UiT The Arctic University of Norway

Faculty of Engineering Science and Technology
Department of Industrial Engineering

Architectures and technologies for increased agility in small-scale manufacturing systems

Beibei Shu

A dissertation for the degree of Philosophiae Doctor, May 2022

To my beloved parents and wife, Ting Yang.

“In the middle of difficulty lies opportunity.”
–Albert Einstein

Abstract

The rapid booming of industry 4.0 technologies has been boosting further development of industrial manufacturing systems in the recent two decades. An increasing number of disruptive yet enabling technologies are becoming available for industrial applications. However, while the factories becoming more complex, a proliferation of incompatible systems that have been developed by different vendors or suppliers become a huge challenge to enterprises in reaping the technological advantages. A generalized architecture for integrating various manufacturing systems will be valuable as it's effectively and efficiently facilitating technology planning, system designing, implementation, maintenance, and upgrading.

From the Small and Medium-sized Enterprises' (SMEs) perspective, the implementation of advanced industry 4.0 technologies is crucial to their business survival. It seems impossible to develop a system that can help all SMEs businesses, however, when the vital technology can be developed as a module, more SMEs can quickly get the benefit. Therefore, a generalized architecture for the manufacturing system is needed, so the different vital technologies can be developed as modules and combined to form the various manufacturing systems with the same architecture.

There are several novel industry 4.0 technologies that have been studied during this PhD project: in **paper 1**, a digitized production system control method is introduced for system remote monitoring/supervision and reducing the hardware configuration; in **paper 2**, a digital twin module with the simulation is designed to enhance the development of high level Human-Robot Collaboration (HRC) tasks; in **paper 3**, various interaction methods between digital twin with human have been proposed to promote the usage of the digital twin; in **paper 4**, a flexible HRC architecture with its demonstration has been proposed to ease the difficulty of the emerging industry 4.0 technologies' fusion and upgrading; in **paper5**, an industrial robot universal remote control graphical user interface has been proposed and the experiment showed the operator could program the robot regardless of the geographic distance.

This study is directed especially towards SMEs in order to strengthen their business operation and contributes to a sustainable development. The dissertation proposes and develops generalized architectures and selected technologies that can be applied to most current and future manufacturing tasks. The main contributions and effects of my work are:

1. Introducing generalized architectures gives a unified and common framework for enterprises, developers and system integrators to work within. A common language and understanding and a holistic view on the manufacturing operation.
2. Analyzing several Industry 4.0 technologies in terms of availability, complexity and readiness will guide small-scale manufacturing enterprises in their choice of direction when developing their manufacturing system.
3. Presenting several system demonstrations offers a glance at the architecture's flexibility and gives insight in the power of selected technologies.

List of Included Papers

Number	Publications
Paper 1	"Introduction of cyber-physical system in robotized press-brake line for metal industry" Shu, Beibei; Sziebig, Gabor; Solvang, Bjørn. Lecture Notes in Electrical Engineering 2018; Volum 451 (1). ISSN 1876-1100.s 181 - 186.s doi: 10.1007/978-981-10-5768-7_20.
Paper 2	"Human-Robot Collaboration: Task sharing through Virtual Reality" Shu, Beibei; Sziebig, Gabor; Pieskä, Sakari. IEEE conference proceedings 2018 (1) ISBN 978-1-5090-6684-1. ISSN 1553-572X.s 6040 - 6044.s doi: 10.1109/IECON.2018.8591102.
Paper 3	"Architecture for Safe Human-Robot Collaboration: Multi-Modal Communication in Virtual Reality for Efficient Task Execution" Shu, Beibei; Sziebig, Gabor; Pieters, Roel. Proceedings of the IEEE International Symposium on Industrial Electronics 2019. ISSN 2163-5145.s 2297 - 2302.s doi: 10.1109/ISIE.2019.8781372.
Paper 4	"Architecture for task-dependent human-robot collaboration" Shu, Beibei; Solvang, Bjørn. 2021 IEEE/SICE International Symposium on System Integration (SII). doi: 10.1109/IEEECONF49454.2021.9382703.
Paper 5	"Platform independent interface for programming of industrial robots" Shu, Beibei; Arnarson, Halldor; Solvang, Bjørn; Kaarlela, Tero; Pieskä, Sakari. IEEE conference proceedings, 2022 IEEE/SICE International Symposium on System Integration (SII), 2022, pp. 797-802, doi: 10.1109/SII52469.2022.9708905.

In addition to the included papers, I am continuously working on the easy deployment of more edge technologies into the manufacturing industry. The current topics includes digital twin, virtual reality, cloud computing and network technology.

List of additional works

Number	Works
1	"Collaboration with High-Payload Industrial Robots: Simulation for Safety" Shu, Beibei; Sziebig, Gabor. Lecture Notes in Electrical Engineering 2019; Volum 484. ISSN 1876-1100.s 34 - 38.s doi: 10.1007/978-981-13-2375-1_5.
2	"An Introduction of the Role of Virtual Technologies and Digital Twin in Industry 4.0" Azarian, Mohammad; Yu, Hao; Solvang, Wei Deng; Shu, Beibei. Lecture Notes in Electrical Engineering 2020; Volum 634. ISSN 1876-1100.s 258 - 266.s doi: 10.1007/978-981-15-2341-0_32.
3	"The application of open access middleware for cooperation among heterogeneous manufacturing systems" Arnarson, Halldor; Solvang, Bjørn; Shu, Beibei. IEEE conference proceedings 2020 ISBN 978-1-7281-6419-9.s doi: 10.1109/SIMS49386.2020.9121537.
4	"The application of virtual reality in programming of a manufacturing cell" Arnarson, Halldor; Solvang, Bjørn; Shu, Beibei. IEEE conference proceedings 2021 ISBN 978-1-7281-7658-1.s doi: 10.1109/IEEECONF49454.2021.9382657.
5	"Robot cell digital twins as a tool for remote collaboration between organizations" Kaarlela, Tero; Pieskä, Sakari; Pitkäaho, Tomi; Solvang, Wei Deng; Shu, Beibei; Arnarson, Halldor; Solvang, Bjørn. IEEE conference proceedings, 2022 IEEE/SICE International Symposium on System Integration (SII), 2022, pp. 766-771, doi: 10.1109/SII52469.2022.9708902.
6	Developing modules and demonstrations in Horizon 2020 TRINITY project

Acknowledgements

This thesis paper is submitted to UiT The Arctic University of Norway to meet the requirement of the degree of Philosophiae Doctor (PhD). I thank all the people who support me during the PhD study.

First and foremost I am extremely grateful to my supervisor, Professor Bjørn Solvang, for the invaluable advice, continuous support, and patience during my PhD study.

