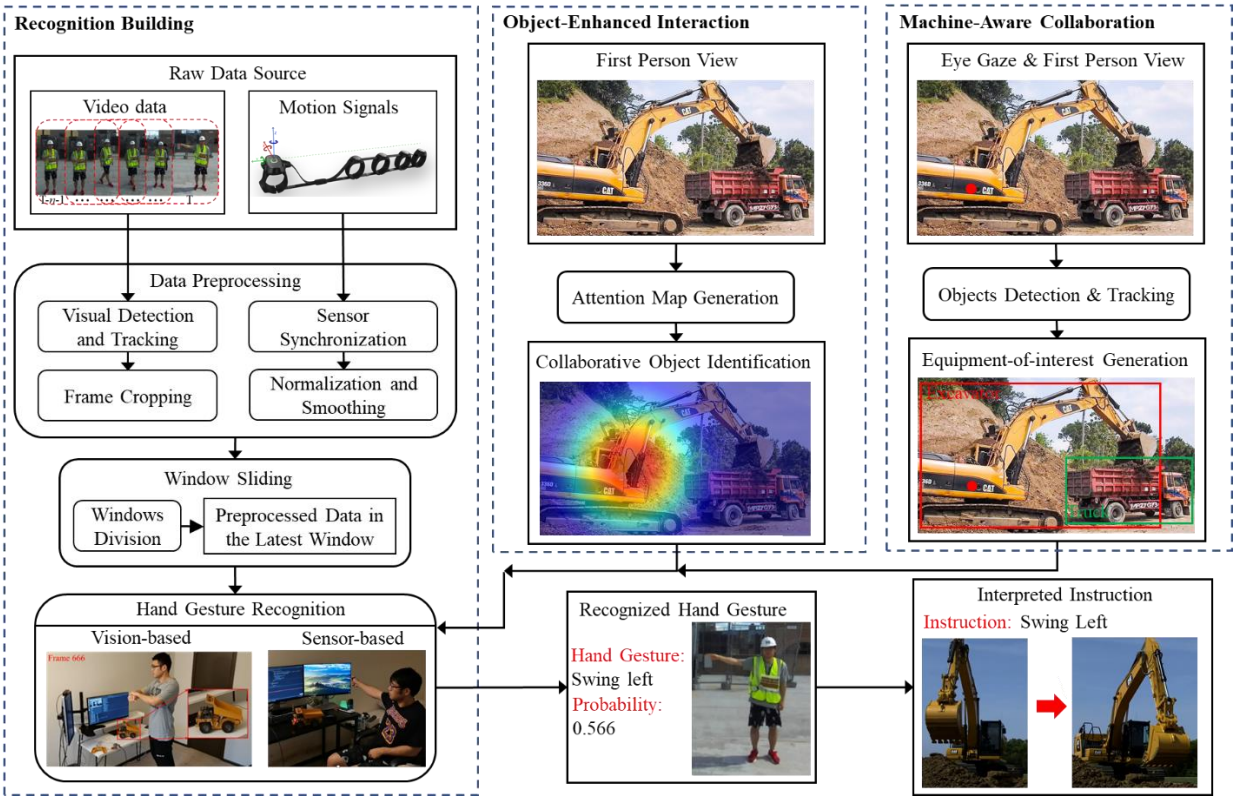


Figure 3.1 Overview of the proposed methodology

Figure 3.1 shows an overview of the proposed methodology. It consists of three components: recognition building, object-enhanced interaction and machine-aware collaboration. These first component is expected to answer the first question while the second and third components are devised to answer the second question. Specifically, a novel vision-based method and a novel wearable sensors-based method are developed to achieve hand gesture recognition for human-robot interactions in construction. Then, a novel object-aware method which integrates first-person vision and gesture recognition is proposed to improve the interaction efficiency.

Finally, a novel machine-aware method is established to indicate which machine the user intends to interact with and support one-to-many collaborations on construction sites. Experimental evaluation on different construction sites has demonstrated that the proposed methodology could capture and interpret worker intents effectively for human-robot collaborations in the construction field. The details of these components are discussed in the following chapters.

CHAPTER FOUR: RECOGNITION BUILDING FOR HAND GESTURES

The purpose of this chapter is to automatically capture and interpret the construction worker's hand gestures. First, a novel vision-based method is developed to achieve gesture recognition. Since computer vision technologies may be easily affected by the construction environment (e.g., diverse dust and light conditions), a novel wearable sensors-based method is then proposed. Finally, a comparison study is conducted to analyze the pros and cons of both vision-based and sensor-based methods.

Vision-Based Recognition of Construction Workers' Gestures

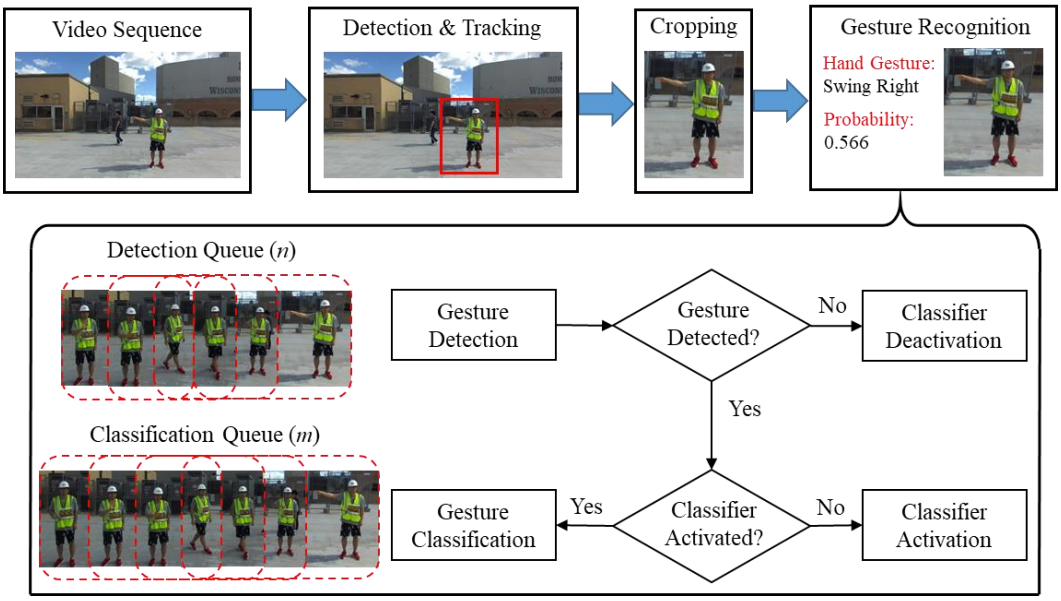


Figure 4.1 Overview of the vision-based method

Vision-Based Method

The overview of the vision-based method is illustrated in Figure 4.1. The method consists of three components: visual detection and tracking of workers, frame cropping for recognition queue, and hand gesture recognition. Specifically, the construction worker who gives hand gestures is visually detected and tracked in a camera video sequence to generate the regions of interest. Based on the detection and tracking results, the regions are then cropped to form hand

gesture recognition queues. Finally, a hierarchical architecture, which consists of a detector and a classifier, is employed to conduct the task of hand gesture detection and classification. More details are discussed in the following sections.

Visual Detection and Tracking of Construction Worker

The purpose of this component is to extract the construction worker who gives hand gestures in video sequences. A tracking-by-detection paradigm proposed in [123] is employed here due to its superior performance in tracking objects through long periods of occlusions. Within this paradigm, the detection module identifies the construction worker in each frame and obtains his/her bounding box. Given detection results, both trajectory and appearance information is modeled to associate current detections with existing tracks for the lifespan tracking of the worker. When there are multiple workers appearing in the scan, the construction worker who gives hand gestures could be identified through the tracking identification number (ID).

YOLOv3 [124] is selected to detect the construction worker because of its fast and accurate nature and ability to provide a multi-scale prediction. Additionally, many research results have verified the high performance of YOLOv3 in various construction object detections [125,126]. Multi-object deep Simple Online and Real-Time (SORT) tracker is employed to relate the same construction worker detected in the previous process across all the frames [123]. The deep SORT tracker is selected here since it is able to track the objects through long periods of occlusions and reduce the number of identity switches. Both trajectory and appearance information provided by the detection results are adopted to track the construction worker in video frames.

Frame Cropping for Recognition Queue

The purpose of this component is to crop the region of the construction worker who gives hand gestures from the original frame to form the queues for detecting and classifying hand

gestures. This component can be divided into two steps: the horizontal extension of the extracted region, and the formation of the hand gesture recognition queues. The extracted region is firstly expanded horizontally by 25% to fully capture the hand gestures made by the worker based on trials and errors. As shown in Figure 4.2 (left), the region which was directly obtained by the detection and tracking component might miss a part of the hand area when the worker was swinging his/her arms. After the horizontal extension both to the left and right (Figure 4.2 (right)), the region of the construction worker could capture the whole hand area, which is crucial for the recognition of hand gestures.

Figure 4.2 Region of construction worker before (left) and after (right) the horizontal extension

Further, the cropped frames are compiled to form the hand gesture detection and classification queues. Both detection and classification queues take the current frame as a basis. The detection queue includes *n* previous frames, while the classification queue consists of *m* previous frames. Following the guideline of [65], *n* is chosen as 8 frames since a smaller window size allows the detector to discover the start and end of the gestures more robustly. Besides, *m* is determined as 32 frames because the classification queue with 32 frames input achieves the best performance in [65]. The frames in the queues are further proportionally resized at a resolution of 112×112 pixels. After the formation of the recognition queues, they are input into the hand gesture recognition component.

Hand Gesture Recognition Based on a Hierarchical Architecture

The purpose of this component is to employ a hierarchical CNN architecture to detect and classify the hand gestures made by the worker. By introducing a time factor *L*, the hierarchical architecture would function in identifying a consecutive gesture made by the worker. Figure 4.3 shows the hierarchical architecture of hand gesture recognition, which consists of a detector and a classifier. The function of the detector is to detect whether there is a hand gesture in the detection queue. It is expected to be lightweight and run fast. This way, the video frames not containing any hand gesture could be discarded without being further processed. Here, the ResNet-10 model [65] is adopted since it requires small-sized features in each network layer.

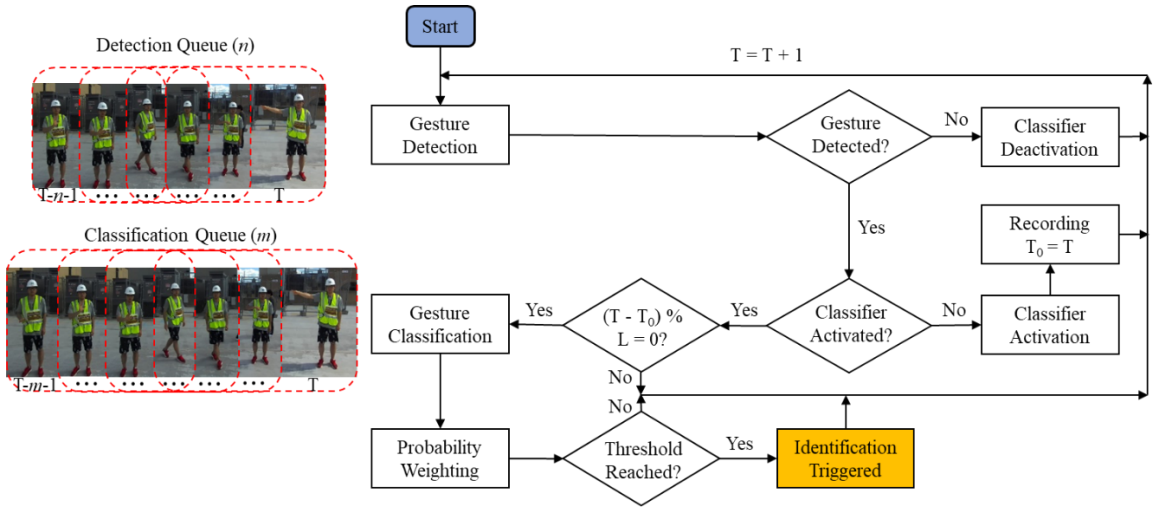


Figure 4.3 Hierarchical architecture of hand gesture recognition

When a hand gesture is detected, the classifier is utilized to classify its gesture class. The selection of the classifier should consider the balance between acceptable classification accuracy and fast reaction. Based on the findings from previous studies [65,79,121], ResNeXt-101 is selected. Table 4.1 summarized the specific network configurations of both ResNet-10 and ResNeXt-101.

Table 4.1 Networks of ResNet-10 and ResNeXt-101 (N_1, N_2, and F correspond to the number of ResNet blocks, ResNeXt blocks and feature channels, separately)

Layer Name	Conv1	Conv2_x	Conv3_x	Conv4_x	Conv5_x	---	Parameters
Output size	112×112	56×56	28×28	14×14	7×7	1×1	---
ResNet-10	Conv(3×7×7), Stride(1,2,2)	N_1: 1, F: 16	N_1: 1, F: 32	N_1: 1, F: 64	N_1: 1, F: 128	Average pooling, fc layer with Softmax	862K
ResNeXt-101		N_2: 3, N_2: 128	N_2: 24, F: 256	N_2: 36, F: 512	N_2: 3, F: 1024		47497K

The workflow of the hand gesture recognizing is illustrated in Figure 4.3. It combines the gesture detector and classifier. The detector acts as a switch to decide whether the classifier needs to be activated. If a gesture gets detected and the classier has not been activated yet, the classifier will be activated and record the current frame index T as T_0. It refers to the first frame index when a gesture gets detected. Then, for the subsequent video frames received later, the classification queue will be input into the classifier to calculate the raw probability of each type of the hand gesture, only if the detector keeps detecting a gesture and the difference between the current frame index T and T_0 equals to a multiple of the time factor L. A weight function (Equation 4.1) [65] is further applied to the raw classification probabilities to remove potential data noise.

$$w_T = \frac{1}{1+e^{-0.2\times(T-T_0-u/(4\times s))}} \tag{4.1}$$

where w_T refers to the weight at frame T, u corresponds to the mean duration of the gestures (i.e. the number of frames) in the dataset, and s is the stride length which can be determined as 1 to be small enough not to miss any gestures [65].

The difference between the highest and the second-highest weighted probabilities is calculated. If this difference is more than a threshold τ, the identification of the hand gesture will

be triggered; otherwise, it means that the classifier is not confident enough in classifying the hand gesture type. The architecture will conduct another round for the gesture detection and classification until the detector no longer detects the gesture and deactivates the classifier. It should be noted that the selection of τ and L depends on how likely and frequently the user intends to trigger the identification. Here, τ and L are chosen as 0.20 and 15 after trial and error.

Results

Dataset and Offline Training

To capture the characteristics of construction site environments, several factors were considered including weather/environmental conditions, motions in the background, the way to make hand gestures, etc. Following these factors, a total of 7 scenes have been designed when creating the dataset. Three of them are indoors and the other four are outdoors. The indoor scenes are created as follows. In the first indoor scene, the subject who makes hand gestures was requested to sit in a chair under a static but cluttered background. Then, the subject was requested to move when making hand gestures as the second scene. In the third indoor scene, the subject was moving and making hand gestures, and his or her background was cluttered with other moving persons. The outdoor scenes are classified into two categories: two of them are under sunny conditions and the other two are under cloudy conditions. The subjects in all these four scenes were moving and making hand gestures with or without background motions. Table 4.2 summarizes the characteristics of all the designed scenes mentioned above.

The hand gestures made by the subject in each scene are those commonly seen on construction sites. For example, tower cranes are the most frequently shared resources [127,128], which are mainly used for lifting heavy things and transporting them to other places. Hand gestures for directing tower crane operations were selected here. According to the American Society of

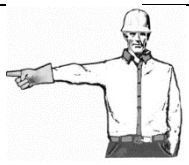
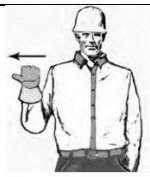
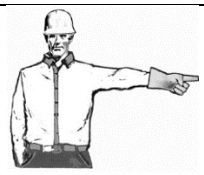
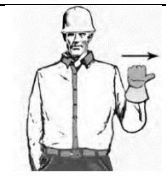
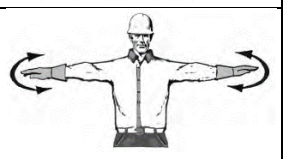
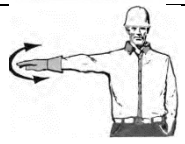
Mechanical Engineering (ASME) [129] and National Commission for the Certificate of Crane Operations (NCCCO) [130], there are 11 classes of hand gestures that can be used for signalman to instruct tower crane operations, as indicted in Table 4.3. In addition, the hand gestures in each scene were recorded in two modalities (RGB and depth) under a second-person view, where a camera is kept a short distance towards a subject and the subject is asked to perform hand gestures to interact intentionally with the camera.

Table 4.2 Characteristics of the designed scenes

No.	Scene	Subject status	Weather conditions	Background conditions
1	Indoor	Sitting on a chair	--	Static
2	Indoor	Moving	--	Static
3	Indoor	Moving	--	Dynamic
4	Outdoor	Moving	Sunny	Static
5	Outdoor	Moving	Sunny	Dynamic
6	Outdoor	Moving	Cloudy	Static
7	Outdoor	Moving	Cloudy	Dynamic

To create the dataset, a ZED 2 stereo camera [131] is selected as a recording device. The camera could capture the video clips under the RGB and depth modalities. The maximum resolution of the videos could reach up to 2208 × 1242 pixels at 15 frames per second (fps). When capturing the hand gestures into a video clip, a gesture list with a random selection of 5 gestures is generated first. Then, the subject was asked to continuously perform these gestures in the list at different locations to make sure the gestures appear in different video regions.

Table 4.3 Hand gestures for instructing tower crane operations adapted from [129,130]

No.	Hand signal	Examples	No.	Hand signal	Examples
1	Hoist		7	Dog everything	
2	Lower		8	Move slowly	
3	Tower travel		9	Swing right	
4	Trolley travel right		10	Swing left	
5	Trolley travel left		11	Emergency stop	
6	Stop				

A total of 364 RGB-D video clips were collected which are equivalent to more than 426,602 frames in each modality. Among them, there are 1820 gesture samples which are distributed in 7 scenes. Each hand signal category consists of 165 samples on average. The average length of a gesture is 110 frames. The minimum and maximum gesture lengths are 21 and 322 frames separately. The details of the dataset are listed in Table 4.4. The created dataset contains the video clips which are collected from real construction sites. Examples of the collected data

could be found in Figure 4.4. The start and end frame indices of the subject's gesture in each video clip are manually labeled as shown in Figure 4.5.

Table 4.4 Data statistics of the dataset

		Scene 1	Scene 2	Scene 3	Scene 4	Scene 5	Scene 6	Scene 7
Duration (s)		2,170.7	3,909.1	3,985.7	4,291.1	4,797.1	4,383.8	4,902.7
# of total frames		32,560	58,637	59,785	64,366	71,956	65,757	73,541
Gesture category (# of samples / # of frames)	Hoist	23/1,713	22/2,427	22/2,836	26/2,512	23/2,916	24/2,874	23/3,170
	Lower	21/1,415	22/2,354	22/2,741	24/2,485	23/2,939	24/2,617	25/3,103
	Tower travel	25/1,561	22/2,334	23/2,706	24/2,316	23/2,891	24/2,277	24/2,988
	Trolley travel right	22/1,426	23/2,649	22/2,645	24/2,524	23/3,057	25/2,525	25/3,432
	Trolley travel left	24/1,880	23/2,732	23/2,681	26/2,747	24/3,154	23/2,676	25/3,169
	Stop	23/1,483	23/2,451	23/2,519	24/2,553	26/2,916	24/2,704	24/3,058
	Dog everything	24/1,433	23/2,460	23/2,624	26/2,675	25/3,314	24/3,129	22/2,881
	Move slowly	21/1,622	23/2,519	23/2,619	22/2,374	25/3,248	25/2,540	24/3,022
	Swing right	23/1,373	23/2,671	23/2,712	24/2,811	25/3,083	27/2,800	24/3,584
	Swing left	22/1,458	23/2,511	23/2,845	25/2,638	24/2,858	25/2,847	24/2,997
	Emergency stop	22/2,018	23/2,464	23/2,799	25/2,851	24/3,509	25/2,773	25/3,501

Figure 4.4 Examples of the collected data (top: RGB; bottom: depth)

Figure 4.5 Manual labeling and cropping process

The above dataset was employed to conduct the offline training for hand gesture recognition. In order to train and test the detector, the gesture and non-gesture samples in the dataset were randomly split into the training subset (60%), validation subset (20%) and testing subset (20%). As for the training and testing of the classifier, only the gesture samples in the dataset were randomly divided into the training subset (60%), validation subset (20%) and testing subset (20%).

Table 4.5 summarized the parameters set for the training of the detector and classifier. The network configuration of the classifier is much more complicated than the detector, which typically requires more training data to prevent underfitting. Here, the transfer learning strategy was adopted for the classifier. The classifier is pre-trained firstly using the Jester dataset [132], which is the largest hand gesture dataset publicly available. Then, the specific training process for the detector and classifier is conducted as follows. The learning rate and the batch size are initially set as large as possible. The cross-entropy loss (Equation 4.2) is employed as the loss function.

$$L_{CE} = -\sum_{i=1}^{n} t_i \, \boldsymbol{log}\left(p_i\right) \tag{4.2}$$

where n is the number of classes, t_i is the truth label and p_i is the Softmax probability for the i-th class. When the training loss is steady, the learning rate is reduced with a fixed decay factor. Stochastic gradient descent (SGD) with Nesterov momentum of 0.9, damping factor of 0.9, and weight decay of 0.001 is employed as the optimizer. Moreover, all images of hand gesture samples

are randomly cropped with a spatial size of 112×112 as the inputs for the data augmentation purpose. Figure 4.6 shows the loss reduction along with the training progress for the detector and classifier, respectively. After training, the detector achieves a classification accuracy of 91.1%. For the classifier, the classification accuracy is 91.9%.

Table 4.5 Parameter settings of the detector and classifier

Components	Networks	Learning rate	Step size of learning rate decay	Batch size	Length of input frames
Detector	ResNet-10	0.01	10	8	8
Classifier	ResNeXt-101	0.01	15	20	32

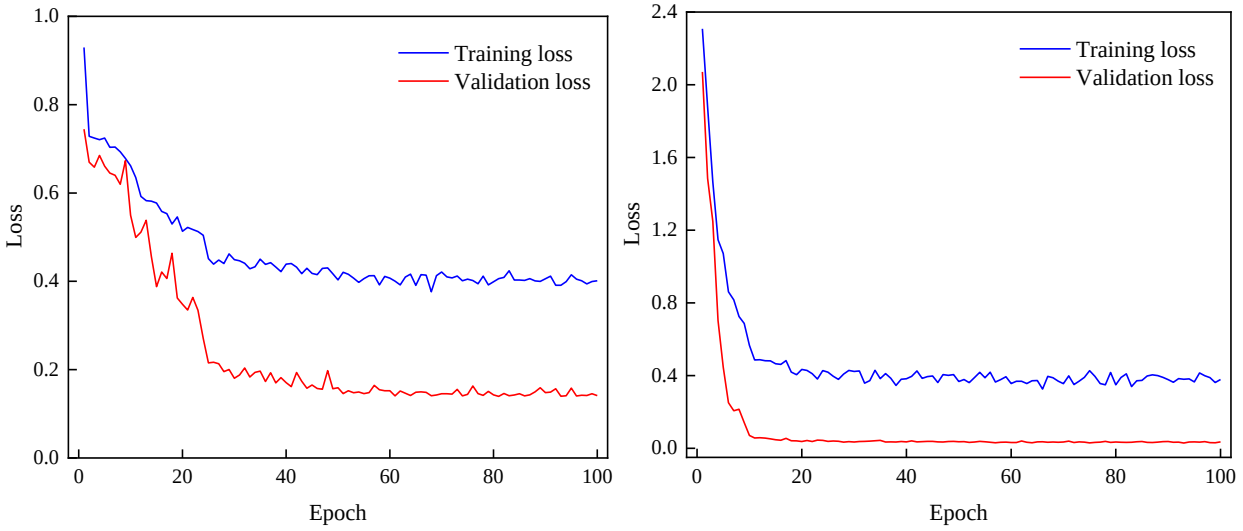


Figure 4.6 Loss reduction along with the training progress for the detector (left) and classifier (right)

Field Experiments

The effectiveness of the vision-based method was tested through field experiments. The focus of the field experiment was placed on testing whether the method could detect and track workers and capture and interpret their hand gestures on construction sites. A construction site near Milwaukee, WI. was selected for this field experiment. A ZED 2 stereo camera was set up on

the site to record the hand gestures made by construction workers. Six video clips were collected, which included 30 gesture samples in total.

Figures 4.7 and 4.8 showed two examples of testing the proposed method to detect and track the workers (i.e. Subject 1 and 2) and then identify their hand gestures, e.g. “swing right” and “emergency stop”. In each test, the detection and tracking results were represented within a series of bounding boxes along the video sequence. As shown in Figure 4.9, when there are multiple workers in the scan, only the bounding box of the worker who makes hand gestures is returned based on his tracking ID. When the worker performed a hand gesture, the proposed method triggered the gesture identification module and reported the corresponding gesture type.

Compared with the worker detection and tracking, the recognition of the worker’s hand gestures in construction has not been widely tested and evaluated before [79]. For this reason, the gesture identification component in the proposed method was specifically evaluated here. Two quantitative indicators, i.e. precision and recall, were adopted and their definitions were given in Equations 4.3 and 4.4 [133].

$$\text{Precision}_i = \frac{TP_i}{TP_i + FP_i} \tag{4.3}$$

$$\text{Recall}_i = \frac{TP_i}{TP_i + FN_i} \tag{4.4}$$

where i is the gesture class, TP_i is the number of gesture samples in gesture class i which are correctly predicted as i, FP_i is the number of non-gesture and gesture samples which are falsely predicted as i, and FN_i is the number of gesture samples in gesture class i which are falsely predicted as non-gesture or other gesture classes.

Figure 4.7 Examples of the field test results for subject 1

Figure 4.8 Examples of the field test results for subject 2

Figure 4.9 Identification of the signalman-of-interest

Table 4.6 compiled the recognition performance of the hand gesture under each type using the videos collected from the site. It was found that the overall precision and recall achieved 87.0% and 66.7%, respectively. The identification of "lower", "tower travel" and "dog everything" could reach up to 100% precision and 100% recall. The lowest precision and recall happened on the identification of "trolley travel right" and "swing left".

Table 4.6 Gesture recognition performance for vison-based method

Gestures	Hoist	Lower	Tower travel	Trolley travel right	Trolley travel left
Precision (%)	100.0	100.0	100.0	50.0	100.0
Recall (%)	66.7	100.0	100.0	33.3	66.7
Gestures	Dog everything	Swing right	Swing left	Emergency stop	Overall
Precision (%)	100.0	100.0	50.0	75.0	87.0
Recall (%)	100.0	71.4	25.0	75.0	66.7

Pilot Study

Further, a pilot study was conducted in a laboratory environment to test whether the vision-based method could serve as an interface to help workers control and/or interact with construction machines. In the study, the video frames were captured by the camera in real time. Each captured frame will be input into the method immediately. Figure 4.10 illustrated the setup of the laboratory experiment and the related data flow. Specifically, a subject was asked to perform hand gestures, which were captured by a video camera connecting to a computer. The captured hand gestures would be fed into the method and processed there in real time. Based on the recognition results, the corresponding instructions would be sent to a remote controller, where the control signals would be transmitted to operate the truck model remotely.

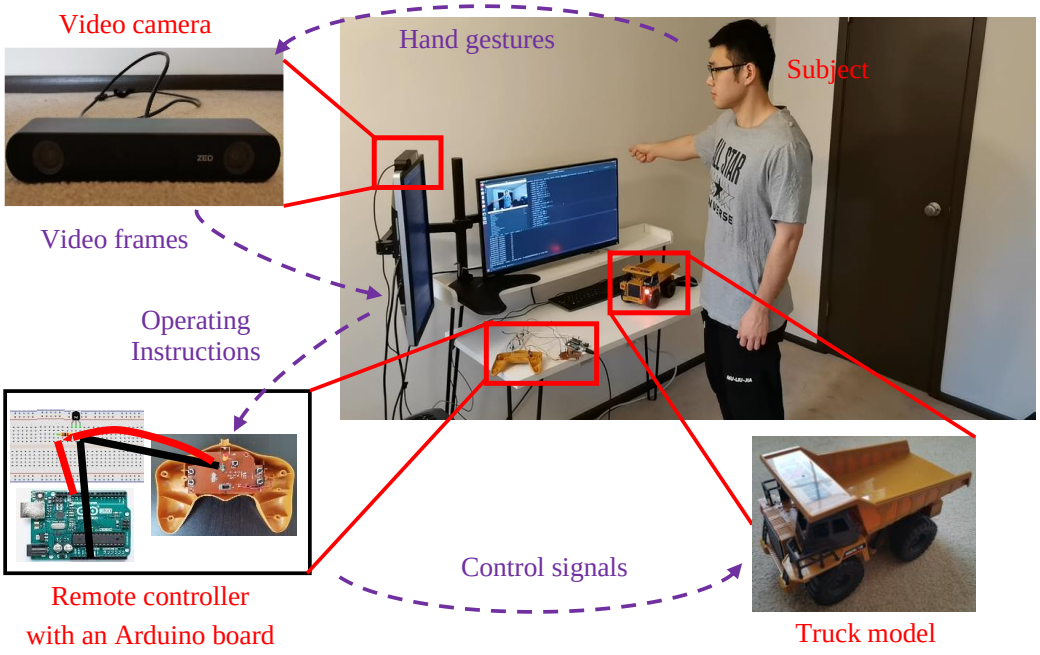


Figure 4.10 Pilot study setup and data flow for vision-based method

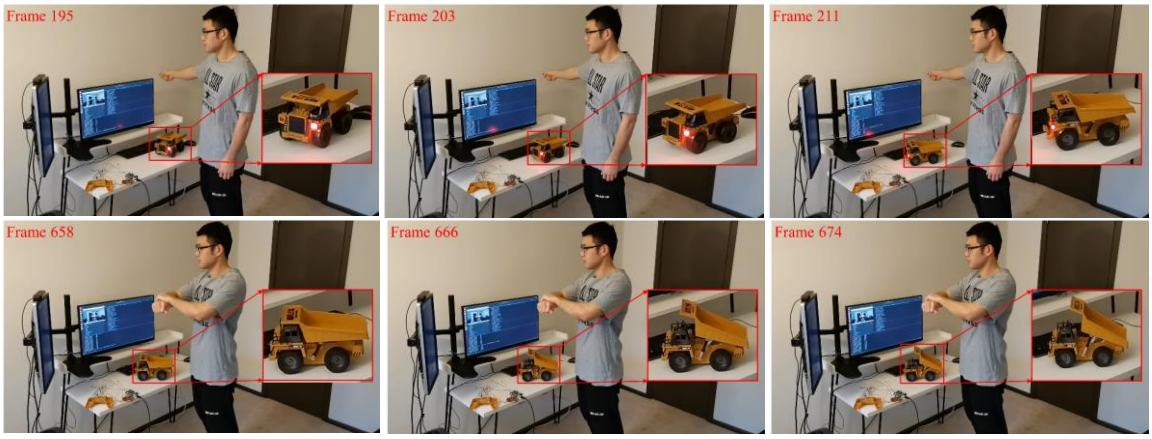


Figure 4.11 Examples of the test results in the laboratory environment

Figure 4.11 showed an example of using the proposed method to remotely control a toy truck to move and lift its dump box. The subject firstly made the hand gesture of "swing right" to request the truck model to turn right. The gesture was captured by the method and the corresponding instruction was sent to the truck model through the remote controller in 1.5 ms. Following the instruction, the truck model drove towards the right gradually. After a short pause,

the subject then performed the gesture of "dog everything" to request the truck model to lift its dump box. The truck model received the corresponding instruction in 1.4 ms and then lifted its dump box.

Wearable Sensors-Based Recognition of Construction Workers' Gestures

Sensor-Based Method

The proposed sensor-based method comprises three modules as illustrated in Figure 4.12. The raw signals coming from wearable sensors are fed into the Data Preprocessing module for synchronization, normalization and smoothness. Then, a window with a fixed window length slides through all the preprocessed signals and the preprocessed data of the latest window are fed into an enhanced FCN classifier to achieve hand gesture recognition.

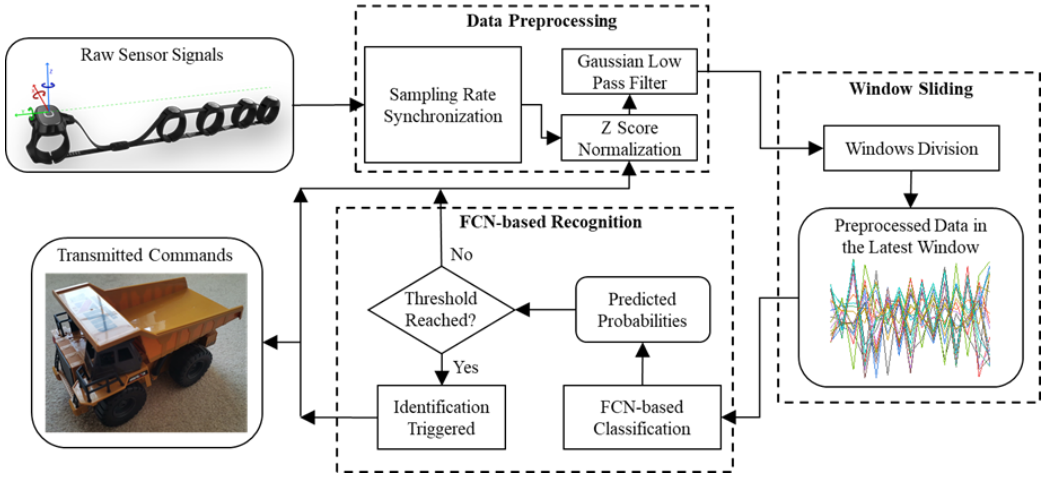


Figure 4.12 Overview of the wearable sensors-based method

Data Preprocessing

In this study, the accelerometer and gyroscope signals are captured directly from the wearable sensors as raw data. The accelerometer signals are used to measure the vibration or acceleration of hand/finger movements while the gyroscope signals refer to the rotational motions of hands/fingers. Several techniques are employed to preprocess the raw data. First, the accelerometer and gyroscope sensors are calibrated to a unified sampling rate for synchronization

since the original sampling rates may be different for various sensors. An optimal sampling rate can be defined as the minimum sampling rate at which all relevant characteristics of activities that are of interest can be captured [134]. Previous research [135] suggests that a sampling rate of approximately 20 Hz is reasonable for standard human activities (e.g., running and cycling). Therefore, 20 Hz is selected here considering the balance between capturing the details of gestures and decreasing the computational cost. Then, the mean and standard deviation of the raw signals are computed from the raw data. Z score normalization (Equation 4.5) is utilized to rescale the data, which allows all the signal channels to be considered with equal importance.

$$z_i = \frac{x_i - \mu_i}{\sigma_i} \tag{4.5}$$

where z_i is the standard score for *i*-th signal channel, x_i refers to the raw data for *i*-th signal channel, μ_i and σ_i are the mean and standard deviation of *i*-th signal channel, respectively. Besides, the signals coming from sensors may be affected by external data noise. For example, the noise caused by irrelevant human walking would make the signals vibrate more sharply. To reduce the impact of the noise, a Gaussian low pass filter algorithm is applied to attenuate high frequency points. After smoothing, the signals would become subdued and get closer to those without walking noise.

Window Sliding

To recognize consecutive gestures made by workers, a sliding window approach is employed for continuous classification of hand gestures. Specifically, the sliding window is used to divide the data as shown in Figure 4.13. The window length is set to 3 seconds. The stride size of the two consecutive sliding windows is set to 1 second in order to avoid missing any gestures. With the signals coming in continuously, the window moves through the incoming signals and the preprocessed data in the latest window are fed into an enhanced FCN classifier to achieve hand gesture recognition.

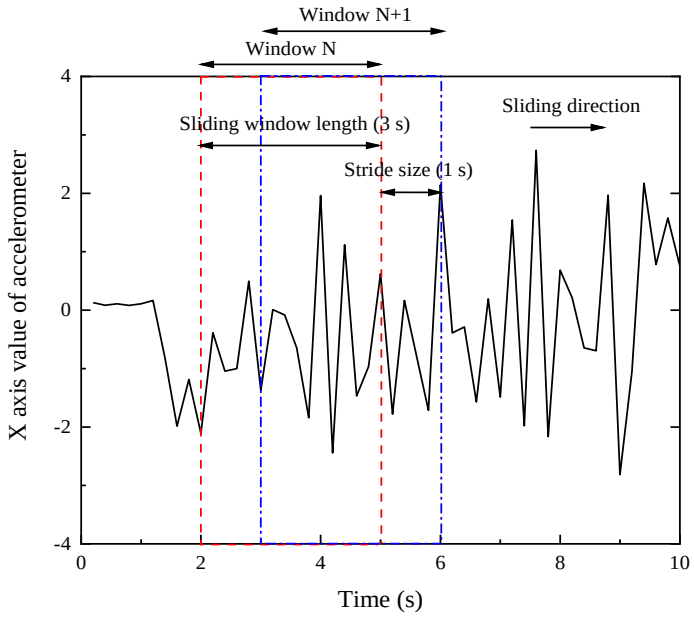


Figure 4.13 Extraction of signal segments by the sliding window approach

FCN-based Hand Gesture Recognition

The preprocessed data of the sliding window are fed into an enhanced FCN to extract features through deep learning and achieve hand gesture recognition. The FCN is selected as the base model since FCN classifiers exhibit superiority on multivariate time series classification tasks based on motion sensor data [136,137] compared to other deep learning networks (e.g., Encoder, Time Le-Net, Time-CNN). In our enhanced FCN classifier, the basic block is a convolutional layer followed by a batch normalization layer, a ReLU activation layer and a dropout layer. The length of the time series data is kept unchanged throughout the convolutions, which helps preserve data characteristics of different stages in a dynamic gesture. It also applies batch normalization to speed up the convergence speed and includes dropout which is a regularization technique to help improve generalization. Here, three blocks are employed to extract the representative features of the time series data. The third convolutional layer is fed to an attention mechanism [138] that enables the network to learn which parts of the time series data are important for a certain classification.

Subsequently, the features are fed into a Global Max Pooling (GMP) layer, which largely reduces the number of weights. Finally, a traditional Softmax classifier is fully connected to the GMP layer's output for the prediction label. Figure 4.14 shows an overview of the enhanced FCN architecture.

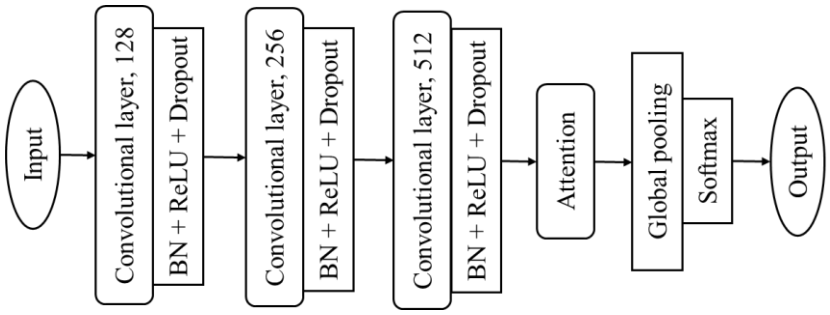


Figure 4.14 Architecture of enhanced FCN

There are three significant ways by which the enhanced FCN differs from the previously proposed FCN [136]. First, the dropout regularization technique is added to improve the generalization of the model. Second, the attention layer is used to connect the convolutional layers and the GMP layer. The attention mechanism helps the network to memorize long sequences of information, which is useful for learning which parts of the time series data are important for a certain classification. Third, the GMP layer is employed to replace the Global Average Pooling (GAP) layer to retain the most prominent features.

The triggering for the gesture recognition is conducted as follows. The preprocessed data in the sliding window are first fed into the enhanced FCN classifier to calculate the probabilities for each class of hand gesture. If the highest probability is more than a threshold δ, the identification of the hand gesture will be triggered; otherwise, it means that the classifier is not confident enough to classify the hand gesture class. Then, the data of the next sliding window is collected and input into the classifier for recognition until the user terminates the prediction process. It should be noted that the selection of δ depends on how much confidence the classifier

should have to make a classification decision. Here, δ is selected as 0.70 based on the average confidence for making correct predictions on an offline test set.

Results

Dataset and Offline Training

Tap Strap 2 [139], which is a IMU-based sensor, is employed as the wearable sensor in this study. This sensor was chosen since IMU-based sensors have a high sensitivity to dynamic hand gestures [85,86,140] and can keep the signals relatively stable when different subjects perform gestures [140]. Moreover, compared to other IMU-based sensors, the Tap sensor is portable, lightweight and easy to wear on the fingers as shown in Figure 4.15. The Tap sensor can transmit the raw signal data to the computer over a Bluetooth connection. It includes five 3-axis accelerometers and one IMU (3-axis accelerometer + 3-axis gyroscope). The five accelerometers are located at five fingers, separately, while the IMU is placed on the thumb. There are totally 21 signal channels from the Tap sensor.

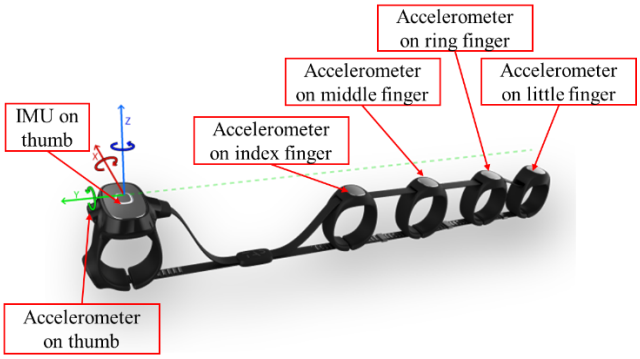


Figure 4.15 Structure of Tap Strap 2

Eight classes of hand gestures for directing tower crane operations are selected here since they are commonly observed at construction sites [79]. The gestures include “hoist”, “lower”, “tower travel”, “trolley travel left”, “trolley travel right”, “stop”, “swing left” and “swing right”. The classifier in the proposed method is trained and tested using these hand gestures. The use of

the hand gestures for directing tower crane operations will not limit the capability of the method to capture and interpret other classes of hand gestures. Instead, the method could be expanded without any loss of generality. This is because the hand gestures selected here fully represent the characteristics of other classes of construction hand gestures which generally involve the swinging of both hands and arms.

During the data collection, the subject was first requested to wear the Tap sensor on his/her hand. To capture the characteristics of the hand gestures in construction site environments, the subject was asked to move and make hand gestures simultaneously. Each class of gesture was performed 20 times by the subject, resulting in 160 (20 × 8) gesture samples in the dataset. The duration for each gesture is 10 seconds, which means the capture of one gesture generates 200 signals per channel. Examples of the collected data for one gesture are shown in Figure 4.16. The boxplot of the dataset is indicated in Figure 4.17. The average signal values for these 8 classes of gestures are -0.043, -0.873, 0.200, -0.069, 0.148, 0.042, 0.233 and 0.361, respectively. The gestures of “hoist”, “lower” and “tower travel” have a larger value range since they generally involve more drastic movements of hands/fingers.