I would also like to thank Assicate Professor Gabor Sziebig for the discussions, professional suggestions and encouragement on my research. It was he, at the very beginning of my study, who oriented me in the correct direction.

I also appreciate Professor Arne Lakså for the help and advice during the PhD mathematical course geometric modeling. Even if the study process is very challenging, the outcome really impressed and inspired me a lot.

I would like to express gratitude to Professor Wei Deng Solvang for her treasured support which trained me and provided me with the opportunities to work in several international R&D projects.

I would like to thank Associate Professor Hao Yu who gave me so much help in both professional development and personal life.

My sincere thanks also to my friend Halldor Arnarson, who shared valuable ideas and developed many exciting systems with me. I feel happy to work with you and would like to continue work with you in the future.

Thanks to my friend Monan Yao and his mother, who guided me to come to study in Norway. I would never have had the idea of coming to study in this faraway country without your timely information.

I am deeply thankful to my parents and grandparents. The inculcation from you is constantly creating me to be a better person, and I am forever grateful.

My special and deepest gratitude to my wife, Ting Yang. You are my own jewel of inestimable value. Words cannot express how grateful I am. Thanks for the incredible supports and belief.

Table of Contents

List of Figures

List of Tables

List of Abbreviations

ANNs	Artificial Neural Networks
API	Application Programming Interface
AR	Augmented Reality
CNC	Computer Numerical Control
CNN	Convolutional Neural Network
CPS	Cyber-Physical System
GPS	Global Positioning System
GUI	Graphic User Interface
HRC	Human-Robot Collaboration
HTTP	The Hypertext Transfer Protocol
IIoT	Industrial Internet of Things
IoT	Internet of Things
MQTT	Message Queuing Telemetry Transport
NASA	National Aeronautics and Space Administration
OPC-UA	Open Platform Communications Unified Architecture
PID	Proportional–Integral–Derivative
RFID	Radio-Frequency Identification
ROS	Robot Operating System
SMEs	Small and Medium-sized Enterprises
SVMs	Support Vector Machine
TCP	Tool Center Point
VR	Virtual Reality
VUI	Voice User Interface
WCF	Windows Communication Foundation

/1

Introduction

With the rise of Industry 4.0, the technologies employed in the Industrial Internet of Things (IIoT) and Cyber-physical system (CPS) have been greatly improved in recent years. The IIoT and CPS systems enable Internet connectivity capability for industrial machines (such as industrial robots, conveyor systems and Computer Numerical Control (CNC) machines) and sensors (such as temperature sensors, humidity sensors and wind speed sensors). IIoT and CPS is not a single technology but is an aggregation of different technologies [1]. Typically, to build an IIoT and CPS enabled factory, various systems from different vendors will be needed. The increasing number of factory components or subsystems lead to higher factory complexity [2]. Increasing factory complexity also implies an increased level of difficulty in the operation and management of machines/equipment [3], and it is simply impossible to build and manage systems of this kind if they are not based on standards [4]. It is hard to develop standardization that can handle all the different industrial systems' connections and data exchange needs, but we can "break down" the overall industrial system into smaller components to develop possible standardization for each module. Such is demonstrated by STEP-NC [5] and ISO-14649 [6] with focus on the standardization of CNC machines. These standards attempt to achieve machine code transparency between different CNC machines and inter-machine communication. However, currently these types of standards are too few and too difficult to develop. In order to promote the development of standardization, there is a need to divide the industrial system into a hierarchy of layers and clarify the functionality of modules in each of the different layers, so an overall view of the whole system can be achieved.

The division of industrial systems into layers not only helps simplify industrial systems by breaking them into smaller, more manageable modules, but also offers greater flexibility. It provides for effective updates and improvements to individual modules without affecting other modules or having to redesign the entire industrial system. It allows vendors to compartmentalize their design efforts to fit a modular design, which eases implementation and simplifies troubleshooting. It standardizes industrial system modules to allow multiple vendor development and support.

The highly modularized system shows the potential of the system's ability for rapid innovation, which is emphasized in the industry 5.0 [7]. Industry 4.0 focuses on the development and progress of science and technology for the purpose of improving production efficiency and profits, further reducing the proportion of human workers in the production line [8]. But this ignores the expectations from the society, where its people should have meaningful and interesting jobs. For example, industry 4.0 technologies create a safe working environment for humans, but often requires operators with strong technical education and experiences. Those without relevant backgrounds will be left out. The concept of industry 5.0 is rooted in industry 4.0 but goes beyond industry 4.0. It is committed to enable ordinary workers to reap the benefits of technological progress. So far, industry 4.0 is still in the developing stage, so it will be more challenging to upgrade the industrial system according to the triple helix (human-centricit, sustainability and resilience) of industry 5.0. Further, by considering as many indicators of industry 5.0 as possible in the development of industry 4.0 technologies, this may smooth the upgrade of industrial systems. Therefore, the design of industry systems not only needs to consider the key indicators of industry 4.0 (such as efficiency, speed), but also needs to focus on human centric and social sustainability [9]. Human centric and social sustainability means that the popularization and development of technology should not require people to adapt, but require technology to adapt more people [10]. Ideally, workers should not need to master advanced technologies when using modern industrial production tools, and workers without a solid technological background should also be able to find jobs in modern factories.

For Industry 4.0 systems, improving the speed or efficiency of the system is not the only goal; improving resilience will be the new focus. The resilience of a system can be reflected in agility and the speed of responding to market changes. Agility was first introduced in 1991 by the Iacocca Institute of Lehigh University in a report [11, 12]. Thereafter, Rick Dove defined the agile as "an overall strategy focused on thriving in an unpredictable environment" [13]. Goldman et al. consider that agility is a comprehensive response to the business challenges of profiting from rapidly changing, continually fragmenting, global markets for high-quality, high-performance, custome-configured goods and services [14]. Agility emerged as the result from market changes such as increased product variability combined with decreased product lifecycles [15]. Further, the production systems switched from high-volume to small-batch production with a principle that aims to ensure efficiency while minimizing waste. As suggested by Sharifi and Zhang [16], agility comprises two major factors: 1, responding to changes (anticipated and unexpected) in due time; and 2, exploiting and taking advantage of changes as opportunities. In other words, agility indicates the ability of the next generation factory to renew itself in the rapidly changing environment. The highly resilient system can quickly re-plan the corresponding production line to meet the market demand and promptly occupy the new market when encountering business environment's changes such as new batch size or new product design, so the SMEs can convert the challenge into various opportunities [17]. Agile manufacturing enables this resiliency. The agility of a manufacturing system can also benefit from the increasing level of standardization and automation in the production process [18]. In addition, the higher agility system leads to faster reconfiguring and easier optimization of a factory setup. Faster reconfiguration is often associated with flexible manufacturing, which allows a system to be adjusted based on a pre-established setting, but we need agile manufacturing to enable the system to face unpredictable changes [19]. Agile manufacturing requires an advanced control system to offer this flexibility while keeping the concern of human centric [20], and inspired by the ideas from the H2020 project "Digital Technologies, Advanced Robotics and increased Cyber-security for Agile Production in Future European Manufacturing Ecosystems" — the TRINITY project, many newer technologies can contribute to this.