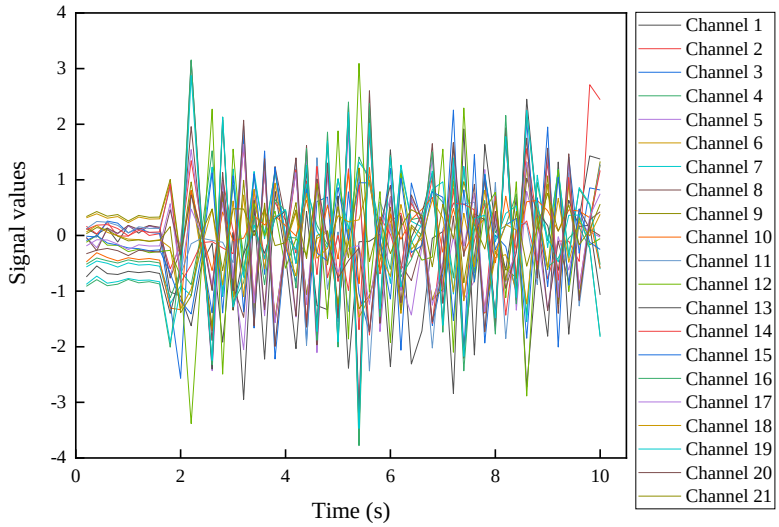


Figure 4.16 Examples of the signal values for one gesture

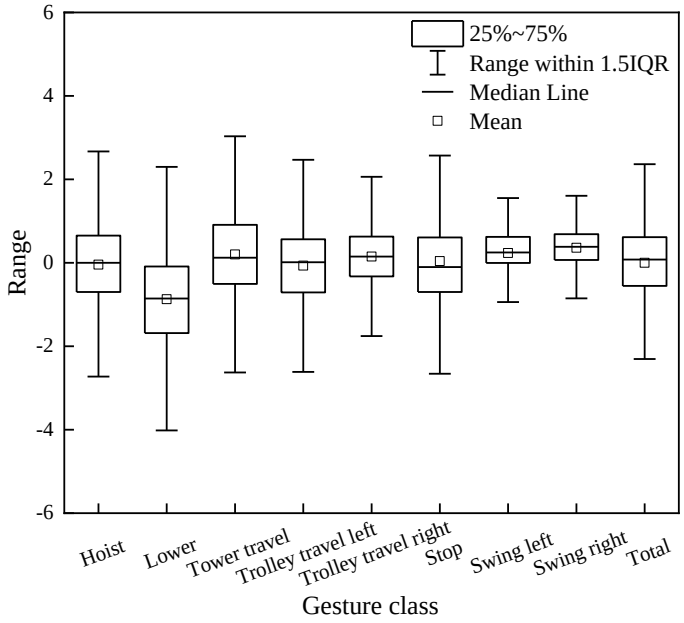


Figure 4.17 Boxplot of the dataset for sensor-based method

In order to train and test the enhanced FCN classifier, the dataset is randomly split into the training subset (80%) and validation subset (20%). The training subset includes 128 gesture samples for the training of the network parameters in the classifier. The validation subset includes 32 gesture samples which were used to validate the recognition performance.

The specific training process is conducted as follows. The learning rate (0.001) and the batch size (16) are initially set as large as possible. The cross-entropy loss (Equation 4.2) is employed as the loss function. When the training loss is steady, the learning rate is reduced with a fixed decay factor which is set to 10. Adam is employed as the optimizer. Figure 4.18 shows the loss reduction along with the training progress. The training and validation loss tend to be stable when the epoch reached about 50. The training was completed after 200 epochs and achieved the best validation performance at epoch 163. The enhanced FCN classifier achieves a precision of 96.9% and a recall of 96.9% on the validation subset.

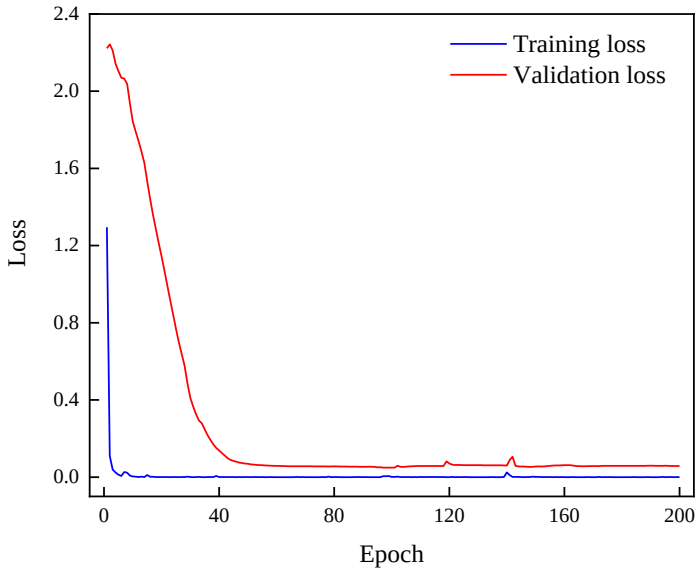


Figure 4.18 Loss reduction along with training progress for sensor-based method

Method Validation Test

The effectiveness of the proposed method was tested through a method validation test. The test setting was established similar to [141]. Specifically, in each session, the subject was asked to continuously move and perform 4 hand gestures wearing the Tap sensor. After each session, the subject lowered his/her arms naturally and relaxed for 5 seconds to reduce muscle fatigue. Data from eight testing sessions were collected, which included 32 gesture samples in total. The collected data were fed into the method to investigate whether and how well the hand gestures could be automatically captured and interpreted.

Figure 4.19 shows an example of testing the proposed method to conduct the hand gesture identification (e.g., “hoist” and “lower”). When the subject performed a hand gesture, the proposed method triggered the gesture identification module and reported the corresponding gesture class. No gesture was recognized when the subject was just randomly walking since the enhanced FCN classifier does not have enough confidence to make a classification decision. Table 4.7 shows the recognition performance of the hand gesture under each class using all the sessions. The overall

precision and recall achieved was 85.7% and 93.8%, respectively. The identification of “hoist”, “stop” and “swing right” reached up to 100% precision and 100% recall. The lowest precision and recall happened on the identification of “trolley travel left” and “lower”, separately. This may have happened because the movements of these gestures are similar to other gestures. To improve the recognition performance, multi-modality and multi-location sensor fusion can provide possible solutions since they have been proven effective to reduce uncertainty and enhance the reliability of classifiers [142].

Figure 4.19 An example of the validation test for sensor-based method

Table 4.7 Gesture recognition performance for sensor-based method

Gestures	Hoist	Lower	Tower travel	Trolley travel left	Trolley travel right
Precision (%)	100.0	100.0	80.0	60.0	80.0
Recall (%)	100.0	75.0	100.0	75.0	100.0
Gestures	Stop	Swing left	Swing right	Overall	
Precision (%)	100.0	80.0	100.0	85.7	
Recall (%)	100.0	100.0	100.0	93.8	

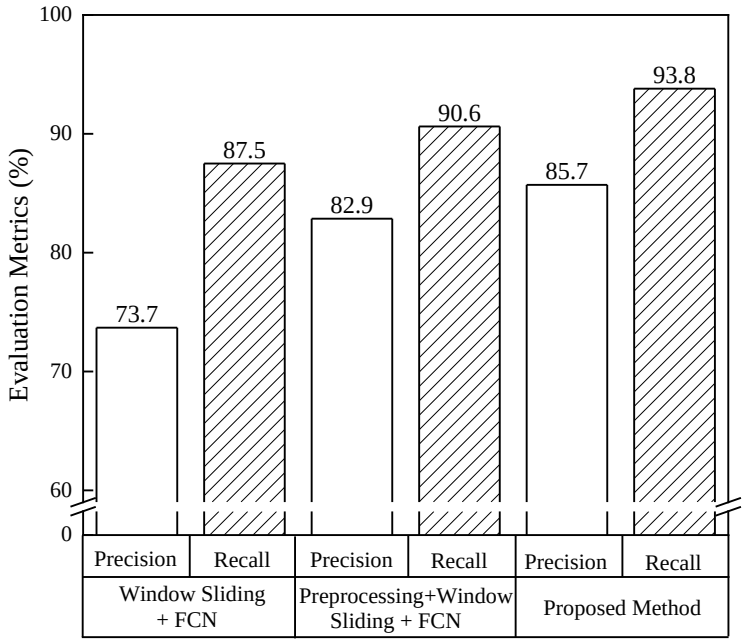


Figure 4.20 Comparison between FCN [136] **+ window sliding, FCN** [136] **+ preprocessing + window sliding and proposed method**

Further, to validate the effectiveness of the proposed method, a comparison among FCN [136] + window sliding, FCN [136] + preprocessing + window sliding and the proposed method with the enhanced FCN is carried out. For FCN [136] + window sliding, the inputs are raw signals coming from the sensors after synchronization and the classifier is the unmodified FCN structure. The only difference between FCN [136] + preprocessing + window sliding and the proposed method is that the previous method still employs the unmodified FCN as the classifier. Figure 4.20 shows the comparison results on the method validation test. The FCN + window sliding achieved precision and recall of 73.7% and 87.5%, respectively. In contrast, FCN + preprocessing + window sliding achieved 9.2% and 3.1% higher performance of precision and recall, respectively. The significantly lower precision of the FCN + window sliding method shows that non-gesture samples affected by the signal noise are easier to be classified by the FCN + window sliding method as gesture samples. These results validate that the Data Preprocessing module is effective in reducing the noise coming from irrelevant movements of the worker. In addition, our proposed method with

the enhanced FCN achieved 2.8% and 3.2% higher performance of precision and recall, respectively, compared to FCN + preprocessing + window sliding. These results validate the effectiveness and superiority of the enhancements that we have made to the FCN structure.

Pilot Study

A pilot study was conducted in a laboratory environment to test whether the proposed method could serve as an effective interface for workers to control and/or interact with robotic construction machines. Figure 4.21 illustrates the setup of the laboratory experiment and the related data flow. The subject was asked to perform hand gestures, which were captured by a Tap sensor connected to a computer. The motion sensory data obtained by the Tap sensor were input into the system and processed in real time. Based on the recognition results, the corresponding instructions were sent to a remote controller, where the control signals were transmitted to operate the truck model remotely.

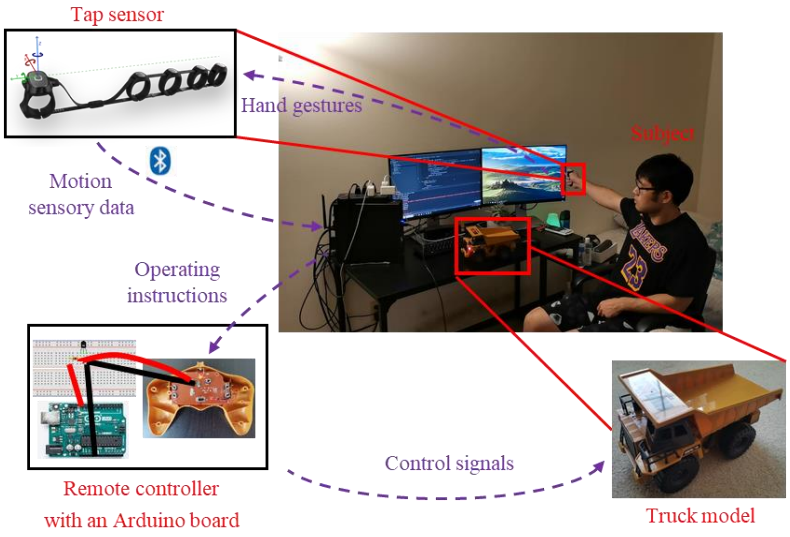


Figure 4.21 Pilot study setup and data flow for sensor-based method

Figure 4.22 shows an example of using the proposed system to remotely control a model truck to move and lift its dump box. The subject first made the hand gesture of “swing left” to

request the model truck to turn left. The gesture was captured by the system 57 frames (approximately 1.9 seconds) later, after the start of hand gesture. The corresponding instruction was sent to the truck model through the remote controller in 1.3 milliseconds. Following the instruction, the truck model drove towards the left gradually. After a short pause, the subject then performed the gesture of "hoist" to request the truck model to lift its dump box. The gesture was captured by the system 62 frames (approximately 2.1 seconds) later, after the start of the hand gesture. Then, the truck model received the corresponding instruction in 1.4 milliseconds and lifted its dump box.

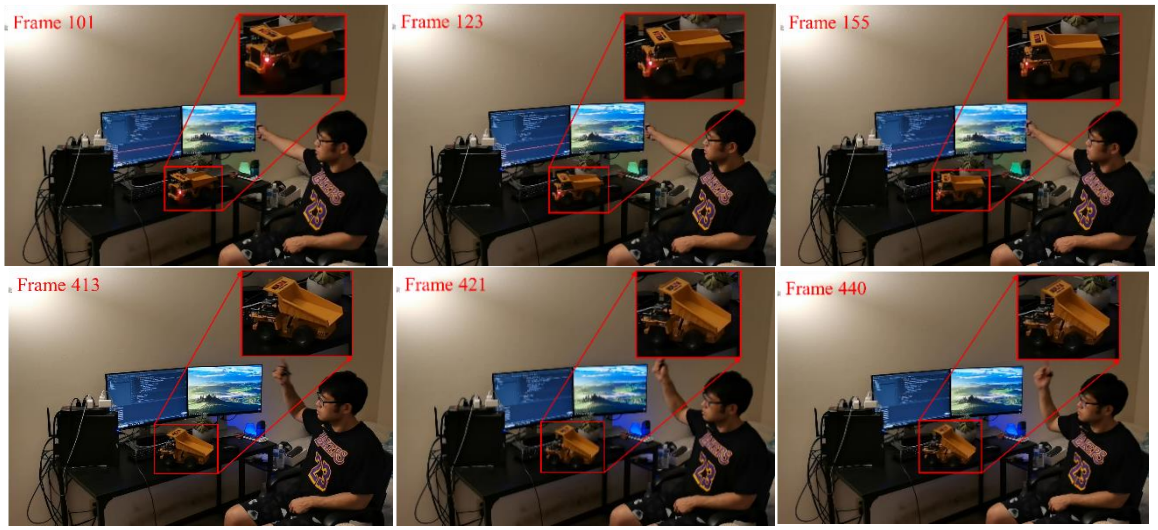


Figure 4.22 Examples of the sensor-based test results

Comparison Study

Finally, a comparison was conducted to investigate the benefits and limitations of performing the hand gesture recognition tasks in construction using visual versus motion sensory data. The vision-based and sensor-based methods were compared in both quantitative and qualitive aspects. The quantitative comparison was evaluated in terms of the precision and recall drops on the new unseen test data, as well as the triggering ratio. The triggering ratio is defined as the ratio of the recognition triggering time after the start of the gesture to the data processing time of a 10

seconds gesture. The precision and recall of the sensor-based method decreased by 9.4% and 9.4%, respectively. In contrast, the recognition precision and recall of the vision-based method decreased by 14.5% and 8.0%, respectively. The triggering ratios of the sensor-based and vision-based methods are 21.3% and 40.7%, respectively.

Several lessons were learned from the comparison results. First, the sensor-based hand gesture classifier had a performance drop on new test data similar to that of the vision-based classifier. The decrease in precision and recall performance on new data is expected. Extant research shows that most recognition models do not reach their original accuracy scores when being tested with new data due to the generalization issue. The accuracy reduction ranged from 11% to 14% or 3% to 15% depending on the datasets adopted [143]. Based on this experience, the overall precision and recall drops of the vision-based and sensor-based methods are in an acceptable range.

Second, the sensor-based classifier always triggered the hand gesture recognition earlier than the vision-based one. The vision-based classifier resulted in a triggering ratio of 40.7%. The delayed recognition may be due to the fact that the vision-based classifier cannot make a reliable classification decision until the gesture stays in its nucleus part [80]. In this study, the triggering ratio based on the fingers' accelerations and rotational motions reached 21.3%. It may be because wearable sensors can capture the nucleus part of a gesture in an earlier period. The early capture of the nucleus part of a hand gesture is expected to increase the human-machine interaction efficiency especially when an urgent hand gesture, such as "stop" needs to be recognized as soon as possible.

In addition, the motion sensory data were robust to various environmental noises such as self-occlusions, background noise or illumination variations. Considering that the outdoor

environments on construction sites are complex due to weather conditions and cluttered with tools, materials, machines, etc., such noise can impact the video or RGB-D data quality more easily [79,122]. In contrast, wearable sensors are usually robust to these interferences and not limited by scene or time [67]. These characteristics enable them to capture the nature of hand gestures in complicated outdoor environments.

It should be noted the non-contact nature of vision-based methods could eliminate the cost of additional sensors and alleviate the communication load. The vision-based methods only need cameras which are typically equipped on the intelligent construction robotic machine. Under this scenario, the machine can identify the worker through a detection and tracking module [79] and observe the worker behaviors. This could eliminate the cost of wearable sensors for the workers and save the data transition load from worker to machine. In contrast, each worker needs to wear a set of sensors to interact with a construction robotic machine for the sensor-based method. Therefore, multiple wearable devices are needed. Further, a one-to-one communication strategy needs to be established, which significantly increases the communication cost and complexity between workers and the intelligent machine.

CHAPTER FIVE: OBJECT-ENHANCED HUMAN-ROBOT INTERACTION

The purpose of this chapter is to incorporate the collaborative object information to improve the gesture recognition performance. First, a two-stream network is designed to integrate the contextual spatiotemporal regions of first-person vision into the gesture interface, which could improve the interaction efficiency for human-robot collaboration. Then, the proposed method is validated by a dataset collected on five different construction sites. Finally, a comparison study is conducted to show the efficiency improvement of the proposed method over single sensor-based method without object information.

Object-Aware Human-Robot Interaction Method

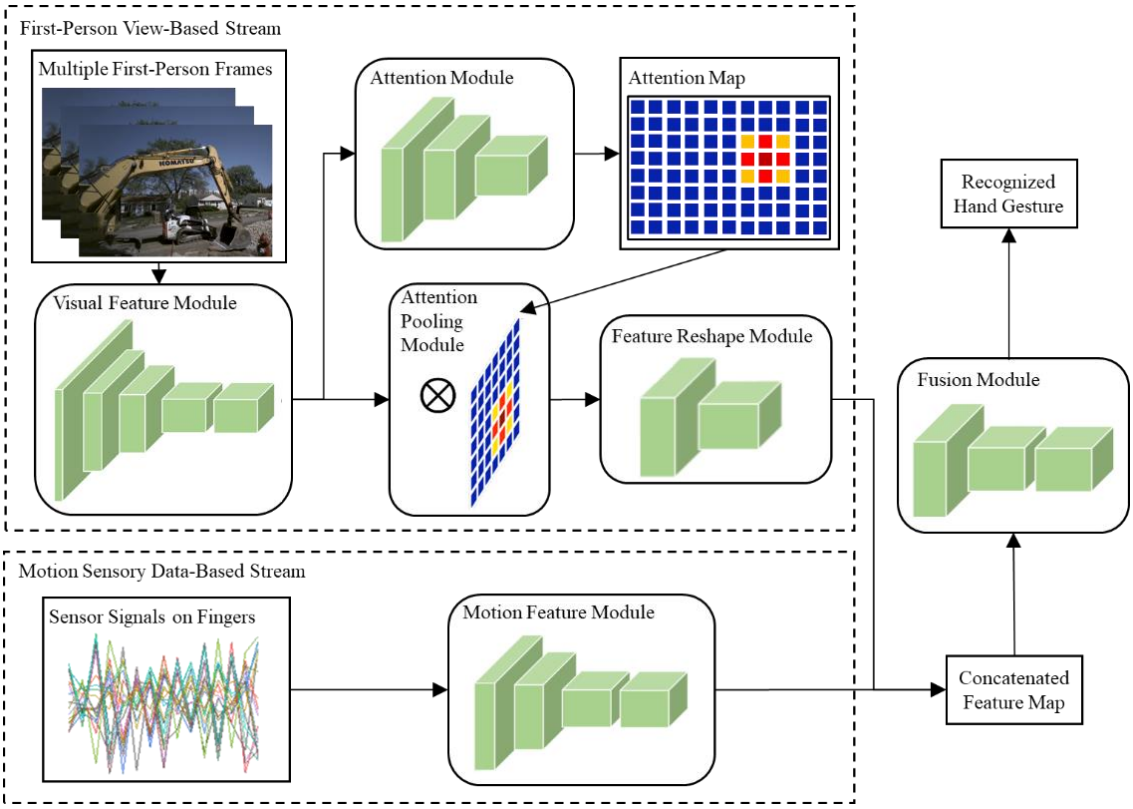


Figure 5.1 Overview of two-stream architecture

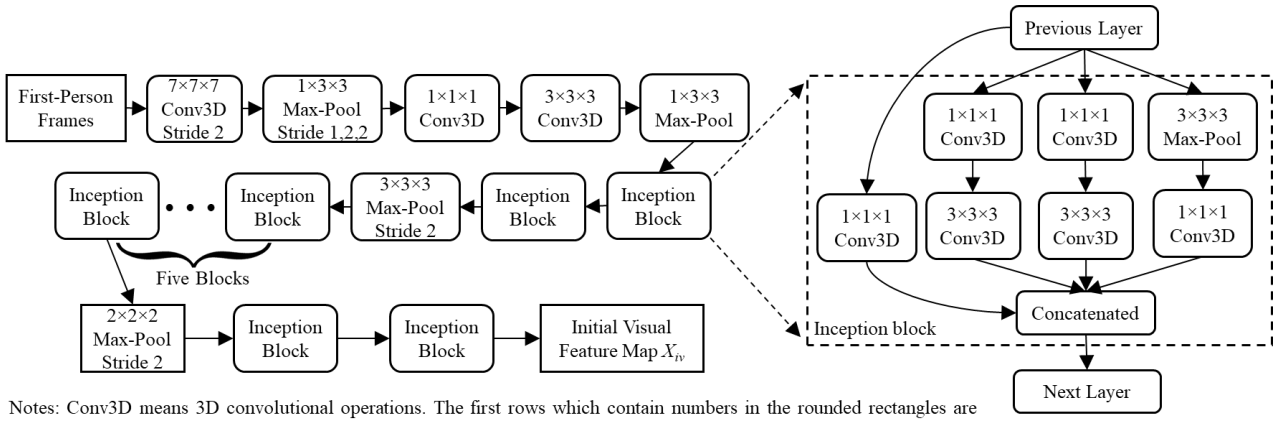
In this section, a novel object-aware method is proposed for human-robot collaboration in construction, integrating first-person vision and gesture recognition. To achieve the above research

objective, an end-to-end two-stream network which includes a first-person view-based stream and a motion sensory data-based stream is designed as shown in Figure 5.1. The first-person view-based stream models the user's gaze using an attention module to concentrate on the important spatiotemporal regions of first-person video for context extraction. The motion sensory data-based stream is used to process the motion sensory data to extract features related to the hand motions. Finally, the feature maps coming from these two streams are fused to achieve the hand gesture recognition.

First-Person View-Based Stream

The first-person view-based stream takes RGB images in first-person view as inputs and outputs a visual feature map $X_v \in R^{256\times 2}$ through a visual feature module, an attention module, an attention pooling module and a feature reshape module. Specifically, the first-person frames are first input into a visual feature module to capture an initial visual feature map $X_{iv} \in R^{C\times T\times H\times W}$. C represents the number of input channels. *T, H, W* denote the resolution of the spatiotemporal feature volume. In particular, *T* is the number of time stamps. *H* and *W* refer to the height of and width of the feature volume. In this study, *C, T, H, W* are determined as 1024, 2, 7 and 7, respectively, following the guideline provided in Lu et al. [144]. The visual feature module employs Inflated 3D ConvNets network (I3D) as the backbone 3D CNN model (Figure 5.2). Compared to the previous I3D model, the visual feature module removes the last two layers (one average pooling layer and one convolutional layer). Specifically, the visual feature module has nine inception blocks and three additional 3D convolutional layers to capture the temporal dynamics of consecutive frames. Each convolutional operation in the inception blocks and additional convolutional layers is followed by a batch normalization and a ReLU activation function. For the inception block, it has four parallel branches. Two branches include two

convolutional operations individually while the other two branches contain one convolutional operation each. The outputs of these four branches are concatenated and then passed to the next layer.