There are too many SMEs in the current manufacturing industry waiting to be rescued by newer technologies. Helping SMEs to march into the Industry 4.0 is the trend of the world. Besides the advantages offered by dividing industrial systems into layers, the SMEs also need a generalized architecture from the further analysis of the industrial systems.

In December 2019, I visited two SMEs from Wuhan (capital city of Hubei province in China). The city has a population of over 11 million and is the most populous city in Central China. The city is quite a big size and developed, so the SMEs can be good examples to show some general problems which other SMEs in the world are also facing.

SME 1, this SME is running a business for washing tableware, and the tableware is including: a dish, a bowl, a spoon, a glass cup and a pair of chopsticks. Each set of tableware is wrapped in plastic film to produce the final product, see Figure 1.1. The used tableware comes from different restaurants in the city by plastic box container and usually arrives with a lot of waste product, see Figure 1.2. They need to manually move the box from transport vehicle to conveyor belt. The tableware's wash time is based on its type and size, the bigger sizes tend to need more time. The shorter washing time uses less energy, increases the production flow efficiency, so the different types of tableware should be washed separately. Then the tableware should be sorted before the wash process, but the sorting machine can only separate the tableware into two groups: dish, bowl, cup as the larger items and spoon, chopsticks as the smaller items. The sorting machine is not efficient and always damages the tableware, see Figure 1.3. There is always need for human workers to identify and remove the broken tableware from the conveyor belt and collect the chopsticks and spoon into a box to wash them in a separate washing machine. The biggest challenge for this SME is that the dishes, bowls and cups must be sorted after the wash process so the product can be packaged. However, the same shape of plate or bowl can be printed with different restaurants' unique brands and these also must be sorted separately. The present solution is to use a lot of human workers, see Figure 1.4. But even for human workers, it is still hard to sort such a big quantity of items accurately and quickly, especially when doing this tedious task repeatedly for a whole day.

Figure 1.1: Final product at dishwash company

SME 2, the second SME is an automation equipment supplier which is also the supplier and the maintainer of the automation equipment in the previously mentioned dishwash company, so their products include: tableware sorting machine, tableware washer, tableware packing machine, etc. But this automation equipment company itself has no automation equipment during their products' manufacturing processes, all the products are handmade. The metal welding

Figure 1.2: Human workers are moving the plastic box for tableware

Figure 1.3: The sorting machine for tableware. Spoon and chopsticks as the smaller ones will fall to the lower conveyor belt from the gaps in the upper conveyor

Figure 1.4: Human workers are sorting the dishes, bowls and cups

and cutting tasks are done solely by human workers, see Figure 1.5, and the raw materials storage is less organized, see Figure 1.6. In their product design stage, there is no simulation or digital twin module to verify if the design will be successful or not, the product design depends only on the designer's experience.

These two SMEs really want to engage with some new technologies to improve their production efficiency, but they do not know what technologies that will be helpful and are afraid to invest without knowing that the investment will represent value for money. In general, there is a lack of information on how to combine relevant and efficient technology with a corresponding industrial task. I believe these difficulties are very common for most SMEs around the world. Many of them are counting on academics and developers to develop a system with new technologies for them. However, there are too many SMEs and different types of business. Therefore, to make an approach of developing a complete system that covers all the possible industrial tasks is impossible and unnecessary. But to identify the key drawback technologies over the different industry sectors is helpful.

An accurate automatic robot sorting system can release at least 6 human workers from the tedious task in SME 1, an automatic robot metal feeding plus cutting system can avoid the need for human workers to be in the dangerous and unhealthy environment in SME 2, and both the SMEs production efficiency

Figure 1.5: Metal welding and cutting are manually work

Figure 1.6: Raw materials storage

can be hugely improved by implementing advanced robot systems in their manufacturing processes. Taking the above mentioned two SMEs as examples, object recognition and object localization are the key technologies that can be extremely helpful to their businesses.

As long as the key technologies are developed as modules, the same modules can be used for multiple industrial tasks, which can benefit multiple SMEs. This method will be more efficient and cost effective when compared to developing one specific complete system target to one industrial task.

The different system developers can focus on the key technologies to decrease the development time, and the system integrators or SMEs can combine the different technologies, in the same manner, to form the different systems according to their particular needs.

To offer this flexible solution, the generalized architecture for IIoT systems is needed. Researchers have already proposed different IIoT system architectures comprised of three, four or five layers. A popular IIoT architecture consisting of the three layers presented below was discussed in [21]:

1. Perception/Sensor layer, collecting information on the physical object using different types of sensors.
2. Middleware layer, receives the information from the first layer and transmits it to the high layer.
3. Application layer, which carries out application specific functionalities.

A similar four-layer IIoT architecture was introduced in [22, 23], see Figure 1.7:

1. Sensor/Actuators layer, receiving/sending the data from/to different devices and machines (primarily analog data).
2. Internet gateways and Data acquisition system layer, analog/digital conversion with data aggregation
3. Edge IT layer, analysis and pre-processing the data from the second layer.
4. Data center/Cloud layer, data archive and management.

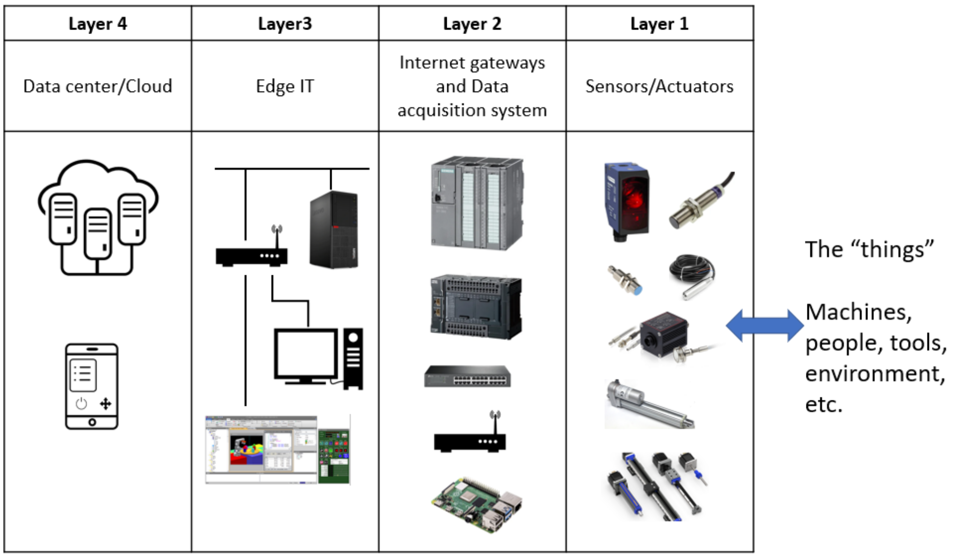


Figure 1.7: Four layers IIoT architecture

In comparison to the architectures presented above, the Middleware layer is divided into two layers, where the first layers are basically the same, but more data processing tasks are carried out in the lower layer to avoid redundant data transmission into the highest layer. This benefits the low latency system.