Notes: Conv3D means 3D convolutional operations. The first rows which contain numbers in the rounded rectangles are kernel sizes. Taking the first rounded rectangle as an example, it denotes convolutional operations with the kernel size of 7×7×7. The strides of convolution and pooling operators are 1 where not specified. Batch normalization and ReLU layers are not shown. These notes are also applicable to Figures 5.3, 5.4.

Figure 5.2 Architecture of the visual feature module

Further, an attention module is introduced to process the initial visual feature map X_{iv} to generate an attention map $A \in R^{1\times T\times H\times W}$. The architecture of the attention module is shown in Figure 5.3. It consists of an inception block (as detailed in Figure 5.2) and an additional 3D convolutional layer. The attention module takes advantage of the 3D convolutions and leverages spatiotemporal features from consecutive frames to forecast the attention map simultaneously. The attention map models the user's gaze distribution and highlights the important regions of interest to the user.

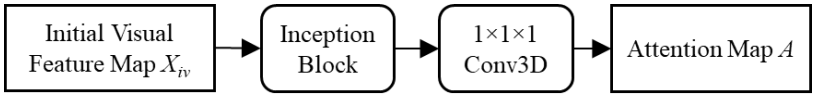


Figure 5.3 Architecture of the attention module

The generated attention map is then applied on the initial features X_{iv} through an attention pooling module to produce a feature map with attention $X_{av} \in R^{C\times T\times H\times W}$. The attention pooling module can be achieved through Equation (5.1):

$$X_{av}^{c} = A \otimes X_{iv}^{c} \tag{5.1}$$

where $X_{av}^{c}, X_{iv}^{c} \in R^{1\times T\times H\times W}$, $c = 1, \cdots, C$. $\otimes$ denotes element-wise multiplication of corresponding entries in A and X_{iv}^{c}. Through this attention pooling module, the higher weights are assigned to the more relevant spatiotemporal features in X_{av}. This practice is beneficial for the following layers to recognize actions.

Finally, the feature map with attention X_{av} goes through a feature reshape module to obtain a reshaped visual feature map $X_v \in R^{256\times 2}$. The architecture of the feature reshape module is shown in Figure 5.4. It has an average pooling layer, a 3D convolutional layer and squeezing operations. Through the feature reshape module, the shape of the visual feature map is resized, which allows for the fusion with motion features coming from the motion sensory data-based stream.

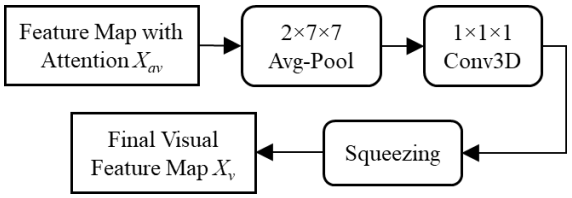
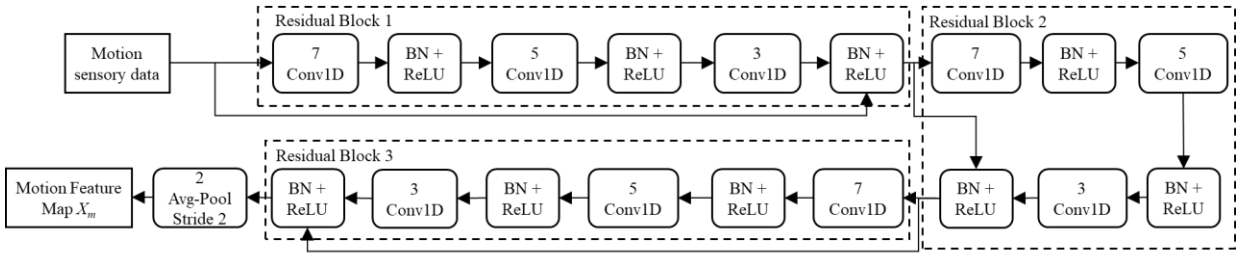


Figure 5.4 Architecture of the feature reshape module

Motion-Sensory Data-Based Stream

The motion sensory data-based stream utilizes motion sensor signals as inputs and processes them through a motion feature module to output a motion feature map $X_m \in R^{256\times 10}$. Figure 5.5 shows the architecture of the motion feature module. In the motion feature module, the ResNet-based network is employed as the backbone model since it is effective for motion sensor-

based multivariate time series classifications [136]. The employed ResNet architecture comprises three residual blocks [145] followed by an average pooling layer. Each residual block is constructed with three convolutional layers. The kernel sizes of these convolutional layers are 7, 5 and 3, separately. Each convolutional operation is followed by a batch normalization and a ReLU activation function. The output of these convolutions is added to the input of the residual block and then passed to the next layer. The numbers of filters for the convolutions in these three residual blocks are 128, 256 and 256, respectively.



Notes: Conv1D means 1D convolutional operations. BN denotes batch normalization. The first rows which contain numbers in the rounded rectangles are kernel sizes. The strides of convolution and pooling operators are 1 where not specified. These notes are also applicable to Figure 5.6.

Figure 5.5 Architecture of the motion feature module

Feature Fusion

The visual feature map X_v and motion feature map X_m are concatenated and processed through a fusion module to achieve the gesture recognition. Specifically, since the sizes of these two feature maps are the same in the first dimension, they can be concatenated to obtain a fused feature map $X \in R^{256\times12}$ along the first dimension (Equation (5.2)).

$$X = [X_v \ X_m] \tag{5.2}$$

The fused feature map X is fed into a fusion module shown in Figure 5.6 to obtain the probabilities for each class of hand gesture. The fusion module consists of a convolutional layer, a max pooling layer and a fully connected layer. After the fusion module, the type of gesture with the highest probability is identified as the final recognition result. In the literature [146,147], this

concatenation + Conv network fusion strategy achieved better performance than other fusion methods such as sum fusion, concatenation fusion, max fusion, bilinear fusion, etc. This practice has exhibited a robust ability of our adopted fusion strategy to leverage complementary information and adaptively learn meaningful joint representations.

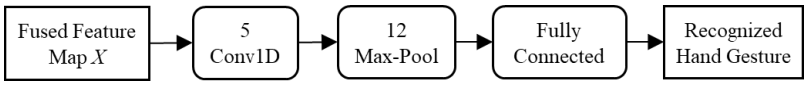


Figure 5.6 Architecture of the fusion module

Loss Function

The loss function of the proposed network consists of two parts including the recognition loss and the attention loss. The recognition loss is the cross-entropy loss L_{CE} which measures the dissimilarity between the predicted probabilities of different gesture classes and the true labels as shown in Equation (4.2).

The attention loss is computed using the mean square error which could evaluate the disparity between the predicted attention map A and ground truth attention map A_g. To generate the ground truth attention map, a Gaussian bump is applied around the gaze points, accommodating the gaze uncertainty to some extent and enhancing attention training. Assume a^{thw} and a_g^{thw} ($t = 1, \cdots, T;\ h = 1, \cdots, H;\ w = 1, \cdots, W$) are the attention weights of A and A_g, respectively. a_g^{thw} can be computed by Equation (5.3):

$$a_g^{thw} = e^{-\frac{(h-x_t')^2+(w-y_t')^2}{2\sigma^2}} \tag{5.3}$$

where x_t', y_t' represent the scaled gaze coordinates in the spatial resolution at time t and σ is determined as $\frac{H}{2}$. Therefore, the attention loss L_A is computed as shown in Equation (5.4). The units of the attention loss should be the square of the units of weights being predicted. Since the weights

of both predicted and ground truth attention maps do not have physical units, the attention loss does not have any units, either.

$$L_A = \frac{1}{THW}\sum_{t=1}^{T}\sum_{h=1}^{H}\sum_{w=1}^{W}(a^{thw} - a_g^{thw})^2 \tag{5.4}$$

The total loss function L_T is a combination of the recognition loss and the attention loss with the ratio of 10:1 as shown in Equation (5.5). The ratio selection process was described in Ablation Study Section.

$$L_T = L_{CE} + 0.1 \times L_A \tag{5.5}$$

Results

Implementation

The device prototype which integrates two sensors for data collection is shown in Figure 5.7. The user is equipped with a Tap Strap [139] and Tobii Pro Glasses [95]. The Tap sensor is placed on the user's hand to obtain the hand movements. It includes five 3-axis accelerometers and one IMU (3-axis accelerometer + 3-axis gyroscope), which could produce 21 signal channels in total. The Tobii Glasses is utilized as the eye tracking glasses to record first-person video and obtain gaze points. The captured data from the Tap sensor and Tobii Glasses can be effortlessly transferred to the computing resources via Bluetooth and Wireless Local Area Network (WLAN), separately.

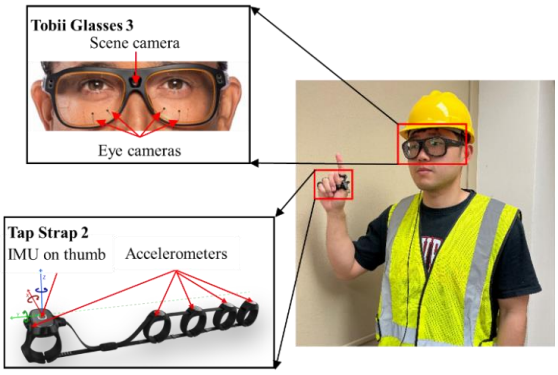


Figure 5.7 Device prototype integrating two sensors

Network Training

To collect training data, ten classes of hand gestures which are designed to direct excavators [148] were selected due to their common utilizations on construction sites for conducting daily activities. These gestures included “Load Up”, “Load Down”, “Swing Right”, “Swing Left”, “Stop”, “Stop Engine”, “Dipper In”, “Dipper Out”, “Open Bucket”, “Close Bucket”. The proposed network was trained and tested with these hand gestures.

When collecting the data, the subject was asked to wear Tobii Glasses on their head and Tap sensor on their hand. To capture the real-world characteristics of hand gestures in construction environments, the subject was requested to move and make gestures simultaneously at construction sites. In the meantime, the subject was requested to stare at the corresponding contextual construction object. For example, the subject would look at the dipper of the excavator when he/she performed the gestures related to the dipper (e.g., dipper in, dipper out). Each gesture class was conducted 45 times and the “no gesture” class was performed 100 times. Hence, 550 (= 45 × 10 + 100 × 1) samples were formed in total. For each sample, the duration was 10 seconds. Since the original sampling rates are different for Tobii Glasses and Tap sensor, the captured frames, gaze points and hand motions were synchronized through unifying their local timestamps. The synchronized sampling rate was 20 Hz. In total, 110000 21-dimensional signal data, 110000 images and 110000 2-dimensional gaze points were collected at five different construction sites. Since the subject only performed one gesture for each sample, the ground truth gesture label could be assigned to the corresponding data sample directly.

For data preprocessing, the collected motion sensor data underwent Z-score normalization to ensure the sensor signal channels to be treated equally important. Also, all the first-person view

frames were resized to the resolution of 320 × 240. The pixel values of these frames were then scaled to [-1, 1]. Figure 5.8 shows some examples of first-person view images and gaze points.

Figure 5.8 Examples of first-person view images and gaze points (red dots)

The compiled dataset above was employed to train the network. Specifically, each sample in the dataset was partitioned into 10 data clips. Each data clip's length was 1 second. Therefore, 5500 data clips were generated in total for the dataset. All these data clips were divided into the training subset (80%) and the validation subset (20%) randomly. The training subset included 4400 data clips while the validation subset contained 1100 data clips.

During training, data augmentation for the input first-person view video clips was applied, involving random crop of 224 × 224 patches and random horizontal flipping. The gaze locations were refined based on the augmentation performed to the frames. Following the training guideline in Carreira and Zisserman [149], the network was trained using stochastic gradient descent with momentum and weight decay set to 0.9 and 10^{-7}, separately. The batch size and initial learning

rate were determined as 12 and 0.1, respectively. The learning rate was reduced by a fixed factor of 0.1 after 300 iterations. The network was first pretrained on the EGTEA Gaze+ dataset [150] and then trained on our compiled dataset for 1000 iterations. This transfer learning strategy could improve the generalization of trained models and meanwhile shorten the training durations required [65]. Figure 5.9 shows the loss reduction along with the training process.

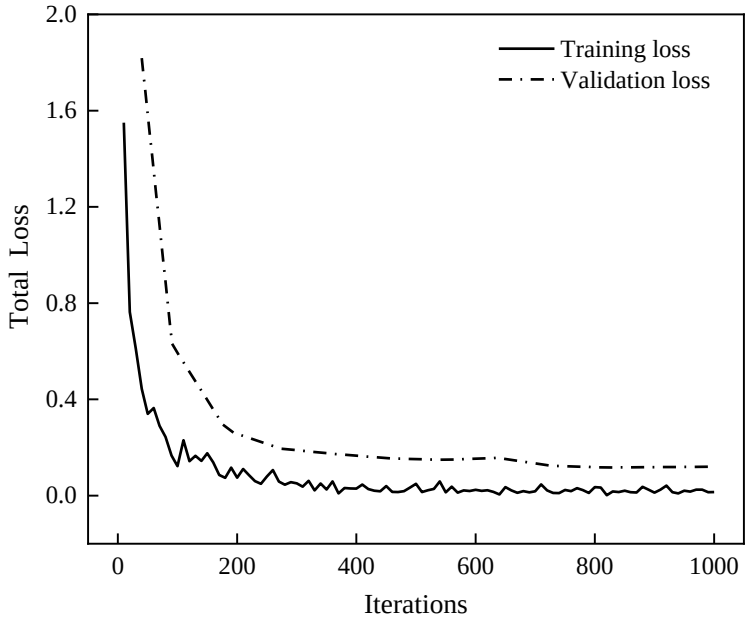


Figure 5.9 Loss function during training process for two-stream network

Network Testing

The recognition performance of the proposed network was tested using a new testing dataset. To collect this dataset, the subject was asked to perform hand gestures and stare at the corresponding contextual construction object simultaneously wearing Tap sensor and Tobii Glasses on sites. Each testing session consisted of five gesture samples, and a total of 24 testing sessions were collected, resulting in 120 gesture samples in total. The reason why five gestures are collected in each testing session is due to the consideration of the time needed to save the collected data on the computer. It would generally take one minute to save the data of one gesture sample.

Therefore, each testing session would entail five minutes for saving the data, which is a reasonable waiting time. The total duration of the testing dataset was 1493.4 seconds.

A cell phone was set up at the construction sites to record the whole data collection process. At the beginning of each testing session, the subject needed to present the local data collection timestamp to the cell phone for the purpose of future calibration. After collection, the calibration between the acquired data and cell phone recordings was manually conducted by aligning their local timestamps. Then, the start and end time of the subject's gestures were manually labeled by observing the recorded data collection process as shown in Figure 5.10. The labeled testing data were fed into the network to investigate whether and how well the performed hand gestures could be automatically captured and interpreted.

Figure 5.10 Manual labeling example

During testing, a sliding window approach was utilized to enable continuous classification of hand gestures. Specifically, the proposed network moved through the data clips with a fixed window of 1 second. The stride size between two consecutive windows was determined as 0.1 second. With the data coming in continuously, the preprocessed data in the latest window would be fed into the proposed network to achieve hand gesture recognition.

Table 5.1 indicates the results of precision and recall for the two-stream architecture. The identification of "Load Up", "Dipper In", "Dipper Out", "Open Bucket" and "Close Bucket" could

reach a higher recognition performance. The overall precision and recall were 93.5% and 94.8%, respectively. Figure 5.11 shows the confusion matrix of the proposed network. The confusion matrix is a specific matrix layout that allows visualization of the performance of the method. Each row *r* of the matrix represents the predicted class while each column *c* means the actual class. The element (*r, c*) of the table refers to the percentage of the actual class *c* which is predicted as the predicted class *r*. The overall classification accuracy achieved 92.6%. The relatively lower classification accuracy happened on the identification of "Swing Right", "Swing Left" and "No Gesture". In general, the more drastic gestures, (e.g., "Load Up", "Close Bucket") exhibit a better performance compared to gestures that resemble static movements (e.g., "No Gesture", "Swing Right").

Table 5.1 Recognition performance of two-stream architecture

Gestures	Load up	Load down	Swing right	Swing left	Stop	Stop engine
Precision (%)	87.8	91.8	98.7	99.7	92.5	95.8
Recall (%)	96.0	94.0	90.1	91.1	94.8	94.5
Gestures	Dipper in	Dipper out	Open bucket	Close bucket	Overall	
Precision (%)	91.9	91.0	92.0	93.8	93.5	
Recall (%)	96.1	96.8	96.1	99.4	94.8	

Figure 5.12 indicates some examples of testing the proposed network to conduct the hand gesture identifications (e.g., "Dipper In" and "Close Bucket"). The results included the generated attention maps and recognized gesture types. Specifically, when the subject performed a gesture and stared at the corresponding contextual object, the proposed network would generate an attention map which reflected the subject's interests and report the corresponding gesture type. Taking the first column in Figure 5.12 as an example, when the subject was performing the gesture

of “Dipper In”, the most important attention (red color) was basically aligned with the shape of excavator dipper in the attention map and the gesture recognition result of “Dipper In” was made. The average delay in recognition was around 0.45 second.

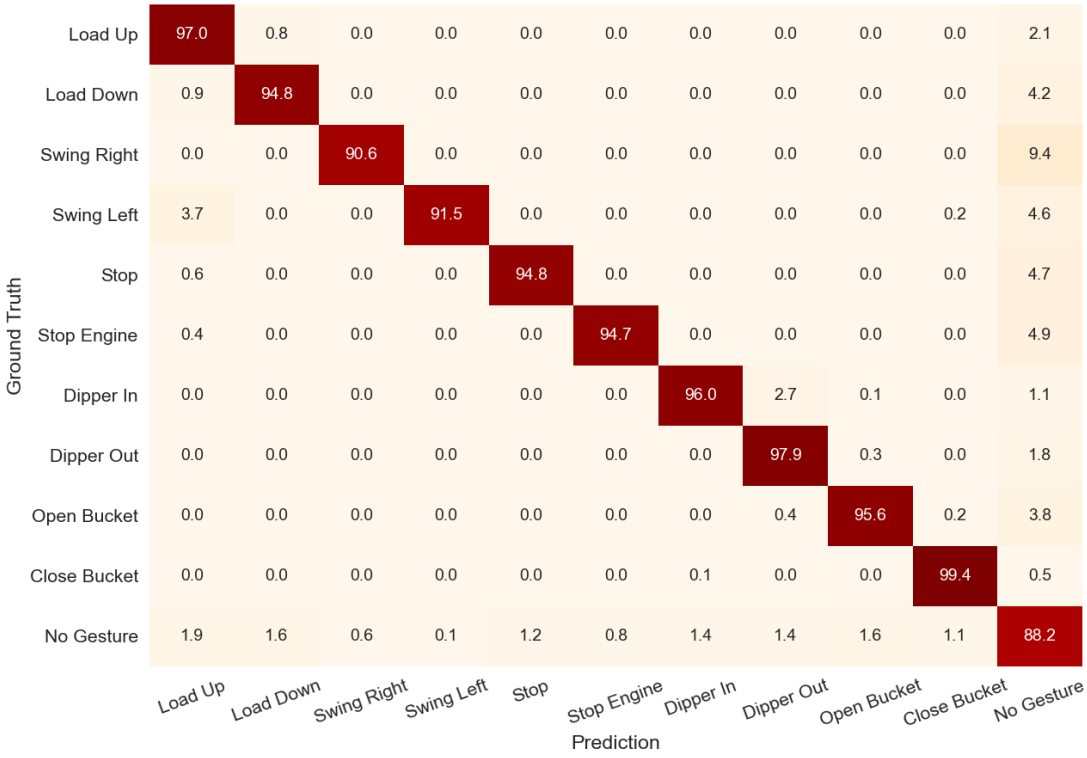


Figure 5.11 Confusion matrix of the two-stream network

Comparison with Single Sensor Stream

To validate the superiority of our two-stream architecture, a comparison was conducted between the single sensor stream and our proposed network. Table 5.2 summarizes the recognition performance of the single sensor stream and two-stream architecture. The single sensor stream achieved an overall classification accuracy of 91.1% on the testing dataset. Compared to the sensor stream, our proposed network improved the accuracy by 1.5%, thereby showing an efficiency improvement for hand gesture recognition in construction. A paired T-test was performed for the classification accuracies between the two-stream network and the single sensor stream. The results

yielded a P-value of 2.02×10^{-6}, which indicated that there was significant performance gain for our proposed two-stream network.