Much research has been undertaken, for example [24, 25] with a focus on how analytics could be efficiently performed for the data generated by CPS and IIoT systems. Results of this research proposed a three-layer architecture:

1. Edge layer, the data acquisition from embedded systems, operator inputs and machine tools.
2. Fog layer, mainly responsible for the local communication, cloud communication, machine learning and stream analytics.
3. Cloud layer, mainly responsible for the aggregation of the data from Edge and Fog layers and federated learning.

The hierarchical architecture of the automation pyramid is very common in production systems [26]. However, by dividing or merging different layers, the versions of the automation pyramid are different in various research. A five-layer Automation Pyramid concept in the framework of ISA-95 delivered by International Society of Automation is shown in Figure 1.8 [27].

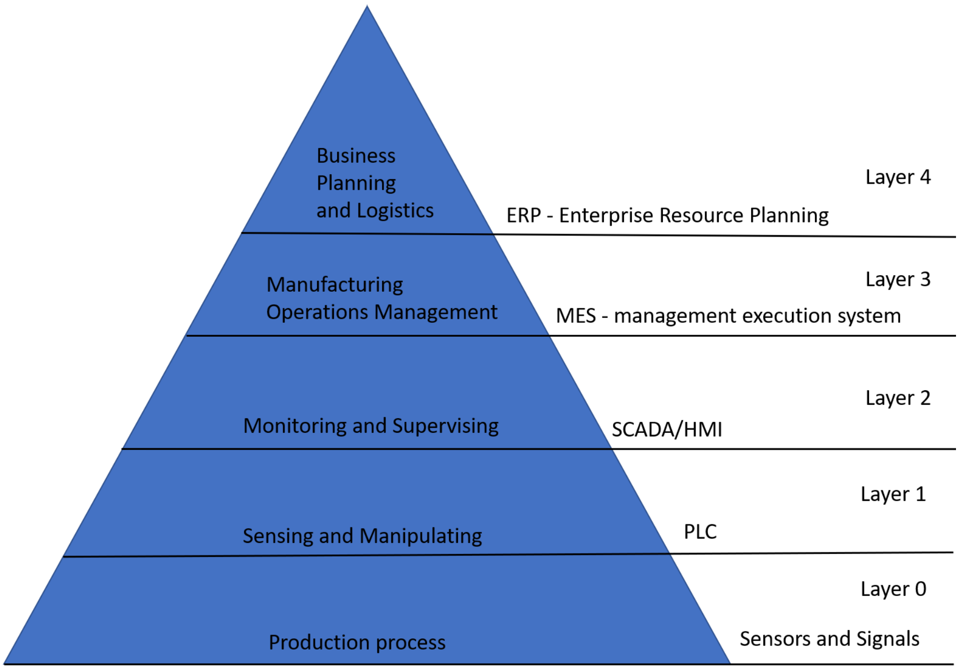


Figure 1.8: Automation Pyramid according to ISA-95

Many studies have proposed different industrial system architectures, however, there have been very few studies that focus on connecting the layers concept of IIoT architecture with the specific industrial tasks and pointing out some novel industrial 4.0 technologies within each of the layers. For many of the SMEs, the layers concept of IIoT architecture still stays at the academic level, which slows down the process of the novel industrial 4.0 technologies being applied into SMEs factories. The actual industrial task with intuitive figures should be developed to illustrate the layers concept, so the SMEs can better understand the needs in their business development and utilize more industrial 4.0 technologies efficiently. Since the IIoT layer architecture creates flexibility and scalability, if a clear definition of an industrial task with an IIoT layer architecture is established, then the same scalable and flexible architecture can readily be applied to any industrial task.

1.1 Roles in industrial systems

Due to the highly complex nature of our industry, it is not easy to directly analyse a task from current industrial practice. Before the first industrial revolution, when only human workers were used in manufacturing processes, this type of analysis is much more intuitive. Generally, the roles in these simplified manufacturing processes can be viewed as three elements: manager, worker and product. In this model, the manager would give the worker a highly abstracted task, such as 'make a table' or 'make a chair', and the worker would comply by carrying out the task.

With the introduction of modern machines, many of the human workers are replaced by high-speed, high-accuracy machines that never become tired — the worker's primary role is replaced by the machine. The highly abstracted task cannot be directly executed by the machine, so there is a need to have a person with a strong technical background, a production planner, who can understand the manager's task and divide it into detailed executable sub-tasks suitable for machines. There are many different types of machines from different vendors, and each machine has a specific machine language. As it is very hard for a person to understand all the different types of machine languages, it is common for a factory to have several specially-trained operators for the different types of machines. In Figure 1.9, the new roles of production planner and operator are added between manager and machine to perform the task translation and machine operation.

The utilization of industrial machines can increase product quality and productivity when the products have limited variety. The greater competition that results from the global economy requires that manufacturing enterprise must not only produce low-priced products of high quality but with an additional need to satisfy customer's personal preferences [28], which means the product variety needs to be very high. Factories can produce highly varied products in two ways:

- Build a different production line for each of the products.
- Reconfigure the current production line to produce the new product.

Obviously, the first option carries a high price in initial investment and the resultant production flexibility is very low. This is not a realistic option for most SMEs. The second method has minimal initial investment, but requires the manufacturing facility to have the ability to achieve a fast changeover [29]. With the increase of product variety, the production planner and operator's workload in the reconfiguration stage will be dramatically increased.