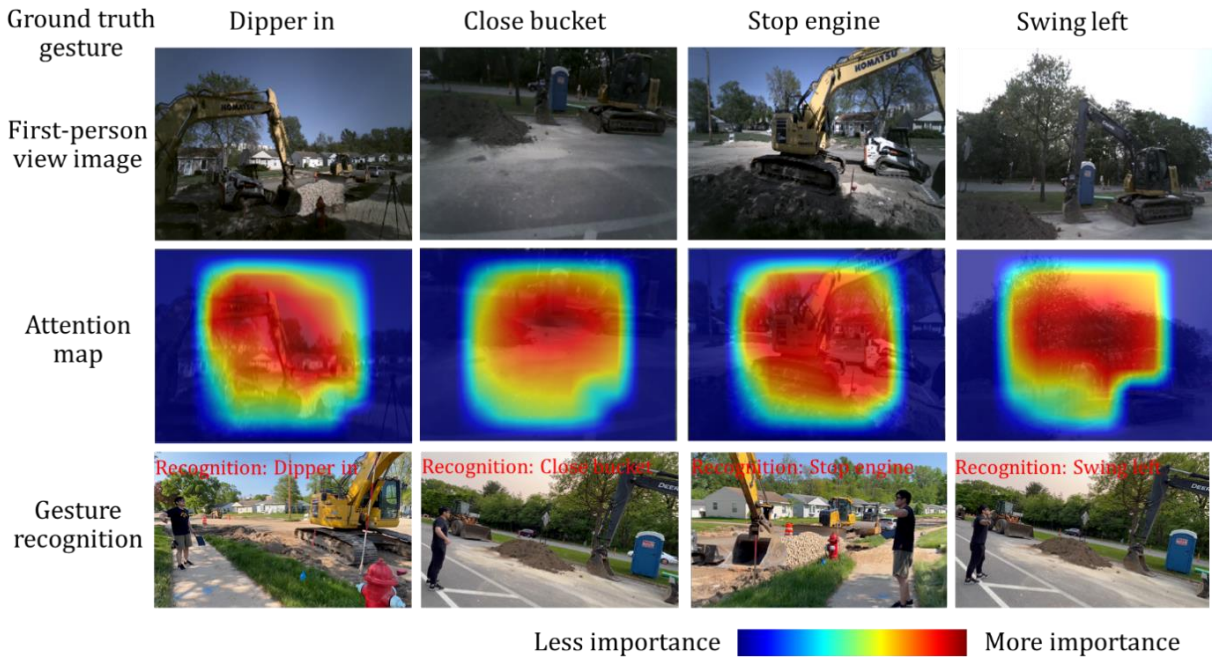


Figure 5.12 Examples of attention maps and gesture recognition results

Table 5.2 Comparison of single sensor stream and two-stream architecture

Accuracy of Single Sensor Stream (%)	Accuracy of Two-Stream Architecture (%)	Performance Gain (%)
91.1	92.6	1.5

From Chapter 4, it is found that the gesture recognition models usually had relatively bad performance on near gesture start and end data, which could lead to more delay in recognition. The near gesture start and end data refer to a specific subset of the testing dataset. This subset is created by selecting data that are close to the start and end time of each gesture sample within the dataset. Specifically, assume ST_i and ET_i are the start and end time of a gesture sample i and L is the half window length. The data in $[ST_i - L, ST_i + L]$ and $[ET_i - L, ET_i + L]$ for all gesture samples

were picked to form this new testing dataset. Figure 5.13 shows the performance gain of our network on this new testing dataset compared to the single sensor stream. With the rise of the half window length, the performance gain increased first and then dropped. The reasons for this phenomenon are analyzed in the Discussion section. The maximum performance gain was achieved at L = 0.6 second with the value of 4.7%, which indicated a notable performance gain on near gesture start and end data. This practice implied that our network could trigger the gesture recognition earlier when the gesture sample started and conclude recognition timely when the gesture sample ended.

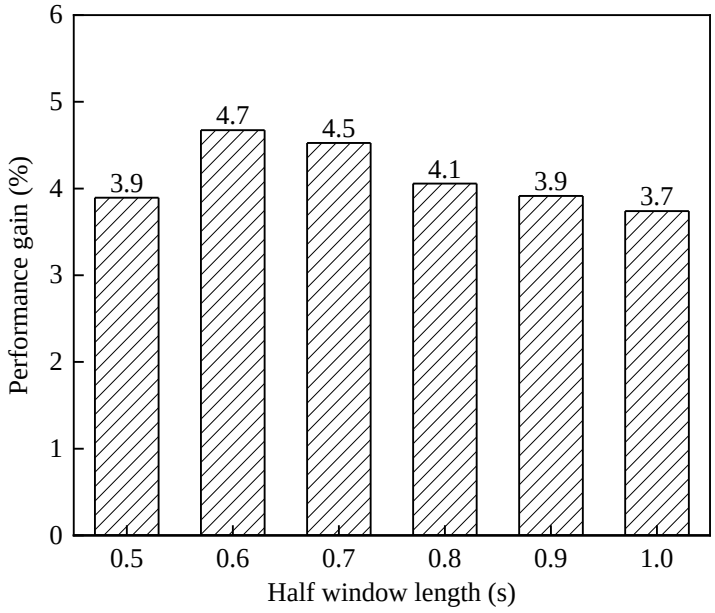


Figure 5.13 Performance gain on near gesture start and end data

Ablation Study

Further, the ablation studies were conducted to investigate the contributions of attention maps and the effects of the ratio between loss functions for the proposed network. First, a comparison between the proposed network and the two-stream network without attention map generation was carried out to explore whether the generated attention maps could improve the recognition performance. In the two-stream network without attention map generation, the

attention module and attention pooling module are removed. The first-person view frames are only fed into the visual feature module and feature reshape module to get the visual context. It is reported that the accuracies of our proposed two-stream network and the two-stream network without attention map generation were 92.6% and 92.0%, separately. With the generated attention maps, the recognition accuracy increased by 0.6%. These results could validate the effectiveness of introducing attention maps to improve the recognition performance.

Second, the proposed network was trained with different loss functions to study their effects. As stated in Loss Function Section, the total loss function is a combination of the recognition loss and the attention loss. In this ablation study, different ratios between the recognition loss and the attention loss were explored, including 1:1, 5:1, 10:1, 15:1 and 20:1. The results showed that the recognition accuracies were 92.0%, 92.4%, 92.6%, 92.2% and 91.8%, respectively. With the increasing ratios, the accuracy increased first and then dropped. The best performance was achieved with the ratio of 10:1, which provided reference for determining the total loss function.

CHAPTER SIX: MACHINE-AWARE INTERACTION FOR ONE-TO-MANY COLLABORATIONS

The purpose of this chapter is to propose a novel eye gaze-aware method to visually indicate which machine the user intends to interact with and control multiple machines on construction sites. Further, this method is tested through a method validation test to assess its interaction performance. The results demonstrate that the proposed method is suitable for designing an effective interface for one-to-many collaborations on construction sites.

Machine-Aware Human-Robot Interaction Method

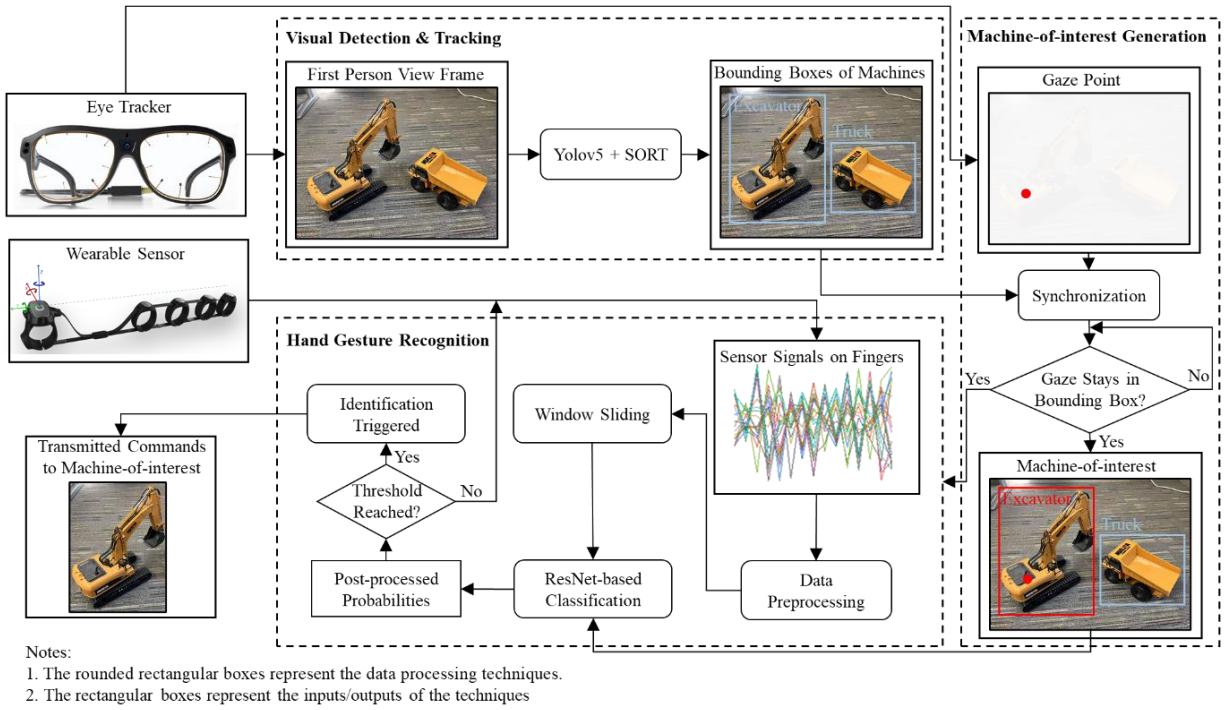


Figure 6.1 Overview of machine-aware interaction method

The main objective is to propose a novel machine-aware hand gesture recognition method as a human-robot interface for use on construction sites having multiple types of machines. The proposed method comprises three components as illustrated in Figure 6.1. It firstly relies on an eye tracker to visually detect and track construction machines in the first-person view. Then, the

machine-of-interest is determined based on the bounding boxes of machines and gaze points. Finally, a hand gesture recognition architecture is incorporated with the machine information for conveying messages to the machine-of-interest.

Visual Detection and Tracking

The purpose of this component is to detect and track the construction machines from the first view frames in eye tracker. Visual detection and tracking refer to the process of identifying and following objects in a video stream using computer vision techniques. A tracking-by-detection paradigm proposed in the work of Bewley et al. [151] is employed here due to its superior performance with respect to both speed and accuracy. Within this paradigm, the detection module identifies the construction machines in each frame and obtains their bounding boxes. Given detection results, the trajectory information is modeled to associate current detections with existing tracks for the life-span tracking of the machines.

Within this paradigm, YOLOv5 [152] is selected to detect the construction machines because of its fast and accurate nature and ability to provide multi-scale prediction. Additionally, many research results have verified the high performance of YOLO series algorithms in various construction objects detection (e.g., workers [80], construction materials [153]). Simple Online and Realtime Tracking (SORT) tracker is employed to relate the same construction machine detected in the previous process across all the frames [151]. The SORT tracker is selected since it is able to track the objects in a fast and accurate manner and also avoids additional training. The trajectory information provided by the detection results is adopted to track the construction machines in video frames.

Machine-of-interest Generation

The purpose of this component is to generate the machine-of-interest based on the bounding boxes of construction machines and gaze points. This component can be divided into two steps: synchronization for the bounding boxes and gaze points, and interaction mode triggering. The bounding boxes are derived by processing the first-person view frames from the scene camera where the gaze points are produced from the eye cameras. Since the original sampling rates are different for these two kinds of cameras, the bounding boxes and gaze points are calibrated for synchronization based on a unified timestamp.

As for the triggering of the interaction mode, if the gaze points stay in the bounding box of one specific machine for a duration longer than a threshold μ, the corresponding machine is determined as the machine-of-interest. Further, the machine-of-interest will enter the interaction mode and the hand gesture recognition component can then be triggered to convey messages to the machine-of-interest; otherwise, it means that the method is not confident regarding which machine the user desires to interact with. The method would continue to check the above condition until the user terminates this process. It should be noted that the selection of μ depends on how likely the user intends to trigger the interaction mode. Generally, the fixation duration, which is the time during which the eyes rest on an object in the surroundings, lasts approximately 0.25 second or longer [154]. Therefore, μ has been set as 0.3 second here.

Hand Gesture Recognition

The purpose of this component is to adapt the hand gesture recognition architecture in Section Chapter 4 to incorporate the machine information for classifying the hand gestures made by the worker. This component is made up of 3 modules: data preprocessing, window sliding and ResNet-based gesture recognition. For data preprocessing module, the accelerometer and gyroscope signals captured directly from wearable sensors are first calibrated to a unified sampling

rate (e.g., 10Hz) for synchronization since the original sampling rates may be different for various sensors. Then, Z score normalization is utilized to transform the data values to have a mean of 0 and a standard deviation of 1, which allows all the signal channels to be considered with equal importance. To recognize consecutive gestures made by workers, a sliding window approach is employed for continuous classification of hand gestures. Further, a ResNet classifier is introduced to extract features and obtain the raw probabilities for each type of hand gesture. The employed ResNet architecture is the same as the architecture introduced in Section Motion-Sensory Data-Based Stream of Chapter 5.

The incorporation of machine information into gesture recognition is conducted as follows. Assume that there are *m* construction machines the user may intend to interact with and *n* types of gestures in total. First, the probability weighting techniques are applied to post process the raw probabilities through Equation (6.1).

$$P_j = \delta_{ij} \times p_j \tag{6.1}$$

where P_j demonstrates the post processed probability for gesture type j, p_j refers to the raw predicted probability for gesture type j, δ_{ij} is the indicator variable (i = 1,…,m; j = 1,…,n). The definition of δ_{ij} is shown in Equation (6.2).

$$\delta_{ij} = \begin{cases} 1, \text{if gesture type } j \text{ can be used for directing machine } i \\ 0, \text{if gesture type } j \text{ can not be used for directing machine } i \end{cases} \tag{6.2}$$

When the machine i (machine-of-interest) is determined, δ_{ij} for each type of gesture can be obtained and only the probabilities of hand gestures which can be used for directing machine-of-interest will be further reserved. Then, the identification of the hand gesture will be triggered if the highest post-processed probability is more than a threshold θ; otherwise, the data of the next sliding window will be collected and further input into the classifier if the machine-of-interest still exists. Here, θ is selected as 0.70 based on the average confidence for making correct predictions. Finally,

the gesture meaning can be obtained based on both the identified gesture type and machine-of-interest. The meaning message would be conveyed to the machine-of-interest for the corresponding operation.

Results

Offline Training for Gesture Recognition

Fourteen classes of hand gestures for directing excavators and trucks are selected since these construction machines are commonly employed on construction sites [155]. Among them, eight classes of gestures unique to excavators and four classes of gestures are unique to trucks. Two classes of gestures can be used for directing both excavators and trucks. These fourteen gestures include “Load Up”, “Load Down”, “Stop”, “Swing Right / Move Right”, “Swing Left / Move Left”, “Stop Engine”, “Dipper In”, “Dipper Out”, “Open Bucket”, “Close Bucket”, “Pull Forward”, “Proceed Backwards”, “Raise Truck Bed, Load, etc.” and “Lower Truck Bed, Load, etc.”. The classifier in the proposed method will be trained and tested using these hand gestures.

During the data collection, the subject was requested to wear the Tap sensor on his/her hand at first. To capture the characteristics of hand gestures in construction site environments, the subject was moving and making hand gestures synchronously. Each class of the gestures was performed 20 times and the “no gesture” class was conducted 35 times by the subject, resulting in 315 (=20 × 14 + 35× 1) samples in the dataset. The duration for each sample is 10 seconds, which means the capture of one sample generates 100 signals per channel. The boxplot of the dataset is indicated in Figure 6.2. It describes the distribution of sensor signals after Z score normalization for each class of gestures. The average normalized signal values for these 14 classes of gestures are 0.046, -0.850, -0.301, 0.325, 0.335, 0.480, -0.111, -0.238, -0.176, -0.184, 0.344, 0.191, 0.352 and -0.342, separately. The gestures of “Load Up”, “Load Down” and “Stop” have a larger value

range since they generally involve a more drastic movements of hands/fingers. In contrast, “Swing Right / Move Right”, “Swing Left / Move Left” and “Pull Forward” have a smaller variance because these gestures basically remain static during the conduction.

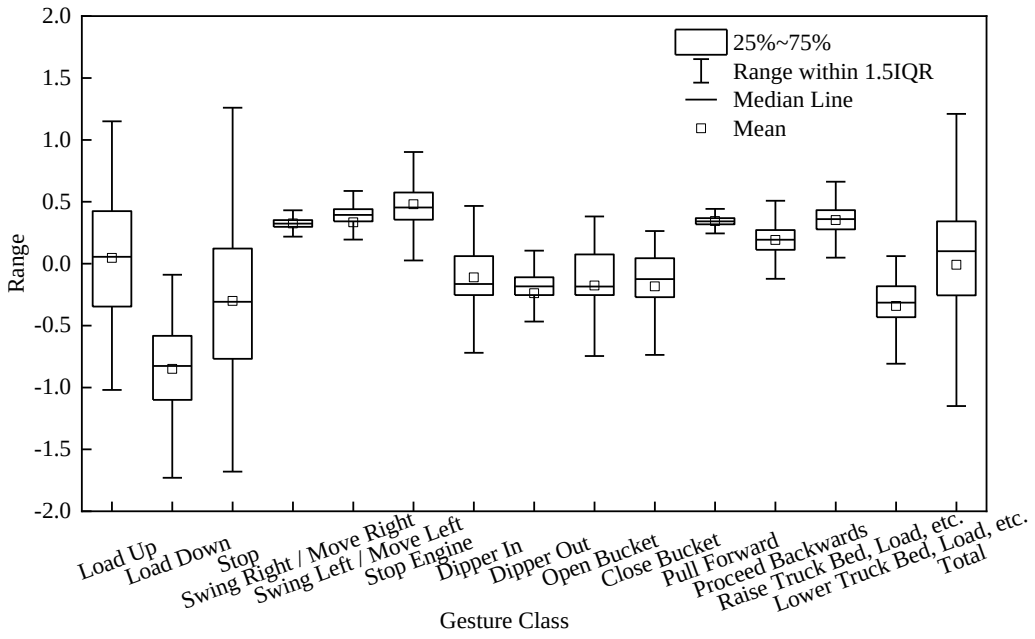


Figure 6.2 Boxplot of the dataset for machine-aware method

The established dataset would provide a benchmark for construction hand gestures. It could be used to test and compare the performance of different algorithms to identify the most effective solution to hand gesture recognition in construction. Also, the benchmark dataset could help researchers to investigate other problems, such as multivariate time series classification, human activity identification, and sensor signal processing.

The created dataset is used for offline training of the gesture classifier. For training, each sample of the dataset is divided into 8 data clips with the clip length of 3 seconds. This practice results in 2520 data clips in total for the dataset. The 10-fold cross-validation strategy is employed to train and validate the classifier. Specifically, the 2520 data clips are randomly split into 10 equal-sized subsets. In this case, each subset contains 252 data clips. For each subset, the classifier is

trained on the remaining 9 subsets (2268 clips) and tested on the one subset (252 clips) that is held out. The above step is repeated for each subset until all the subsets have been tested. After that, the average performance across all 10 subsets is calculated to validate the classifier.

Table 6.1 shows the parameter setting for the training. The specific training process is conducted as follows. The learning rate and the batch size are initially set as 0.001 and 32, respectively. The cross-entropy loss is employed as the loss function. The learning rate is reduced with a fixed decay factor of 20 when the validation loss has no improvements. Adam, which is a stochastic gradient descent method based on adaptive estimation of 1st and 2nd moments, is employed as the optimizer. The 1st moment estimate, the 2nd moment estimate, and weight decay are set as 0.9, 0.999 and 0.001, separately. Figure 6.3 shows an example of the loss reduction along with the training progress. The training and validation loss tend to be stable when the epoch reached about 60.

Table 6.1 Parameter setting for the ResNet classifier in machine-aware method

Classifier	Initial learning rate	Step size of learning rate decay	Minimum learning rate	Batch size
ResNet	0.001	20	0.0001	32

The offline recognition performance of the classifier was reported in terms of precision and recall. The results of 10-fold cross-validation show that the ResNet classifier achieves an overall precision of 98.2% and recall of 98.2% across all the gesture classes.

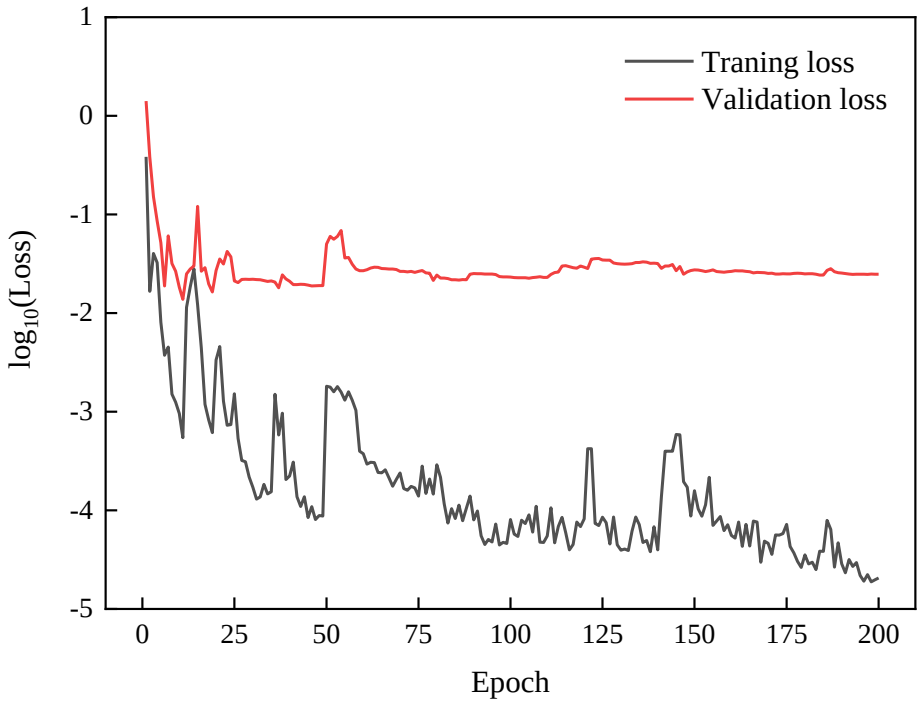


Figure 6.3 Loss reduction along with the training progress for machine-aware method

Method Validation Test

The effectiveness of the proposed method was tested through a method validation test. The subject was asked to stare at a robotic excavator or truck and perform hand gestures wearing Tobii Glasses and Tap sensor in a laboratory environment. Five gesture samples formed one testing session. In total, 32 testing sessions were collected, which included 160 gesture samples. The collected testing data were fed into the method to investigate whether and how well the machine-of-interest could be generated and the performed hand gestures could be automatically captured and interpreted.

Fig. 6.4 shows some examples of testing the proposed method to generate the machine-of-interest and conduct the hand gesture identification. For instance, when the subject stared at the excavator at 22 seconds, the machine-of-interest was determined as “Excavator” and the proposed method triggered the gesture identification module. Then the subject performed a hand gesture and

the method recognized the corresponding gesture type as “Stop”. Also, at 28 seconds, no gesture was identified when the subject was not looking at the robotic machines since he did not specify which machine he intended to interact with.

(a) Machine-of-interest generation

(b) Hand gesture recognition

Figure 6.4 Examples of the method validation test for machine-aware interaction

Tables 6.2 and 6.3 show the performance of generating the machine-of-interest and the recognition performance of hand gestures under each type, separately. The performance of generating the machine-of-interest was evaluated by the generation accuracy while the gesture recognition performance was reported in terms of precision and recall. The generation accuracy is defined as the ratio of the number of correctly generated samples to the number of total samples the subject stared at the machines. For machine-of-interest generation component, the generation

accuracy achieved 97.5%. Besides, it was found that the overall precision and recall achieved 93.8% and 95.0%, respectively, for the gesture recognition. The identification of “Load Up”, “Load Down”, “Swing Right / Move Right”, “Pull Forward”, “Proceed Backwards” and “Raise Truck Bed, Load, etc.” could reach up to 100% precision and 100% recall. The lowest precision happened on the identification of “Open Bucket” while “Dipper In” and “Dipper Out” obtained the lowest recall.