Machines changed industry as a whole and reduced the use of human workers in the manufacturing process. However machines have not fully replace human workers [30] (production planners and operators). By replacing manpower with automation, the factory manufacturing processes can easily make a big productivity leap [31], since machine-involved production processes can directly benefit from technology improvements. There is, therefore, always a need to

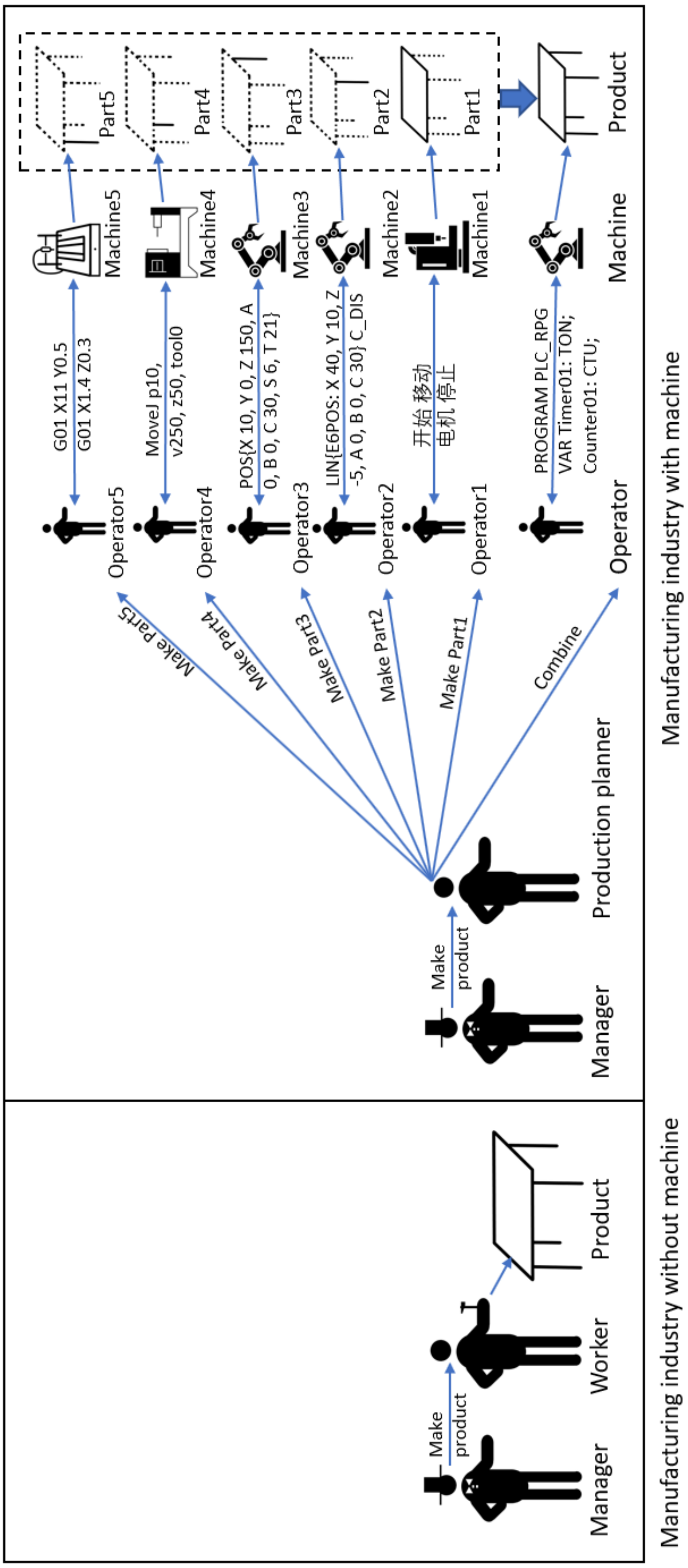


Figure 1.9: Comparison between the industry with or without machine

deepen the mechanization transformation in manufacturing processes. In a manufacturing system, the roles of production planner and operator should be improved but the positions are indispensable.

Many researchers are working on replacing the human element with automated device for the role of operator. A middleware system which acts as a gateway between field devices and decision support systems is introduced in [32]. The system enabled the unified communication between different industrial devices. In [33], the authors present a system concept that attempts to incorporate functionality from multiple machines. The system allows the operator to reconfigure the setup without specialized engineering background knowledge and increased the interoperability between different devices. In [34], the authors present a concept of a middleware-layer in IIoT systems to discuss different industrial communication protocols' interoperability.

All these presented research studies aim to contribute to the operation and communication between different devices by the adoption of IIoT middleware, and it is very common to find middlewares in the current IIoT systems [35]. In the current advanced IIoT manufacturing system, the middleware should be regarded as the replacement for human operators. In Figure 1.10, the roles in a manufacturing system are: manager, production planner, middleware, machine and product.