Table 6.2 Performance of generating the machine-of-interest

Machines	Excavator	Truck	Overall
# Correctly generated samples	97	59	156
# Total	100	60	160
Generation accuracy (%)	97.0	98.3	97.5

Table 6.3 Recognition performance of the machine-aware interaction method

Gestures	Load Up	Load Down	Stop	Swing Right / Move Right	Swing Left / Move Left
Precision (%)	100.0	100.0	100.0	100.0	100.0
Recall (%)	100.0	100.0	90.0	100.0	95.0
Gestures	Stop Engine	Dipper In	Dipper Out	Open Bucket	Close Bucket
Precision (%)	100.0	100.0	72.7	62.5	90.9
Recall (%)	90.0	80.0	80.0	100.0	100.0
Gestures	Pull Forward	Proceed Backwards	Raise Truck Bed, Load, etc.	Lower Truck Bed, Load, etc.	Overall
Precision (%)	100.0	100.0	100.0	100.0	93.8
Recall (%)	100.0	100.0	100.0	90.0	95.0

Pilot Study

A pilot study was conducted in a laboratory environment to test whether the proposed method could serve as an effective interface to help workers control and/or interact with multiple robotic construction machines. Figure 6.5 illustrates the setup of the laboratory experiment and the related data flow. The subject was asked to stare at the machine he/she intended to interact with for at least 0.3 second. This intention was captured by Tobii Glasses. Then the subject performed hand gestures, which were recorded by a Tap sensor. The obtained frames, gaze points and motion sensory data were input into the method and processed there in real time. Based on the identified machine-of-interest and recognition results, the corresponding instructions would be sent to a remote controller, where the control signals would be transmitted to operate the machine-of-interest remotely.

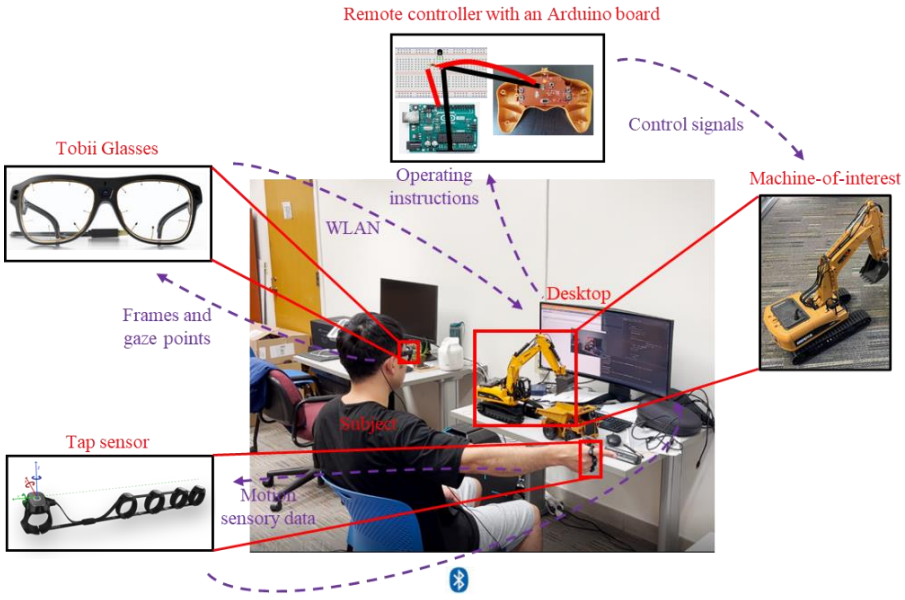


Figure 6.5 Pilot study setup and data flow for machine-aware interaction

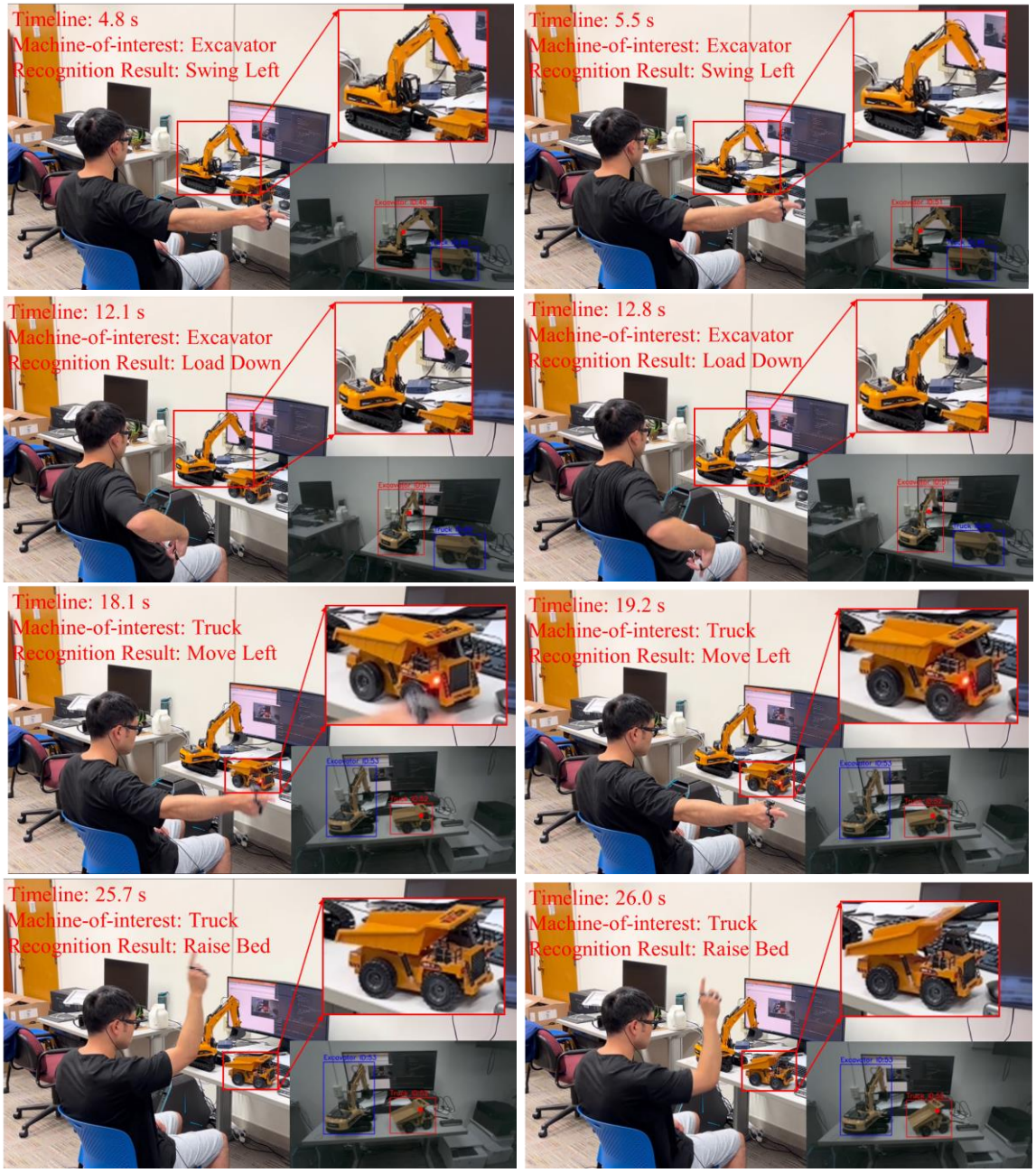


Figure 6.6 Examples of the pilot study for machine-aware interaction

Figure 6.6 shows an example of using the proposed method to remotely control an excavator model and a truck model. The subject first made a series of hand gestures of “Swing Left/Move Left” and “Load Down” when staring at the excavator. The machine-of-interest was determined as “Excavator” by the method. Then, these two gestures were captured and interpreted

as “Swing Left” and “Load Down”, separately. The corresponding instructions were sent to the excavator model through the remote controller. Following instructions, the excavator model swung its boom from right to left and then lowered its boom gradually. After a short pause, the subject performed the gestures of “Swing Left/Move Left” and “Raise Truck Bed, Load, etc.” when looking at the truck model. The gestures were captured by the method and recognized as “Move Left” and “Raise Bed”, respectively. The truck model received the corresponding instructions sent by the remote controller. Finally, it drove towards the left and then lifted its dump bed gradually.

Ablation Study

Further, we conducted the ablation studies to investigate the contributions of attention mechanism and post processing for the proposed method. First, a comparison between ResNet-based method and the proposed method with the enhanced ResNet is carried out to explore whether the introduction of the attention mechanism could improve the proposed method. The only difference for these two methods is that different networks (ResNet, enhanced ResNet) are employed as the gesture classifier. Figure 6.7 shows the comparison results on the method validation test. The ResNet-based method achieved precision and recall of 89.9% and 94.4%, respectively. In contrast, the proposed method with the enhanced ResNet achieved 3.9% and 0.6% higher performance of precision and recall, respectively. These results validate the effectiveness and superiority of introducing the attention mechanism into the ResNet structure.

Second, the proposed method is evaluated against the method without post processing to investigate the contribution of incorporating the machine information into the gesture classifier. For the method without post processing, the probability weighting techniques are not applied to post process the raw probabilities from the gesture classifier. Figure 6.8 shows the comparison results on the method validation test. For the method without post processing, the precision and

recall achieved 91.0% and 94.4%, respectively. In contrast, the proposed method increased the precision and recall by 2.8% and 0.6%, respectively. The relatively large improvement of precision performance indicated that there were less non-gesture/other gesture samples falsely recognized as some gesture samples by the proposed method. Due to the incorporation of post processing techniques, only the probabilities of hand gestures which can be used for directing machine-of-interest will be reserved. Therefore, the identification of hand gestures which could not be applied for the machine-of-interest would never be triggered. This practice could largely reduce the misclassifications of motion outliers as gestures.

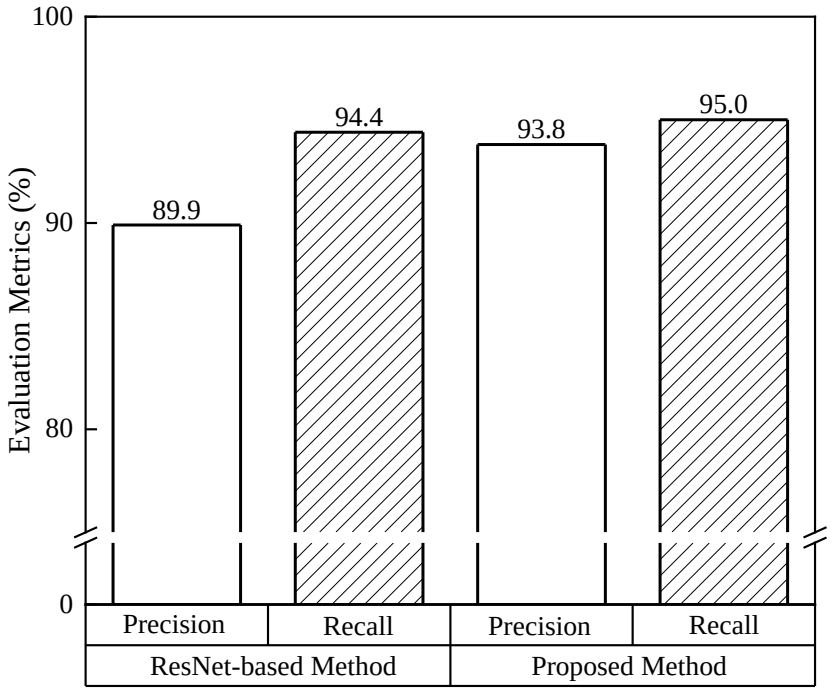


Figure 6.7 Comparison between ResNet-based method and the proposed method

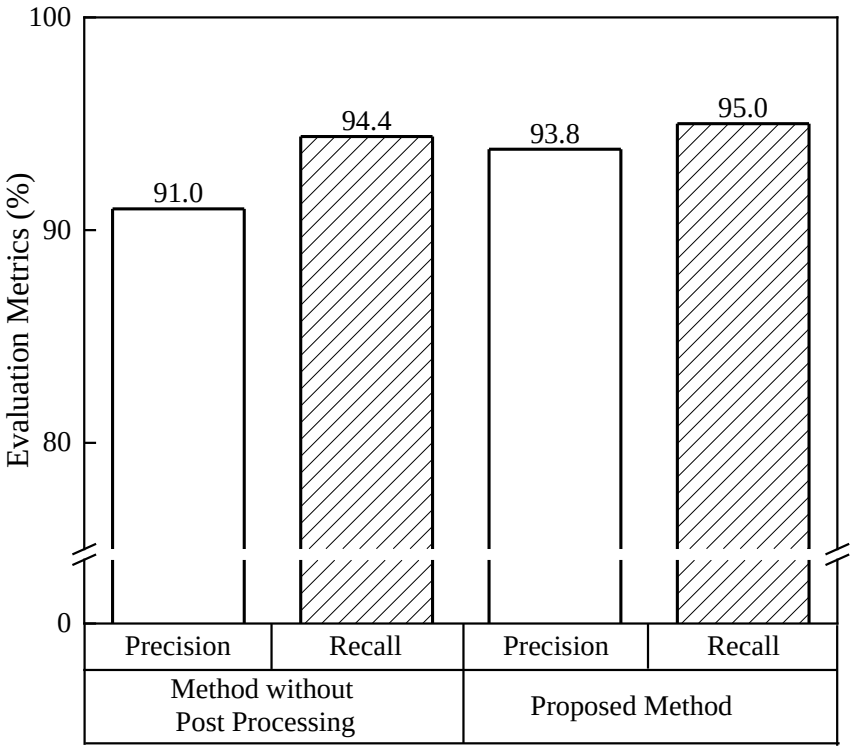


Figure 6.8 Comparison between the methods with and without post processing

CHAPTER SEVEN: DISCUSSIONS

The gesture recognition results show that hand gesture classes impact the recognition performance of the gesture classifier. As shown in Tables 4.6 and 4.7, the gesture classes with relatively low precisions or recalls are “trolley travel left”, “lower”, and “swing left”. This is partly because the movements of these gestures are similar to other gestures. For example, the captured motion sensory data of “trolley travel left” was similar to that of “trolley travel right” since only the direction information is different for these two classes of gestures. Such situations can easily lead to misclassifications especially when the subject is moving and making hand gestures simultaneously.

Besides, the performance of the proposed vision-based and sensor-based methods may be affected by outlier motions. For example, there was one false prediction of “trolley travel right” as shown in Figure 7.1 (a). The subject was managing his T-shirt, which was similar to some movements of the gesture “trolley travel right” (Figure 7.1 (b)). This matter may become worse on real construction sites considering the human activities in construction can be quite complicated. The regular construction tasks of the worker (e.g., turning the screws, loading and unloading materials, etc.) may make gesture classifiers confusing since some movements of these tasks are similar to the predefined gestures. To solve this limitation, an interaction activation/deactivation mechanism could be incorporated into the system to activate the gesture recognition only when the worker is performing the hand gestures. Also, sensor fusion is a promising approach since it would generate richer signal contents and increase robustness for fine-grained gesture classification [140].

Figure 7.1 An example of false prediction of "trolley travel right" (left: false prediction, right: ground truth)

Table 7.1 Performance comparison with different subjects in the field videos

Indicators	Subject 1	Subject 2
Precision (%)	89.5	75.0
Recall (%)	58.0	60.0

A diversity is expected when different subjects perform a same hand gesture. Therefore, the extensive training to capture this diversity plays an important role in improving the hand gesture recognition performance. Table 7.1 compared the recognition performance of the hand gestures made by two subjects in the vision-based method. The recognition performance of the hand gestures made by subject 1 was superior to those made by subject 2. It may be because the hand gesture samples conducted by subject 2 were never used for the training of the hand gesture detector and classifier in the proposed method. It is challenging for them to capture and differentiate the features of the gestures they had never seen before. The generalization issue is a universal problem for machine learning models. A wide range of classification models did not reach their original accuracy scores on unseen data. For example, it is reported that the accuracy drops of different models range from 3% to 15% on CIFAR-10 and 11% to 14% on ImageNet

[143]. In the vision-based method, the drops for precision and recall are 14.5% and 8.0%, respectively, which are basically in an acceptable range.

The hand gesture was not recognized immediately after it was made. Here, the moment for triggering the hand gesture recognition was investigated. Figure 7.2 indicated the frame indexes when hand gestures were started and finished for Subject 2 in the vision-based method. Compared with the ground truth, the recognition of a hand gesture was typically triggered only when enough hand gesture motions were captured and interpreted. Typically, the recognition was always made 61 frames (approximately 4 s) later after the start of the hand gesture. The late recognition may be due to the preparation, nucleus and retraction parts from the beginning to the end of a dynamic gesture [156]. The nucleus is the most discriminative part while the other two parts can be quite similar for different gesture types. Thus, the classifier in the proposed method can only make reliable classification decisions until the gesture enters its nucleus part. It should be noted that the late response is acceptable for most of the gesture types, such as “swing right” and “hoist”, but may decrease the interaction efficiency when a gesture needs to be recognized as soon as possible like “emergency stop”.

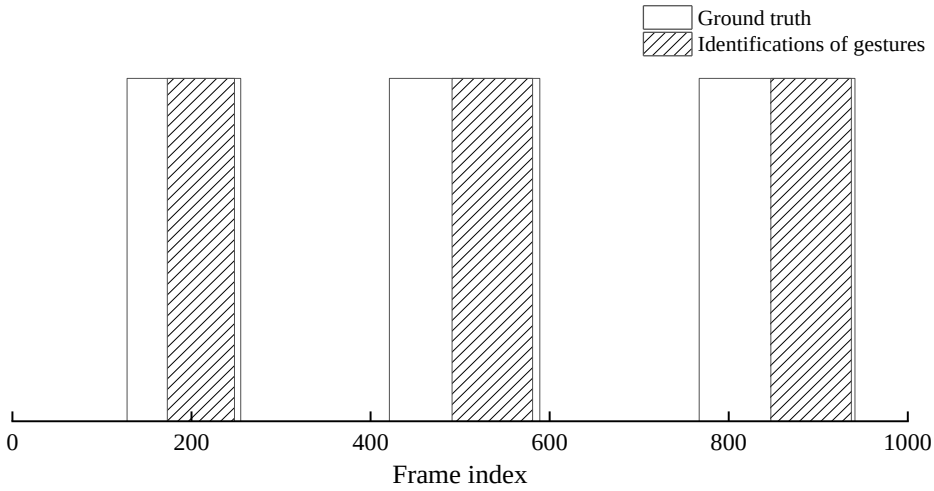


Figure 7.2 Frame indices of the identifications in the vision-based method

It is found that the performance gain of object-aware method increased first and then dropped with the rise of half window length as shown in Figure 5.13. This phenomenon is reasonable and interpretable. When the half window length tends to be a very big value, the picked new testing dataset would be closer to the whole dataset. Then, the performance gain on the picked dataset would approach the gain (1.5%) on the whole testing dataset. Conversely, when the half window length tends to be 0 second, the picked dataset would be closer to contain only the gesture start and end data. In this case, it is challenging for both the two-stream architecture and single stream to recognize gestures based on such data, which would lead to a lower performance gain. Based on these two scenarios, the performance gain would increase first and then drop with the rise of half window length. In our study, the maximum performance gain of 4.7% was achieved at L = 0.6 second.

The more drastic gestures generally have a better recognition performance compared to those which are close to static gestures. As illustrated in Table 6.3, the more drastic gestures (e.g., “Load Up”, “Load Down” and “Stop”) achieved the precision and recall of 100.0% and 96.7%, separately. In contrast, the more static gestures (e.g., “Swing Right / Move Right”, “Swing Left / Move Left”, “Stop Engine”, “Dipper In”, “Dipper Out”, “Pull Forward”) obtained the precision and recall of and 95.5% and 90.8%, respectively. Other researchers [157,158] have also found that the classification accuracy of drastic human movements was higher compared to static movements. This might be because the more drastic gestures generally have a large value range of signal values as shown in Figure 6.2, which contain richer signal contents that help the gesture classifier to extract more prominent features from them.

The proposed methods have the potential for developing a real-world human-robot collaboration system on construction sites. For example, the methods could be equipped with

autonomous machines (e.g., autonomous excavator and truck) and the construction workers would wear the required sensors. In this case, the workers could adjust the behaviors of machines easily based on the workers' intentions. Also, the movements of the machines could be passed back to the workers as feedback. This two-way interaction could enhance the usability and accessibility of autonomous machines, making them more user-friendly and efficient in dynamic construction environments.

Although the benefits of accuracy and context awareness for our proposed methods are evident, there are still several aspects that need to be further investigated. First, it is essential to carefully consider the trade-off between recognition performance and real-time responsiveness. In the object-aware method, the integration of multiple streams would introduce additional computational complexity, which may lead to a longer inference time. This would require more powerful computing resources in time-sensitive applications. Second, the current connection modes between the sensors and computer may have potential data loss risks. Both Bluetooth and WLAN signals could face interference from physical obstructions, environmental factors, or other devices operating on the same frequency. Therefore, additional signal enhancement devices may be needed to increase the data transmission reliability. Third, the machine-aware method highly depends on the detection and tracking of construction machines. In the validation test, there were 3 times that the machine-of-interest was not identified since the robotic machines were not successfully detected and tracked. These led to the method missing all the corresponding gestures since the motion sensory data were not input into the method. Considering that construction scenarios are complex and cluttered with tools, materials, workers, etc., the trained detection and tracking models need to be robust to accommodate such challenging characteristics of the environment. Fourth, the gaze point accuracy is critical for determining which machine the user

intends to interact with. However, several complicating environmental factors at construction sites, such as diverse weather conditions and sunlight intensities, pose challenges for accurate estimation of eye gaze points. Five, the proposed methods still need to be trained and validated by more complicated construction scenarios to improve the generalization. In our testing sessions, the operator, background, and distance from the machines are different. However, the current testing is only focused on the single interaction with the excavator/truck without the inference of other materials and workers. Also, the effects of diverse habits of performing gestures and different equipment have not been fully investigated in this study.

CHAPTER EIGHT: RESEARCH CONTRIBUTIONS

In summary, the proposed methodology makes the following technical contributions: (1) creating new datasets of construction hand gestures under different scenarios, (2) introducing a time factor into the hierarchical gesture recognition architecture to identify consecutive gestures made by the workers, (3) integrating attention mechanism into the gesture classifier for performance improvement, (4) extracting useful visual context with attention maps from first-person view, (5) fusing different data sources (video + motion) to support object-aware human–robot interactions in construction settings, and (6) incorporating machine information into gesture recognition to map one gesture to multiple meanings.

The expected practical contributions of the proposed methodology include improving interaction efficiency with construction robots, decreasing onsite safety issues, refining the design and implementation of construction robots, promoting the adoption of robots in construction, etc. The establishment of the proposed methodology could facilitate the interaction efficiency between construction robots and workers in dynamic construction environments. It allows workers to adjust the behaviors of robots easily based on the workers' intentions. This practice would greatly enhance the operability and adaptability of robots in fast-changing onsite working conditions.

The developed human intent representation could decrease the safety issues (e.g., collisions) between robots and workers. The built representation makes it possible for workers to work with robots at a distance to reduce potential collisions. Also, it provides an opportunity for all the onsite workers to communicate with the robot. Therefore, any worker can stop or move away the robot when he/she anticipates that the robot's behavior would be life-threatening to him/herself or other existing workers (e.g., the distance between the worker and the robot being too close).

Besides, the intent interpretation of construction workers could help understand how human workers and robots work collaboratively and safely on construction sites, which will further refine the design and implementation of construction robotic systems. By integrating worker intent representation into the design of robotic control systems, engineers can create interfaces that align more closely with the natural ways in which humans communicate and interact with their environment. For instance, robots equipped with sophisticated vision systems can track the gaze of construction workers to understand where their attention is focused, allowing them to provide assistance or adjust their actions in real-time.

In addition, the proposed methodology could help to promote the adoption of robots in construction. Based on the previous literature [12], the operational and personnel barriers occupied 30.5% of factors which led to the low adoption of robots in construction. Consider that the proposed methodology could contribute to improving interaction efficiency in operational barriers and decreasing safety issues in personnel barriers. Predictably, the adoption of construction robots would be thus promoted. The promotion of construction robots would bring further benefits such as improving the construction productivity and quality.

In the future, the proposed methodology has the potential to develop a new construction paradigm. On the future sites, there would be very few workers. The construction machines would have a high level of autonomy. The machines are able to conduct the daily construction activities autonomously. The present workers only need to supervise and monitor the whole construction site. They are equipped with advanced sensors to express their intents. The communications between the construction robots and workers could be established by building reliable human-robot interfaces. This kind of construction paradigm has the potential to be extended to space construction, underwater construction, etc.

CHAPTER NINE: CONCLUSIONS AND FUTURE WORK

With the evolution of Industry 4.0, construction robotics is emerging as a driven force for transforming traditional and potentially hazardous construction processes into more automated and safer ones. However, robots and autonomous machines have not been widely adopted on construction sites. An intuitive and accurate human intent representation can help contribute to promoting the adoption of construction robots. It could allow the robotics systems to conduct multiple tasks guided by construction workers. Also, such representation helps to build a safe environment for worker-robot collaboration.

In this study, a context-aware human intent representation is proposed to support human-robot collaboration on construction sites. It consists of three components: recognition building, object-enhanced interaction and machine-aware collaboration. In the first component recognition building, a novel vision-based method is developed to achieve gesture recognition. Then, a novel wearable sensors-based method is developed. The vision-based method and the sensor-based method are compared to find their advantages and disadvantages. In the second component object-enhanced interaction, a novel object-aware method is proposed to improve the interaction efficiency for human-robot collaboration in construction, integrating first-person vision and gesture recognition. The proposed method models the user's gaze using an attention module to concentrate on the important spatiotemporal regions of first-person video for object context extraction. In the third component machine-aware collaboration, a novel machine-aware hand gesture recognition method is developed as a human-robot interface for use on construction sites having multiple types of machines. The developed method relies on an eye tracker to visually detect and track construction machines in the first-person view and indicate which machine the user intends to interact with.

The above methods have been evaluated and tested by experiments across various construction sites. The results of these evaluations have shown that the proposed methods could effectively capture and interpret worker intents with context awareness, thereby enhancing human-robot collaboration on construction sites. The anticipated contributions of these methods include enhancing interaction efficiency with construction robots, reducing onsite safety concerns, optimizing the design and implementation of construction robots, and fostering the adoption of robots in construction, etc.

Future work will focus on the following four aspects. First, more construction sites, subjects, equipment, tasks and types of hand gestures would be included in the dataset to make the training and testing of the proposed methods more robust to various scenarios (e.g., worker distractions, motion outliers, habits of performing gestures). Second, it will investigate how to overcome the electronics, computation and communication challenges to automate the proposed methods. For example, the Tobii Glasses and Tap sensor are connected with the computer through WLAN and Bluetooth, respectively. These two connection modes both have a limited transmission distance (e.g., around 200 meters for WLAN, around 100 meters for Bluetooth) which would be an inadequate range in a large construction site. Third, the sensor/data fusion could be incorporated into the methods to increase the reliability of message communication between workers and machines. In cases where gesture recognition cannot be successfully performed, other communication channels (e.g., voice commands) could provide a supplementary means for human-robot interaction. Fourth, the feedbacks from workers towards novel human-robot interfaces will be collected to investigate whether workers trust in construction robots and the potential safety implications during the human-robot interactions on construction sites.

REFERENCES

[1] P. Teicholz, Labor-Productivity Declines in the Construction Industry: Causes and Remedies (Another Look) (Accessed on 29 of July, 2014), AECbytes Viewp. (2013).

[2] X. Li, W. Yi, H.L. Chi, X. Wang, A.P.C. Chan, A critical review of virtual and augmented reality (VR/AR) applications in construction safety, Autom. Constr. (2018). https://doi.org/10.1016/j.autcon.2017.11.003.

[3] J.M. Davila Delgado, L. Oyedele, A. Ajayi, L. Akanbi, O. Akinade, M. Bilal, H. Owolabi, Robotics and automated systems in construction: Understanding industry-specific challenges for adoption, J. Build. Eng. (2019). https://doi.org/10.1016/j.jobe.2019.100868.

[4] P. Pradhananga, M. ElZomor, G. Santi Kasabdji, Identifying the Challenges to Adopting Robotics in the US Construction Industry, J. Constr. Eng. Manag. (2021). https://doi.org/10.1061/(asce)co.1943-7862.0002007.

[5] AAB Robotics, ABB Robotics 2021 Construction Survey, (2021). https://express.adobe.com/page/5QaTFLrupXbYh/ (accessed October 11, 2023).

[6] M.J. Kim, H.L. Chi, X. Wang, L. Ding, Automation and Robotics in Construction and Civil Engineering, J. Intell. Robot. Syst. Theory Appl. (2015). https://doi.org/10.1007/s10846-015-0252-9.

[7] International Federation of Robotics, World Robotics Report, (2023). https://ifr.org/img/worldrobotics/2023_WR_extended_version.pdf (accessed October 11, 2023).

[8] N. Melenbrink, J. Werfel, A. Menges, On-site autonomous construction robots: Towards unsupervised building, Autom. Constr. (2020). https://doi.org/10.1016/j.autcon.2020.103312.

[9] H. Ardiny, S. Witwicki, F. Mondada, Construction automation with autonomous mobile robots: A review, in: Int. Conf. Robot. Mechatronics, ICROM 2015, 2015. https://doi.org/10.1109/ICRoM.2015.7367821.

[10] B. García de Soto, I. Agustí-Juan, J. Hunhevicz, S. Joss, K. Graser, G. Habert, B.T. Adey, Productivity of digital fabrication in construction: Cost and time analysis of a robotically

built wall, Autom. Constr. (2018). https://doi.org/10.1016/j.autcon.2018.04.004.

[11] J.G. Martinez, M. Gheisari, L.F. Alarcón, UAV Integration in Current Construction Safety Planning and Monitoring Processes: Case Study of a High-Rise Building Construction Project in Chile, J. Manag. Eng. (2020). https://doi.org/10.1061/(asce)me.1943-5479.0000761.

[12] F. Bademosi, R.R.A. Issa, Factors Influencing Adoption and Integration of Construction Robotics and Automation Technology in the US, J. Constr. Eng. Manag. (2021). https://doi.org/10.1061/(asce)co.1943-7862.0002103.

[13] Y. Liu, M. Habibnezhad, H. Jebelli, Brain-computer interface for hands-free teleoperation of construction robots, Autom. Constr. (2021). https://doi.org/10.1016/j.autcon.2020.103523.

[14] A. Vysocky, P. Novak, Human - Robot collaboration in industry, MM Sci. J. (2016). https://doi.org/10.17973/MMSJ.2016_06_201611.

[15] A. Ajoudani, A.M. Zanchettin, S. Ivaldi, A. Albu-Schäffer, K. Kosuge, O. Khatib, Progress and prospects of the human–robot collaboration, Auton. Robots. (2018). https://doi.org/10.1007/s10514-017-9677-2.

[16] C. Bartneck, D. Kulić, E. Croft, S. Zoghbi, Measurement instruments for the anthropomorphism, animacy, likeability, perceived intelligence, and perceived safety of robots, Int. J. Soc. Robot. (2009). https://doi.org/10.1007/s12369-008-0001-3.

[17] S. You, J.H. Kim, S.H. Lee, V. Kamat, L.P. Robert, Enhancing perceived safety in human–robot collaborative construction using immersive virtual environments, Autom. Constr. (2018). https://doi.org/10.1016/j.autcon.2018.09.008.

[18] V. Villani, F. Pini, F. Leali, C. Secchi, Survey on human–robot collaboration in industrial settings: Safety, intuitive interfaces and applications, Mechatronics. (2018). https://doi.org/10.1016/j.mechatronics.2018.02.009.

[19] J. Huschilt, L. Clune, The use of socially assistive robots for dementia care, J. Gerontol. Nurs. (2012). https://doi.org/10.3928/00989134-20120911-02.

[20] H. Liu, L. Wang, Collision-free human-robot collaboration based on context awareness,

Robot. Comput. Integr. Manuf. (2021). https://doi.org/10.1016/j.rcim.2020.101997.

[21] R.C. Luo, L. Mai, Human Intention Inference and On-Line Human Hand Motion Prediction for Human-Robot Collaboration, in: IEEE Int. Conf. Intell. Robot. Syst., 2019. https://doi.org/10.1109/IROS40897.2019.8968192.

[22] BigRentz Inc, How Construction Robots Will Change The Industry, (2020). https://www.bigrentz.com/blog/construction-robots (accessed January 12, 2022).

[23] E. Gambao, C. Balaguer, F. Gebhart, Robot assembly system for computer-integrated construction, Autom. Constr. (2000). https://doi.org/10.1016/S0926-5805(00)00059-5.

[24] T. Greaves, B. Jenkins, Capturing Existing Conditions with Terrestrial Laser Scanning: a Report on Opportunities, Challenges and Best Practices for Owners, Operators., Eng. Contract. Surv. Built Assets Civ. Infrastruct. (2004).

[25] Alex M., The 4 Types of FFF / FDM 3D Printer Explained, 3D Print. NEWS NEWS. (2017). https://www.3dnatives.com/en/four-types-fdm-3d-printers140620174/#! (accessed January 13, 2022).

[26] R. Duballet, O. Baverel, J. Dirrenberger, Classification of building systems for concrete 3D printing, Autom. Constr. (2017). https://doi.org/10.1016/j.autcon.2017.08.018.

[27] S. Lim, R.A. Buswell, T.T. Le, S.A. Austin, A.G.F. Gibb, T. Thorpe, Developments in construction-scale additive manufacturing processes, Autom. Constr. (2012). https://doi.org/10.1016/j.autcon.2011.06.010.

[28] F. Bos, R. Wolfs, Z. Ahmed, T. Salet, Additive manufacturing of concrete in construction: potentials and challenges of 3D concrete printing, Virtual Phys. Prototyp. (2016). https://doi.org/10.1080/17452759.2016.1209867.

[29] D. Herzog, V. Seyda, E. Wycisk, C. Emmelmann, Additive manufacturing of metals, Acta Mater. (2016). https://doi.org/10.1016/j.actamat.2016.07.019.

[30] S.C. Ligon, R. Liska, J. Stampfl, M. Gurr, R. Mülhaupt, Polymers for 3D Printing and Customized Additive Manufacturing, Chem. Rev. (2017). https://doi.org/10.1021/acs.chemrev.7b00074.

[31] Process Solutions Inc., What are the Different Types of Collaborative Robots?, (2018).

https://processsolutions.com/what-are-the-different-types-of-collaborative-robots/ (accessed January 13, 2022).

[32] M. Gautam, H. Fagerlund, B. Greicevci, F. Christophe, J. Havula, Collaborative Robotics in Construction: A Test Case on Screwing Gypsum Boards on Ceiling, in: Proc. 2020 5th Int. Conf. Green Technol. Sustain. Dev. GTSD 2020, 2020. https://doi.org/10.1109/GTSD50082.2020.9303061.

[33] W. Jiang, Y. Zhou, L. Ding, C. Zhou, X. Ning, UAV-based 3D reconstruction for hoist site mapping and layout planning in petrochemical construction, Autom. Constr. (2020). https://doi.org/10.1016/j.autcon.2020.103137.

[34] S. Sun, A new method for monitoring machinery movement using an Unmanned Aerial Vehicle (UAV) system, Univeristy of Twente, 2019. http://essay.utwente.nl/79827/.

[35] R.R.S. de Melo, D.B. Costa, J.S. Álvares, J. Irizarry, Applicability of unmanned aerial system (UAS) for safety inspection on construction sites, Saf. Sci. (2017). https://doi.org/10.1016/j.ssci.2017.06.008.

[36] M.R. Srnoyachki, Automated Drilling Application for Autonomous Airfield Runway Surveying Vehicles: System Design and Validation, University of Dayton, 2018. https://etd.ohiolink.edu/apexprod/rws_olink/r/1501/10?clear=10&p10_accession_num=dayton1544537004159348.

[37] J. Thangavelautham, K. Law, T. Fu, N. Abu El Samid, A.D.S. Smith, G.M.T. D'Eleuterio, Autonomous multirobot excavation for lunar applications, Robotica. (2017). https://doi.org/10.1017/S0263574717000017.

[38] H. Fernando, J.A. Marshall, J. Larsson, Iterative Learning-Based Admittance Control for Autonomous Excavation, J. Intell. Robot. Syst. Theory Appl. (2019). https://doi.org/10.1007/s10846-019-00994-3.

[39] S. Dadhich, U. Bodin, U. Andersson, Key challenges in automation of earth-moving machines, Autom. Constr. (2016). https://doi.org/10.1016/j.autcon.2016.05.009.

[40] Q.P. Ha, L. Yen, C. Balaguer, Robotic autonomous systems for earthmoving in military applications, Autom. Constr. (2019). https://doi.org/10.1016/j.autcon.2019.102934.

[41] AIST, Development of a Humanoid Robot Prototype, HRP-5P, Capable of Heavy Labor, (2018). https://www.aist.go.jp/aist_e/list/latest_research/2018/20181116/en20181116.html (accessed January 26, 2022).

[42] E. Kisliuk, NASA's R5, NASA. (2017). https://www.nasa.gov/feature/r5/ (accessed January 26, 2022).

[43] Greg Nichols, Before heading to Mars, NASA's Valkyrie humanoid lands at MIT, ZDNet. (2016). https://www.zdnet.com/article/before-heading-to-mars-nasas-valkyrie-humanoid-lands-at-mit/ (accessed January 26, 2022).

[44] VEX Robotics, What are Industrial Robots?, (2021). https://education.vex.com/stemlabs/workcell/stemlab/industrial-robotics/what-are-industrial-robots (accessed January 13, 2022).

[45] G. Plastikmedia, Advantages and disadvantages of industrial robots, (2018). https://www.plastikmedia.co.uk/advantages-disadvantages-of-industrial-robots/ (accessed January 13, 2022).

[46] Process Solutions Inc., The Benefits of Industrial Robot Automation, (2020). https://processsolutions.com/the-benefits-of-industrial-robotic-automation/ (accessed January 13, 2022).

[47] Wingtra, Why and how to use drones in construction and infrastructure, (2020). https://wingtra.com/drone-mapping-applications/drones-in-construction-and-infrastructure/ (accessed January 13, 2022).

[48] BigRentz Inc, 6 Profitable Ways Drones in Construction Are Changing Projects, (2018). https://www.bigrentz.com/blog/drones-construction (accessed January 19, 2022).

[49] F. Moreno-Navarro, G.R. Iglesias, M.C. Rubio-Gámez, Encoded asphalt materials for the guidance of autonomous vehicles, Autom. Constr. (2019). https://doi.org/10.1016/j.autcon.2018.12.004.

[50] J.M. Anderson, Autonomous vehicle technology: How to best realize its social benefits, in: 21st World Congr. Intell. Transp. Syst. ITSWC 2014 Reinventing Transp. Our Connect. World, 2014. https://doi.org/10.7249/rb9755.

[51] G. Silberg, J. Plessers, R. Wallace, C. Brower, G. Matuszak, D. Subramanian, Self-driving cars: The next revolution, (2012). https://assets.kpmg/content/dam/kpmg/pdf/2015/10/self-driving-cars-next-revolution_new.pdf (accessed January 21, 2022).

[52] M. Greenwood, Robot Laborer Could Do the Risky and Tiring Work of Human Laborers, (2018). https://www.engineering.com/story/robot-laborer-could-do-the-risky-and-tiring-work-of-human-laborers (accessed January 26, 2022).

[53] P. Majaranta, A. Bulling, Eye Tracking and Eye-Based Human–Computer Interaction, in: 2014. https://doi.org/10.1007/978-1-4471-6392-3_3.

[54] M.A. Goodrich, A.C. Schultz, Human-robot interaction: A survey, Found. Trends Human-Computer Interact. (2007). https://doi.org/10.1561/1100000005.

[55] J. Berg, S. Lu, Review of Interfaces for Industrial Human-Robot Interaction, Curr. Robot. Reports. (2020). https://doi.org/10.1007/s43154-020-00005-6.

[56] M.S. Ryoo, T.J. Fuchs, L. Xia, J.K. Aggarwal, L. Matthies, Robot-Centric Activity Prediction from First-Person Videos: What Will They Do to Me?, in: ACM/IEEE Int. Conf. Human-Robot Interact., 2015. https://doi.org/10.1145/2696454.2696462.

[57] Y. Kong, Z. Tao, Y. Fu, Deep sequential context networks for action prediction, in: Proc. - 30th IEEE Conf. Comput. Vis. Pattern Recognition, CVPR 2017, 2017. https://doi.org/10.1109/CVPR.2017.390.

[58] Y. Kong, F. Yun, Human action recognition and prediction: A survey, ArXiv Prepr. (2018).

[59] M. Pei, Y. Jia, S.C. Zhu, Parsing video events with goal inference and intent prediction, in: Proc. IEEE Int. Conf. Comput. Vis., 2011. https://doi.org/10.1109/ICCV.2011.6126279.

[60] K. Li, J. Hu, Y. Fu, Modeling complex temporal composition of actionlets for activity prediction, in: Lect. Notes Comput. Sci. (Including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics), 2012. https://doi.org/10.1007/978-3-642-33718-5_21.

[61] K. Li, Y. Fu, Prediction of human activity by discovering temporal sequence patterns, IEEE Trans. Pattern Anal. Mach. Intell. (2014). https://doi.org/10.1109/TPAMI.2013.2297321.

[62] H.S. Koppula, A. Saxena, Anticipating Human Activities Using Object Affordances for Reactive Robotic Response, IEEE Trans. Pattern Anal. Mach. Intell. (2016).

https://doi.org/10.1109/TPAMI.2015.2430335.

[63] G. Yasmeen, S. Arun, J.N. Swaminathan, S.A.K. Jilani, S. Asif, Efficient Hand Gesture Recognition for Traffic Control System Using ti Sensor Tag, in: 2018 Int. Conf. Comput. Commun. Informatics, ICCCI 2018, 2018. https://doi.org/10.1109/ICCCI.2018.8441483.

[64] Z. Lu, X. Chen, Q. Li, X. Zhang, P. Zhou, A hand gesture recognition framework and wearable gesture-based interaction prototype for mobile devices, IEEE Trans. Human-Machine Syst. (2014). https://doi.org/10.1109/THMS.2014.2302794.

[65] O. Köpüklü, A. Gunduz, N. Kose, G. Rigoll, Real-time hand gesture detection and classification using convolutional neural networks, in: Proc. - 14th IEEE Int. Conf. Autom. Face Gesture Recognition, FG 2019, 2019. https://doi.org/10.1109/FG.2019.8756576.

[66] P. Molchanov, X. Yang, S. Gupta, K. Kim, S. Tyree, J. Kautz, Online Detection and Classification of Dynamic Hand Gestures with Recurrent 3D Convolutional Neural Networks, in: Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit., 2016. https://doi.org/10.1109/CVPR.2016.456.

[67] T.-Y. Pan, W.-L. Tsai, C.-Y. Chang, C.-W. Yeh, M.-C. Hu, A Hierarchical Hand Gesture Recognition Framework for Sports Referee Training-Based EMG and Accelerometer Sensors, IEEE Trans. Cybern. (2020). https://doi.org/10.1109/tcyb.2020.3007173.

[68] O. Koller, C. Camgoz, H. Ney, R. Bowden, Weakly Supervised Learning with Multi-Stream CNN-LSTM-HMMs to Discover Sequential Parallelism in Sign Language Videos, IEEE Trans. Pattern Anal. Mach. Intell. (2019). https://doi.org/10.1109/tpami.2019.2911077.

[69] A.A. Neacsu, G. Cioroiu, A. Radoi, C. Burileanu, Automatic EMG-based hand gesture recognition system using time-domain descriptors and fully-connected neural networks, in: 2019 42nd Int. Conf. Telecommun. Signal Process. TSP 2019, 2019. https://doi.org/10.1109/TSP.2019.8768831.

[70] B. Fang, Q. Lv, J. Shan, F. Sun, H. Liu, D. Guo, Y. Zhao, Dynamic gesture recognition using inertial sensors-based data gloves, in: 2019 4th IEEE Int. Conf. Adv. Robot. Mechatronics, ICARM 2019, 2019. https://doi.org/10.1109/ICARM.2019.8834314.

[71] E. Ohn-Bar, M.M. Trivedi, Hand gesture recognition in real time for automotive interfaces: A multimodal vision-based approach and evaluations, IEEE Trans. Intell. Transp. Syst.

(2014). https://doi.org/10.1109/TITS.2014.2337331.