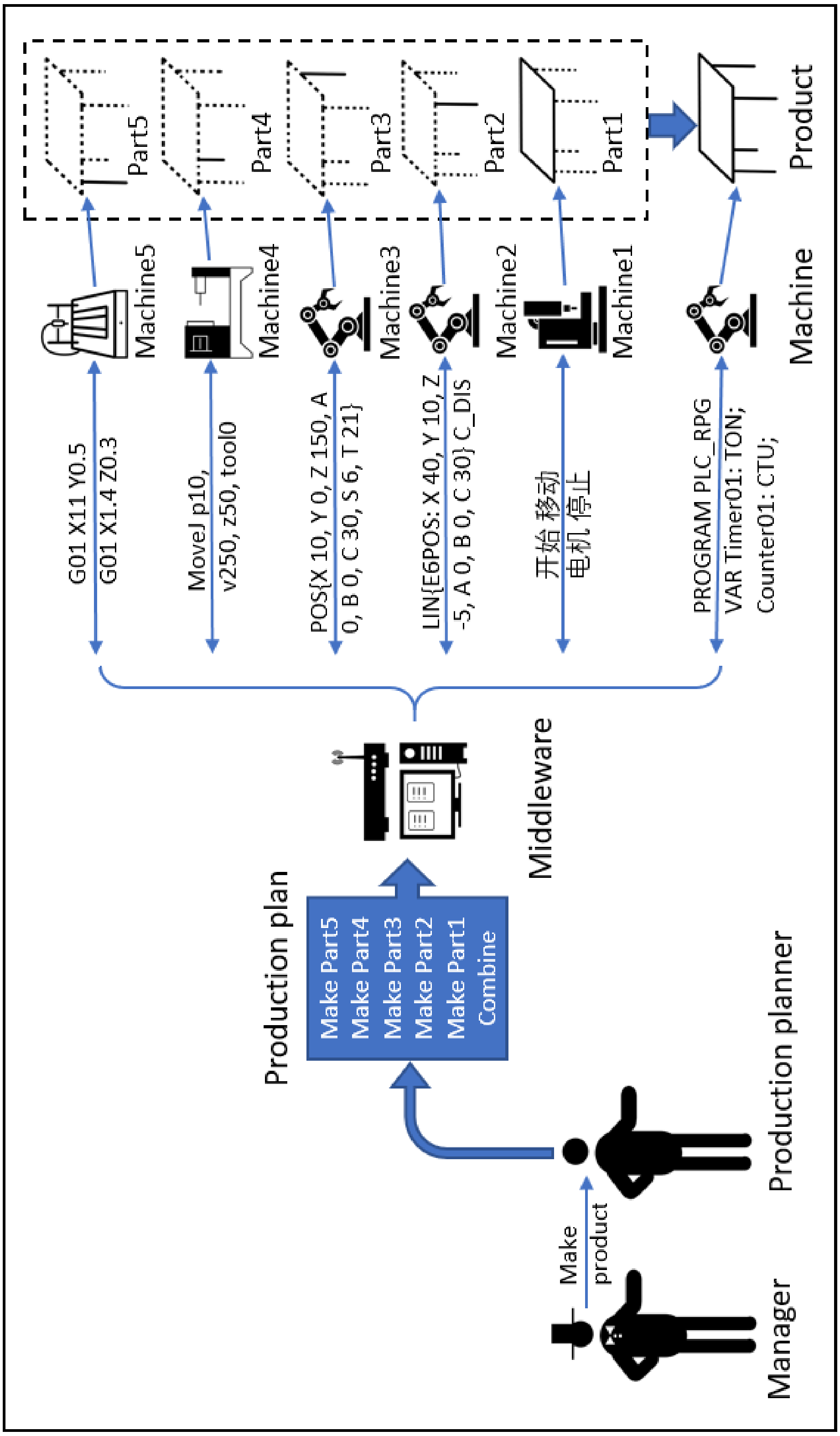


Figure 1.10: Manufacturing industry with IIoT machine

1.2 Matching layers and roles in industrial systems

As the roles in an industry system have been defined, we can then match the IIoT layers with the roles, see Figure 1.11.

The machine, in the first layer, has direct interaction with the physical product. The first layer contains sensors, actuators and network function enabled machines (a machine can be regarded as a combination of sensors and actuators). In this layer, all the data from the physical world should be accurately readable and all the physical machine motions are remotely controllable from the network.

The middleware is in the second layer, which collects data from the first layer and distributes control signals to each machine or actuator. The second layer contains a translator between the machines to enable communication with different machine languages and a data center to store the machine data. A centralized portal should be offered in this layer, so that inter-machines communications, without barriers, can be achieved.

The production planner is in the third layer, which can generate a sequence of machine instructions and supervise the execution of each machine. The third layer should contain the machine motion plans to coordinate the whole factory and manufacture of the different products, the machine motion should also dynamically change according to the environmental parameters.

As presented previously, most research studies do not define the higher layers for the IIoT systems. However, the fourth layer, for the role of manager, has not yet been clearly defined. In this thesis, the fourth layer for the indispensable role of manager is defined.

The fourth layer should contain a highly-abstracted task list which represents the production plans of different products — even a person with no manufacturing background should be able to understand the list. The human manager is the end-user of the industry manufacturing system. This individual can remotely and wirelessly choose the task with a simple interface. A single command from the manager can reconfigure the whole factory to produce a new product.

More details are presented about possible solutions and improvements in each layer in the following sections/chapters.

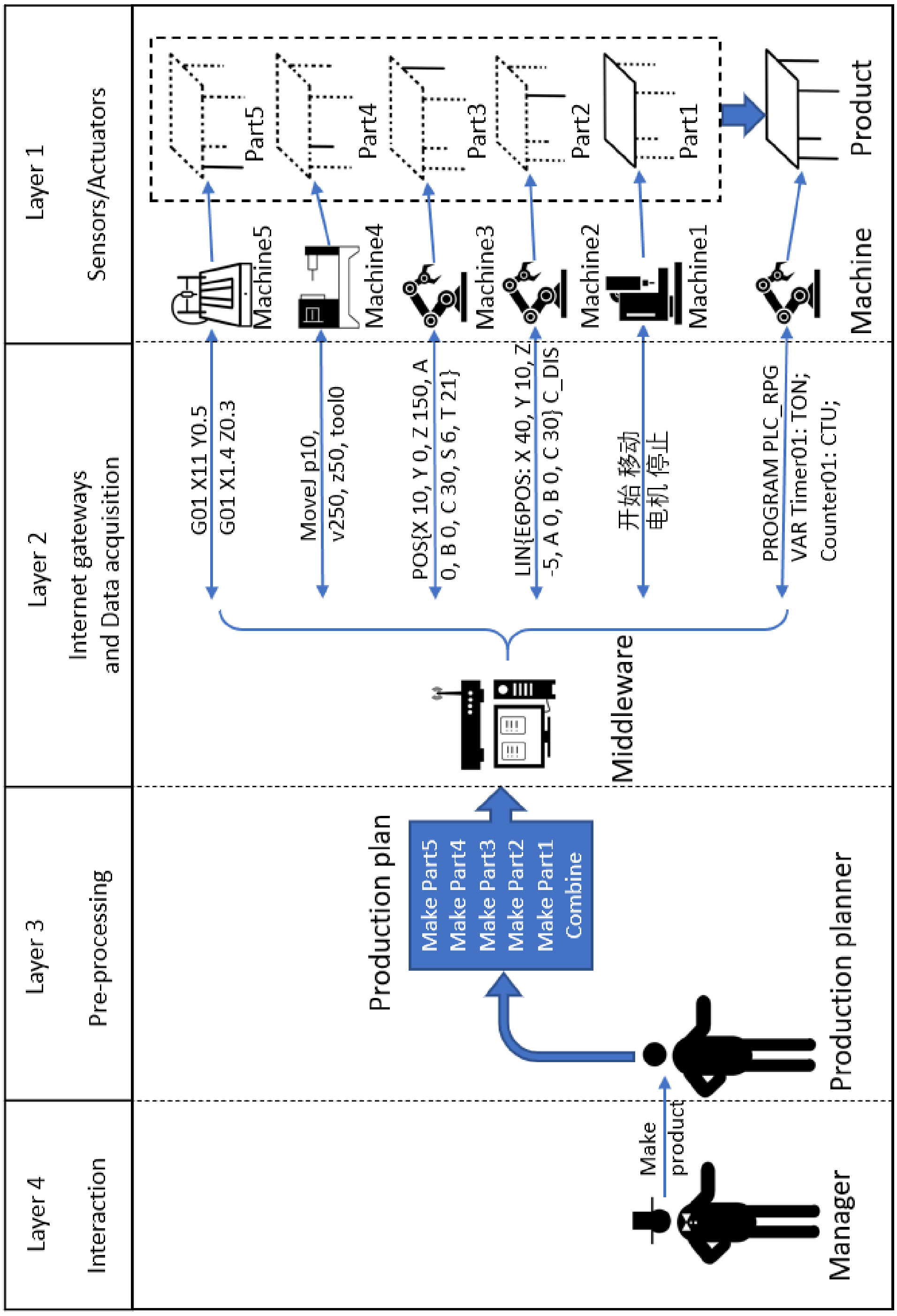


Figure 1.11: Layers of Industrial system

1.2.1 Layer 1 — Sensors/Actuators

In the first layer, we mainly focus on sensors and actuators, which handle the reading and control, respectively, of the object state in the physical world through the network. Most of the actuators can be controlled through the network directly or by an add-on device (such as Raspberry Pi). Current sensor technology has developed various sensors/devices for reading important environmental data like the object's colour [36], magnetism [37], temperature [38] and humidity [39]. But the classification and localization of object type and position (which can be used for improving the autonomy of robots) are still under development.

Object classification

The methods of object classification can be divided into two groups: indirect and direct. For the indirect method, some additional information should be added to the object (object pre-processing) such as colour painting, bar-code, QR-code, magnetic stripe, etc. See example in [40], where a traditional sorting system is using indirect object classification. The indirect object classification method is a mature technology. However, the extra information needs extra production steps which will increase the total processing time and manufacturing cost during production.

The direct classification will be more efficient since object pre-processing is avoided and the method is based on the object's characteristics. There are many object characteristics that can be used for direct classification such as weight, temperature, hardness, magnetism, etc. But these characteristics cannot broadly be applied to different types of object classification. Frequently, the objects of one class can be distinguished from the other objects by their shape which can include color and geometric outline [41]. Under these circumstances, computer vision-based object classification can be a good method, utilizing object shape to distinguish one object from another. There are many researchers working in this object classification field. [42] is improving 3D object classification efficiency, [43, 44] are implementing computer vision-based object classification into autonomous guiding of intelligent vehicles, [45] is developing the 2D object classification method for workpiece recognition.

It is hard to find an industry-ready direct object classification system, but there is an indirect (bar-code scanner) and direct (3D camera) object classification combination solution to detect object type and boundary size, see example in [46], sorting system using both indirect and direct object classification. In this system, the 3D camera can only detect the object's length, width and height, and object pre-processing is still required. More research is needed to develop an easy-to-use industrial system that can achieve direct object classification [47].

Object localization

Methods of object localization can also be divided into the two groups, indirect and direct. For the indirect method, objects also need extra pre-processing such as putting the object at a specific position where the system can easily acquire the needed parameters or attaching reference tags on the object like RFID tag [48] or GPS module. As with indirect object classification, indirect object localization is a mature technology but requires extra process time and exhibits hardware dependencies.

For direct object localization, there are many researchers working on different solutions: [49] introduced an active tactile perception method to acquire the object's localization by force-controlled robot, [50] introduced an object localization method by using electric sensing on underwater robots, [51] introduced an object localization method by joint audio-video data, etc. However, vision/image-based methods have become one of the main industrial object localization methods, based on the rapid and significant development of machine learning methods (artificial neural networks (ANNs) and support vector machine (SVMs)) [52, 53].

In [54], a three-dimensional object measurement system with multi-camera was introduced; in [55], a 3D localization system for automatic robot bin-picking using a stereo camera was developed; in [56], the stereo camera with cloud computing is combined to develop a workpiece localization system for CNC machines; in [57], a depth camera was applied for 3D point cloud data generation in the object localization system.

However, these object localization systems are highly dependent on their tasks and hardware, the authors only present their theory. There is no product that can be directly used for general object localization. Except for buying commercial services to develop a customized system, the only current possible solution for SMEs to own a direct vision-based object localization system will be through using open-source software packages.

Not many suitable software packages can be found online. A software called find_object_2d [58] was tested and the experiment was briefly presented in **paper 4**. In general, system and technologies should be adapted to humans, not the other way around. Thus in **paper 4**, a general architecture for transforming the traditional industrial robot system into an industrial 5.0 standard flexible manufacturing system has been offered. This architecture shows a human-centric approach towards building the future agile manufacturing system. Several new (and open) technologies are gathered and initial experiments are discussed in order to show their potential and examplify the transfer into more human friendly systems. A short video for testing find_object_2d can be found in this link: https://youtu.be/rJdpKDune74. In the video, the object position and rotation data are calculated and displayed in the terminal window, but the detection process is unstable. The rectangle frames around the identified objects are constantly changing, and the objects' small angular adjustments will lead to identification failure. The experiment concluded that a source code modification is required in order to use find_object_2d software in industrial object localization applications. Finding an engineer with sufficient skills to fulfil this task will be a difficult task for an SME.

Object classification and object localization can benefit from, and contribute to, each other's successes [59]. Another software, ros_object_analytics [60], which can acquire object classification and position data together was also tested during this PhD study. The experiment is briefly presented in **paper** 4 and a short video can be found in this link: https://youtu.be/xJcUMexQhKw. The software can detect an object both in 2D and 3D views. In the 2D view, a fast and reasonably accurate rectangular frame with object class name and 2D position information is shown in the window. However, in the 3D view, the bounding-box around the detected object is not accurate enough, and object rotation information cannot be offered by this software. This software is only suitable for a limited situation of object localization (the object rotation parameter can be ignored), so more investigation of object localization for general situations will be needed.

For the commercial solutions of object classification and localization, there are several options in the industrial 3D camera detection market, e.g., Pickit3d [61], ISRA VISION [62], SICK [63], Convergent-it [64], Zivid [65] and Photoneo [66]. The camera hardware and software sometimes can be sold separately, but the software can typically support a specific camera hardware. Even if an SME bought both hardware and software from one company, they still need time, competence and resources to familiarize and configure the camera system with their current production equipment and setup. Some of the commercial companies may as well offer system integration services which helps SMEs to adopt the camera system faster. This reduces the technical barrier for the SMEs but increases the initial investments.

Direct object classification and localization can significantly improve the autonomy of the industrial system. In order to evaluate the state of the current technologies, two open-source software packages are tested in **paper** 4. The result from the up-mentioned tests indicates the open-source solutions of direct object classification and localization technologies are not fully mature, and the commercialized industrial-ready solutions can be too expensive to acquire. Therefore, the research focus on direct object classification and localization is recommended. If the open-source solutions have a more mature state, there will be a promotion and even attract more customers to commercial products as the technology gets widely spread through an active community. The perfect classification and localization solution cannot be found during this PhD study, but a proposal for the development focus in this layer is stated herein.

1.2.2 Layer 2 — Internet gateways and Data acquisition

In the second layer, we mainly focus on methods that gather information from the lower layer, and distribution of instructions to the lower layer. With the increasing quantity and processing power of industrial embedded devices, more sensors, actuators, and programmable machines are connected to the network. Equipment diversity or heterogeneity requires a common platform for the communication [67]. There are several protocols that have been developed for the communication between different devices, but two newer protocols namely Message Queuing Telemetry Transport (MQTT [68]) and Open Platform Communications Unified Architecture (OPC-UA [69]) are particularly focused on the rise of Industry 4.0 and IoT.