[72] H. Wang, D. Oneata, J. Verbeek, C. Schmid, A Robust and Efficient Video Representation for Action Recognition, Int. J. Comput. Vis. (2016). https://doi.org/10.1007/s11263-015-0846-5.

[73] J. Wan, G. Guo, S.Z. Li, Explore Efficient Local Features from RGB-D Data for One-Shot Learning Gesture Recognition, IEEE Trans. Pattern Anal. Mach. Intell. (2016). https://doi.org/10.1109/TPAMI.2015.2513479.

[74] S.G.M. Almeida, F.G. Guimarães, J. Arturo Ramírez, Feature extraction in Brazilian Sign Language Recognition based on phonological structure and using RGB-D sensors, Expert Syst. Appl. (2014). https://doi.org/10.1016/j.eswa.2014.05.024.

[75] W. Ahmed, K. Chanda, S. Mitra, Vision based Hand Gesture Recognition using Dynamic Time Warping for Indian Sign Language, in: Proc. - 2016 Int. Conf. Inf. Sci. ICIS 2016, 2017. https://doi.org/10.1109/INFOSCI.2016.7845312.

[76] A. Memo, P. Zanuttigh, Head-mounted gesture controlled interface for human-computer interaction, Multimed. Tools Appl. (2018). https://doi.org/10.1007/s11042-016-4223-3.

[77] Q. Miao, Y. Li, W. Ouyang, Z. Ma, X. Xu, W. Shi, X. Cao, Multimodal Gesture Recognition Based on the ResC3D Network, in: Proc. - 2017 IEEE Int. Conf. Comput. Vis. Work. ICCVW 2017, 2017. https://doi.org/10.1109/ICCVW.2017.360.

[78] Y. Liao, P. Xiong, W. Min, W. Min, J. Lu, Dynamic Sign Language Recognition Based on Video Sequence with BLSTM-3D Residual Networks, IEEE Access. (2019). https://doi.org/10.1109/ACCESS.2019.2904749.

[79] X. Wang, Z. Zhu, Vision-Based Hand Signal Recognition in Construction: A Feasibility Study, Autom. Constr. 125 (2021) 103625. https://doi.org/10.1016/j.autcon.2021.103625.

[80] X. Wang, Z. Zhu, Vision–Based Framework for Automatic Interpretation of Construction Workers' Hand Gestures, Autom. Constr. 130 (2021) 103872. https://doi.org/10.1016/j.autcon.2021.103872.

[81] J. Huang, W. Zhou, Q. Zhang, H. Li, W. Li, Video-based sign language recognition without temporal segmentation, in: 32nd AAAI Conf. Artif. Intell. AAAI 2018, 2018.

https://ojs.aaai.org/index.php/AAAI/article/view/11903.

[82] X. Zhang, X. Chen, Y. Li, V. Lantz, K. Wang, J. Yang, A framework for hand gesture recognition based on accelerometer and EMG sensors, IEEE Trans. Syst. Man, Cybern. Part ASystems Humans. (2011). https://doi.org/10.1109/TSMCA.2011.2116004.

[83] U. Côté-Allard, C.L. Fall, A. Drouin, A. Campeau-Lecours, C. Gosselin, K. Glette, F. Laviolette, B. Gosselin, Deep Learning for Electromyographic Hand Gesture Signal Classification Using Transfer Learning, IEEE Trans. Neural Syst. Rehabil. Eng. (2019). https://doi.org/10.1109/TNSRE.2019.2896269.

[84] M. Kim, J. Cho, S. Lee, Y. Jung, Imu sensor-based hand gesture recognition for human-machine interfaces, Sensors (Switzerland). (2019). https://doi.org/10.3390/s19183827.

[85] D. Jirak, S. Tietz, H. Ali, S. Wermter, Echo State Networks and Long Short-Term Memory for Continuous Gesture Recognition: a Comparative Study, Cognit. Comput. (2020). https://doi.org/10.1007/s12559-020-09754-0.

[86] G. Yuan, X. Liu, Q. Yan, S. Qiao, Z. Wang, L. Yuan, Hand Gesture Recognition Using Deep Feature Fusion Network Based on Wearable Sensors, IEEE Sens. J. (2021). https://doi.org/10.1109/JSEN.2020.3014276.

[87] X. Wang, D. Veeramani, Z. Zhu, Wearable Sensors-Based Hand Gesture Recognition for Human-Robot Collaboration in Construction, IEEE Sens. J. (2022) In Revision.

[88] D.Y. Cho, M.K. Kang, Human gaze-aware attentive object detection for ambient intelligence, Eng. Appl. Artif. Intell. 106 (2021) 104471. https://doi.org/10.1016/j.engappai.2021.104471.

[89] hotjar, Eye-tracking definition, (n.d.). https://www.hotjar.com/conversion-rate-optimization/glossary/eye-tracking/ (accessed February 2, 2022).

[90] G. Zhang, J.P. Hansen, K. Minakata, A. Alapetite, Z. Wang, Eye-Gaze-Controlled Telepresence Robots for People with Motor Disabilities, in: ACM/IEEE Int. Conf. Human-Robot Interact., 2019. https://doi.org/10.1109/HRI.2019.8673093.

[91] J. Zhang, Y. Wu, H. Huang, G. Hou, A New Human Eye Tracking Method Based on Tracking Module Feedback TLD Algorithm, in: Proc. - 20th Int. Conf. High Perform.

Comput. Commun. 16th Int. Conf. Smart City 4th Int. Conf. Data Sci. Syst. HPCC/SmartCity/DSS 2018, 2019. https://doi.org/10.1109/HPCC/SmartCity/DSS.2018.00071.

[92] T. Santini, W. Fuhl, E. Kasneci, PuReST: Robust pupil tracking for real-time pervasive eye tracking, in: Eye Track. Res. Appl. Symp., 2018. https://doi.org/10.1145/3204493.3204578.

[93] R.G. Bozomitu, A. Păsăricǎ, D. Tărniceriu, C. Rotariu, Development of an eye tracking-based human-computer interface for real-time applications, Sensors (Switzerland). (2019). https://doi.org/10.3390/s19163630.

[94] A. Laddi, N.R. Prakash, Eye gaze tracking based directional control interface for interactive applications, Multimed. Tools Appl. 78 (2019) 31215–31230. https://doi.org/10.1007/s11042-019-07940-3.

[95] Tobii Inc., Tobii Pro Glasses 3, (2021). https://www.tobiipro.com/product-listing/tobii-pro-glasses-3/ (accessed February 3, 2022).

[96] Pupil Labs, Pupil Core, (2021). https://pupil-labs.com/products/core/ (accessed February 3, 2022).

[97] M. Barz, D. Sonntag, Gaze-guided object classification using deep neural networks for attention-based computing, in: UbiComp 2016 Adjun. - Proc. 2016 ACM Int. Jt. Conf. Pervasive Ubiquitous Comput., 2016. https://doi.org/10.1145/2968219.2971389.

[98] M. Kassner, W. Patera, A. Bulling, Pupil: An open source platform for pervasive eye tracking and mobile gaze-based interaction, in: UbiComp 2014 - Adjun. Proc. 2014 ACM Int. Jt. Conf. Pervasive Ubiquitous Comput., 2014. https://doi.org/10.1145/2638728.2641695.

[99] J.H. Kim, S.J. Choi, J.W. Jeong, Watch do: A smart IoT interaction system with object detection and gaze estimation, IEEE Trans. Consum. Electron. (2019). https://doi.org/10.1109/TCE.2019.2897758.

[100] N. Khosravan, H. Celik, B. Turkbey, R. Cheng, E. McCreedy, M. McAuliffe, S. Bednarova, E. Jones, X. Chen, P. Choyke, B. Wood, U. Bagci, Gaze2Segment: A pilot study for integrating eye-tracking technology into medical image segmentation, in: Lect. Notes Comput. Sci. (Including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics), 2017.

https://doi.org/10.1007/978-3-319-61188-4_9.

[101] R. Cantrell, P. Schermerhorn, M. Scheutz, Learning actions from human-robot dialogues, in: Proc. - IEEE Int. Work. Robot Hum. Interact. Commun., 2011. https://doi.org/10.1109/ROMAN.2011.6005199.

[102] B. Shen, D. Inkpen, Speech intent recognition for robots, in: Proc. - 2016 3rd Int. Conf. Math. Comput. Sci. Ind. MCSI 2016, 2017. https://doi.org/10.1109/MCSI.2016.042.

[103] C. Deuerlein, M. Langer, J. Seßner, P. Heß, J. Franke, Human-robot-interaction using cloud-based speech recognition systems, in: Procedia CIRP, 2020. https://doi.org/10.1016/j.procir.2020.05.214.

[104] P.G. Shivakumar, M. Yang, P. Georgiou, Spoken language intent detection using Confusion2Vec, in: Proc. Annu. Conf. Int. Speech Commun. Assoc. INTERSPEECH, 2019. https://doi.org/10.21437/Interspeech.2019-2226.

[105] J. Brawer, O. Mangin, A. Roncone, S. Widder, B. Scassellati, Situated Human-Robot Collaboration: Predicting intent from grounded natural language, in: IEEE Int. Conf. Intell. Robot. Syst., 2018. https://doi.org/10.1109/IROS.2018.8593942.

[106] Y. Huang, H.K. Kuo, S. Thomas, Z. Kons, K. Audhkhasi, B. Kingsbury, R. Hoory, M. Picheny, Leveraging Unpaired Text Data for Training End-To-End Speech-to-Intent Systems, in: ICASSP, IEEE Int. Conf. Acoust. Speech Signal Process. - Proc., 2020. https://doi.org/10.1109/ICASSP40776.2020.9053281.

[107] L. Wang, R. Gao, J. Váncza, J. Krüger, X. V. Wang, S. Makris, G. Chryssolouris, Symbiotic human-robot collaborative assembly, CIRP Ann. (2019). https://doi.org/10.1016/j.cirp.2019.05.002.

[108] P. Wang, H. Liu, L. Wang, R.X. Gao, Deep learning-based human motion recognition for predictive context-aware human-robot collaboration, CIRP Ann. (2018). https://doi.org/10.1016/j.cirp.2018.04.066.

[109] C. Lenz, Context-aware human-robot collaboration as a basis for future cognitive factories, Doctoral Thesis, Technische Universität München, 2011.

[110] Z. Liu, Q. Liu, W. Xu, Z. Liu, Z. Zhou, J. Chen, Deep learning-based human motion

prediction considering context awareness for human-robot collaboration in manufacturing, in: Procedia CIRP, 2019. https://doi.org/10.1016/j.procir.2019.04.080.

[111] Q. Liu, Z. Liu, B. Xiong, W. Xu, Y. Liu, Deep reinforcement learning-based safe interaction for industrial human-robot collaboration using intrinsic reward function, Adv. Eng. Informatics. (2021). https://doi.org/10.1016/j.aei.2021.101360.

[112] A. Hata, R. Inam, K. Raizer, S. Wang, E. Cao, AI-based Safety Analysis for Collaborative Mobile Robots, in: IEEE Int. Conf. Emerg. Technol. Fact. Autom. ETFA, 2019. https://doi.org/10.1109/ETFA.2019.8869263.

[113] S.H. Choi, K.B. Park, D.H. Roh, J.Y. Lee, M. Mohammed, Y. Ghasemi, H. Jeong, An integrated mixed reality system for safety-aware human-robot collaboration using deep learning and digital twin generation, Robot. Comput. Integr. Manuf. (2022). https://doi.org/10.1016/j.rcim.2021.102258.

[114] F. Junming, Z. Pai, S. Li, Vision-based holistic scene understanding towards proactive human–robot collaboration, Robot. Comput. Manuf. 75 (2022). https://doi.org/10.1016/j.rcim.2021.102304.

[115] L. Bruckschen, N. Dengler, M. Bennewitz, Human motion prediction based on object interactions, in: 2019 Eur. Conf. Mob. Robot. ECMR 2019 - Proc., 2019. https://doi.org/10.1109/ECMR.2019.8870963.

[116] Z. Hu, J. Pan, T. Fan, R. Yang, D. Manocha, Safe Navigation with Human Instructions in Complex Scenes, IEEE Robot. Autom. Lett. (2019). https://doi.org/10.1109/LRA.2019.2893432.

[117] J. Mainprice, D. Berenson, Human-robot collaborative manipulation planning using early prediction of human motion, in: IEEE Int. Conf. Intell. Robot. Syst., 2013. https://doi.org/10.1109/IROS.2013.6696368.

[118] E. Corona, A. Pumarola, G. Alenya, F. Moreno-Noguer, Context-Aware Human Motion Prediction, in: Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit., 2020. https://doi.org/10.1109/CVPR42600.2020.00702.

[119] H. Liu, L. Wang, Human motion prediction for human-robot collaboration, J. Manuf. Syst. (2017). https://doi.org/10.1016/j.jmsy.2017.04.009.

[120] Y. Tang, L. Ma, W. Liu, W.S. Zheng, Long-term human motion prediction by modeling motion context and enhancing motion dynamic, in: IJCAI Int. Jt. Conf. Artif. Intell., 2018. https://doi.org/10.24963/ijcai.2018/130.

[121] G. Benitez-Garcia, J. Olivares-Mercado, G. Sanchez-Perez, K. Yanai, IPN Hand: A Video Dataset and Benchmark for Real-Time Continuous Hand Gesture Recognition, IEEE Conf. Comput. Commun. Work. (2020). https://doi.org/10.1109/ICPR48806.2021.9412317.

[122] Y. Zhang, C. Cao, J. Cheng, H. Lu, EgoGesture: A New Dataset and Benchmark for Egocentric Hand Gesture Recognition, IEEE Trans. Multimed. (2018). https://doi.org/10.1109/TMM.2018.2808769.

[123] N. Wojke, A. Bewley, D. Paulus, Simple online and realtime tracking with a deep association metric, in: Proc. - Int. Conf. Image Process. ICIP, 2018. https://doi.org/10.1109/ICIP.2017.8296962.

[124] J. Redmon, A. Farhadi, YOLOv3: An incremental improvement, ArXiv. (2018). https://arxiv.org/abs/1804.02767.

[125] F. Wu, G. Jin, M. Gao, Z. He, Y. Yang, Helmet detection based on improved YOLO V3 deep model, in: Proc. 2019 IEEE 16th Int. Conf. Networking, Sens. Control. ICNSC 2019, 2019. https://doi.org/10.1109/ICNSC.2019.8743246.

[126] X. Luo, H. Li, H. Wang, Z. Wu, F. Dai, D. Cao, Vision-based detection and visualization of dynamic workspaces, Autom. Constr. (2019). https://doi.org/10.1016/j.autcon.2019.04.001.

[127] M. Al-Hussein, M. Athar Niaz, H. Yu, H. Kim, Integrating 3D visualization and simulation for tower crane operations on construction sites, Autom. Constr. (2006). https://doi.org/10.1016/j.autcon.2005.07.007.

[128] J. Yang, P.A. Vela, J. Teizer, Z.K. Shi, Vision-based crane tracking for understanding construction activity, in: Congr. Comput. Civ. Eng. Proc., 2011. https://doi.org/10.1061/41182(416)32.

[129] The American Society of Mechanical Engineers, Safety Standard for Cableways, Cranes, Derricks, Hoists, Hooks, Jacks, and Slings, 2012.

[130] National Commission for the Certification of Crane Operators, Signalperson Reference Manual, (2014). https://www.nccco.org/docs/default-source/reference-materials-2014/sgp_rm_081213a.pdf?sfvrsn=2 (accessed September 1, 2020).

[131] Stereolabs, ZED 2-AI Stereo Camera, (2019). https://www.stereolabs.com/zed-2/ (accessed May 19, 2020).

[132] J. Materzynska, G. Berger, I. Bax, R. Memisevic, The jester dataset: A large-scale video dataset of human gestures, in: Proc. - 2019 Int. Conf. Comput. Vis. Work. ICCVW 2019, 2019. https://doi.org/10.1109/ICCVW.2019.00349.

[133] J. Davis, M. Goadrich, The relationship between precision-recall and ROC curves, in: ACM Int. Conf. Proceeding Ser., 2006. https://doi.org/10.1145/1143844.1143874.

[134] A. Khan, N. Hammerla, S. Mellor, T. Plötz, Optimising sampling rates for accelerometer-based human activity recognition, Pattern Recognit. Lett. (2016). https://doi.org/10.1016/j.patrec.2016.01.001.

[135] X. Yang, A. Dinh, L. Chen, Implementation of a wearerable real-time system for physical activity recognition based on naive bayes classifier, in: ICBBT 2010 - 2010 Int. Conf. Bioinforma. Biomed. Technol., 2010. https://doi.org/10.1109/ICBBT.2010.5479000.

[136] Z. Wang, W. Yan, T. Oates, Time series classification from scratch with deep neural networks: A strong baseline, in: Proc. Int. Jt. Conf. Neural Networks, 2017. https://doi.org/10.1109/IJCNN.2017.7966039.

[137] H. Ismail Fawaz, G. Forestier, J. Weber, L. Idoumghar, P.A. Muller, Deep learning for time series classification: a review, Data Min. Knowl. Discov. (2019). https://doi.org/10.1007/s10618-019-00619-1.

[138] D. Bahdanau, K.H. Cho, Y. Bengio, Neural machine translation by jointly learning to align and translate, in: 3rd Int. Conf. Learn. Represent. ICLR 2015 - Conf. Track Proc., 2015.

[139] Tap Systems Inc., Meet Tap, (2021). https://www.tapwithus.com/.

[140] S. Jiang, P. Kang, X. Song, B. Lo, P. Shull, Emerging Wearable Interfaces and Algorithms for Hand Gesture Recognition: A Survey, IEEE Rev. Biomed. Eng. (2022). https://doi.org/10.1109/RBME.2021.3078190.

[141] Y. Xue, Y. Yu, K. Yin, P. Li, S. Xie, Z. Ju, Human In-Hand Motion Recognition Based on Multi-Modal Perception Information Fusion, IEEE Sens. J. (2022). https://doi.org/10.1109/JSEN.2022.3148992.

[142] S. Qiu, H. Zhao, N. Jiang, Z. Wang, L. Liu, Y. An, H. Zhao, X. Miao, R. Liu, G. Fortino, Multi-sensor information fusion based on machine learning for real applications in human activity recognition: State-of-the-art and research challenges, Inf. Fusion. (2022). https://doi.org/10.1016/j.inffus.2021.11.006.

[143] R. Roelofs, Measuring Generalization and Overfitting in Machine Learning, Dr. Diss. UC Berkeley. (2019). https://escholarship.org/uc/item/6j01x9mz.

[144] M. Lu, D. Liao, Z.N. Li, Learning spatiotemporal attention for egocentric action recognition, in: Proc. - 2019 Int. Conf. Comput. Vis. Work. ICCVW 2019, 2019. https://doi.org/10.1109/ICCVW.2019.00543.

[145] K. He, X. Zhang, S. Ren, J. Sun, Deep residual learning for image recognition, in: Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit., 2016. https://doi.org/10.1109/CVPR.2016.90.

[146] C. Li, X. Wu, N. Zhao, X. Cao, J. Tang, Fusing two-stream convolutional neural networks for RGB-T object tracking, Neurocomputing. (2018). https://doi.org/10.1016/j.neucom.2017.11.068.

[147] C. Feichtenhofer, A. Pinz, A. Zisserman, Convolutional Two-Stream Network Fusion for Video Action Recognition, in: Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit., 2016. https://doi.org/10.1109/CVPR.2016.213.

[148] Electronic Library of Construction Safety and Health, Excavator Hand Signals, (2022). https://elcosh.org/document/1458/d000068/excavator-hand-signals.html (accessed August 1, 2022).

[149] J. Carreira, A. Zisserman, Quo Vadis, action recognition? A new model and the kinetics dataset, in: Proc. - 30th IEEE Conf. Comput. Vis. Pattern Recognition, CVPR 2017, 2017. https://doi.org/10.1109/CVPR.2017.502.

[150] Y. Li, M. Liu, J. Rehg, In the Eye of the Beholder: Gaze and Actions in First Person Video, IEEE Trans. Pattern Anal. Mach. Intell. (2021).

https://doi.org/10.1109/TPAMI.2021.3051319.

[151] A. Bewley, Z. Ge, L. Ott, F. Ramos, B. Upcroft, Simple online and realtime tracking, in: Proc. - Int. Conf. Image Process. ICIP, 2016. https://doi.org/10.1109/ICIP.2016.7533003.

[152] G. Jocher, A. Chaurasia, A. Stoken, J. Borovec, NanoCode012, Y. Kwon, TaoXie, K. Michael, I. Fang, Imyhxy, Lorna, C. Wong, Z. Yifu, A. V, D. Montes, Z. Wang, C. Fati, J. Nadar, Laughing, ultralytics/yolov5: v6.2 - YOLOv5 Classification Models, Apple M1, Reproducibility, ClearML and Deci.ai integrations, Zenodo. (2022). https://doi.org/10.5281/zenodo.7002879.

[153] X. Luo, H. Li, D. Cao, F. Dai, J. Seo, S. Lee, Recognizing Diverse Construction Activities in Site Images via Relevance Networks of Construction-Related Objects Detected by Convolutional Neural Networks, J. Comput. Civ. Eng. (2018). https://doi.org/10.1061/(asce)cp.1943-5487.0000756.

[154] N. Galley, D. Betz, C. Biniossek, Fixation durations - Why are they so highly variable?, in: Adv. Vis. Percept. Res., 2015.

[155] C. Chen, Z. Zhu, A. Hammad, M. Akbarzadeh, Automatic Identification of Idling Reasons in Excavation Operations Based on Excavator–Truck Relationships, J. Comput. Civ. Eng. 35 (2021) 04021015. https://doi.org/10.1061/(ASCE)CP.1943-5487.0000981.

[156] V.I. Pavlovic, R. Sharma, T.S. Huang, Visual interpretation of hand gestures for human-computer interaction: A review, IEEE Trans. Pattern Anal. Mach. Intell. (1997). https://doi.org/10.1109/34.598226.

[157] S. Ankalaki, M.N. Thippeswamy, Static and Dynamic Human Activity Detection Using Multi CNN-ELM Approach, in: Lect. Notes Electr. Eng., 2022. https://doi.org/10.1007/978-981-16-1338-8_18.

[158] A. Swaminathan, Comparative Analysis of Sensor-Based Human Activity Recognition Using Artificial Intelligence, in: Int. Conf. Comput. Intell. Data Sci., Springer, Cham, 2022: pp. 1–17. https://doi.org/10.1007/978-3-031-16364-7_1.