In [70], the authors use OPC-UA as the data acquisition platform for STEP-NC enabled equipment; in [71], the authors use OPC-UA in web-platform human machine interfaces; in [72], the authors use MQTT with Raspberry Pi for human-robot collaboration system; in [73], the authors use MQTT with Raspberry Pi for remote controlling of KUKA robot in intelligent oil fields.

Both of these two communication protocols are broadly used in industrial systems, but the direct comparison of OPC-UA and MQTT would not produce a balanced result since it could be implemented and individually optimized in different ways [74]. The choice of the communication protocol is largely dependant on the project infrastructure - some old equipment can only naturally support either OPC-UA or MQTT. The protocols have very high flexibility, such as the project presented in [75]. Here, the authors designed a system where the OPC-UA supported devices and MQTT supported devices can share information. So, there is no problem even if the infrastructure is not originally supporting the protocol.

Nowadays' automation systems mostly connect sensors and actuators with local networks. The low latency industrial data sharing is restricted by geographical distance, which limits the possibilities of information aggregation [76]. However, cloud computing technology can connect the IIoT systems beyond any limitation of physical distance and enable a higher level of information aggregation [77].

In the future, the industrial communication platform should be combined with cloud technology [78]. A cloud-based prototype system is presented in [79], in which, multiple devices and online services can share information through the Internet. A comprehensive network evaluation study has been conducted in [80], in which, three of the most commonly used cloud providers, i.e. Amazon Web Services, Google Cloud Platform and Microsoft Azure are analyzed. Their experimental results show Microsoft Azure in the WestEU data center has the lowest latency for their experiment location and OPC-UA comparing the other two communication protocols (HTTP and WCF) is just slightly slower, the mean latency for OPC-UA is 63.49 ms. A conclusion is summarized from the testing: processes that require latency/control period lower than 40 ms cannot be controlled through cloud services.

The cloud service latency can be affected by various factors, i.e. distance between data center and end user, server computer performance, network speed. During this PhD study, a cloud service experiment was also conducted to verify the possibility of combining industrial communication platform with cloud technology. The experiment was conducted with the collaboration of Centria University in Kokkola, Finland. Microsoft Azure was chosen as the service provider, and the data center "North Europe" that is located in Ireland was chosen. The experiment methodology is:

- Set up a virtual machine in the cloud.
- Run the OPC-UA server on the virtual machine.
- Run a python script on a local machine in Narvik, Norway, to measure the time it takes to read and write a variable in the OPC-UA server.
- Run the same python script on a local machine in Kokkola, Finland, to measure the read and write speed.

The experimental setup is shown in Figure 1.12 and Table 1.1, and the average speed in shown in Table 1.2.

Based on the promising result from the experiment shown in Figure 1.12, a further experiment was carried out and described in **paper 5**. A platform independent graphic user interface was set up on the Azure cloud server, and users from different geographical locations (Finland and Norway) can control the same physical robot in Norway by any browser-supported devices, e.g., smartphones, tablets, and PCs. The OPC-UA is used as the main industrial communication platform for the experiment. The result shows that the robot operator can remotely manipulate the industrial robot and even create and execute a robot program with a few TCP points. However, a delay was observed

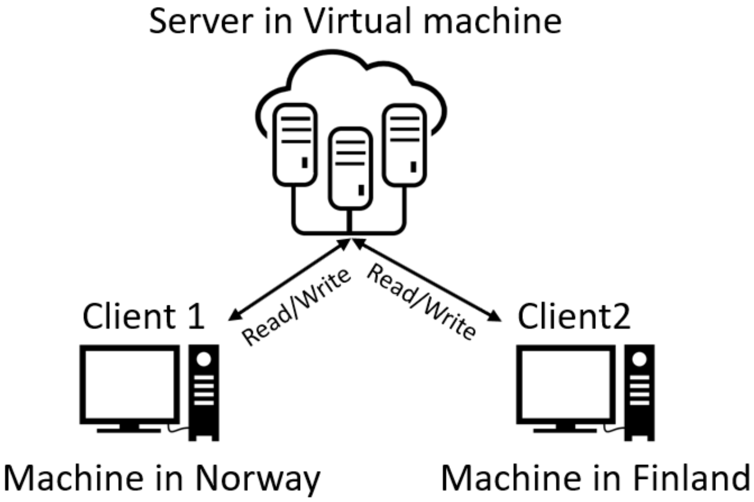


Figure 1.12: Test OPC-UA with cloud service.

Table 1.1: Experiment setup.

Azure data center	North Europe
Experiment location	Narvik, Norway, and Kokkola, Finland
Azure Virtual machine type	D2s_v3
Virtual machine vCPUs	2
Virtual machine RAM(GiB)	8
Virtual machine operating system	Ubuntu 18.04-LTS
OPC-UA client and server version	Python OPC-UA [81]
Tool for reading server values	UaExpert [82]

Table 1.2: Average communication speed.

Location	Read	Write
Norway:	61.74 ms	62.10 ms
Finland:	59.69 ms	59.48 ms

during the experiment. It can be caused by the limited performance of the cloud server and may be fixed by upgrading the cloud server computation power.

The general industrial communication platform is needed when multiple industrial devices have to share some common data. To find and evaluate the proper communication protocol, OPC-UA and MQTT are studied, and two experiments about hosting OPC-UA on Cloud are conducted as described in Figure 1.12 and **paper 5**. The result from the Table 1.2 shows that the OPC-UA server, regarded as an industrial communication platform, can be set up in the cloud service, but the setting in the experiment cannot be used for low latency services. More investigation with better performance cloud virtual machines will be needed. The possible improvement in this layer can be through developing a low latency cloud-based industrial communication platform.

1.2.3 Layer 3 — Pre-processing

As we described earlier, in the third layer, a strong-technical-background production planner will present the factory's working state to the manager and divide the manager's task to coordinate each machine. However, with the maturity of technologies and strong concern in productivity improvement and digital transformation, the role of production planner can be replaced by a digitized system, see Figure 1.13. In the system, we consider the second layer has present a common platform that all machines' parameters can be read and modified, and all the necessary environmental data can be accurately acquired in the first layer. Then, to enable this replacement, two main topics — digital twin and robot task planning will be the focus in this layer.

Digital twin

The digital twin concept was first mentioned by NASA in a space project monitoring a satellite's behaviour and simulating the possible changes in the setting [83]. The concept is recently defined as a set of virtual information constructs that fully describe a potential or actual physical manufactured product [84]. Digital twin has advanced rapidly in various industries [85], i.e. health, medical, food, and manufacturing industry. In the manufacturing industry, the establishment of digital twin not only benefits system monitoring and simulation, presenting a better view of the overall system, but also design optimization and system improvement [86]. Great interest in digital twin has been observed in the manufacturing industry, but currently, most research is available in the form of conceptual models and simulations and the application to real-world scenarios is limited [87].

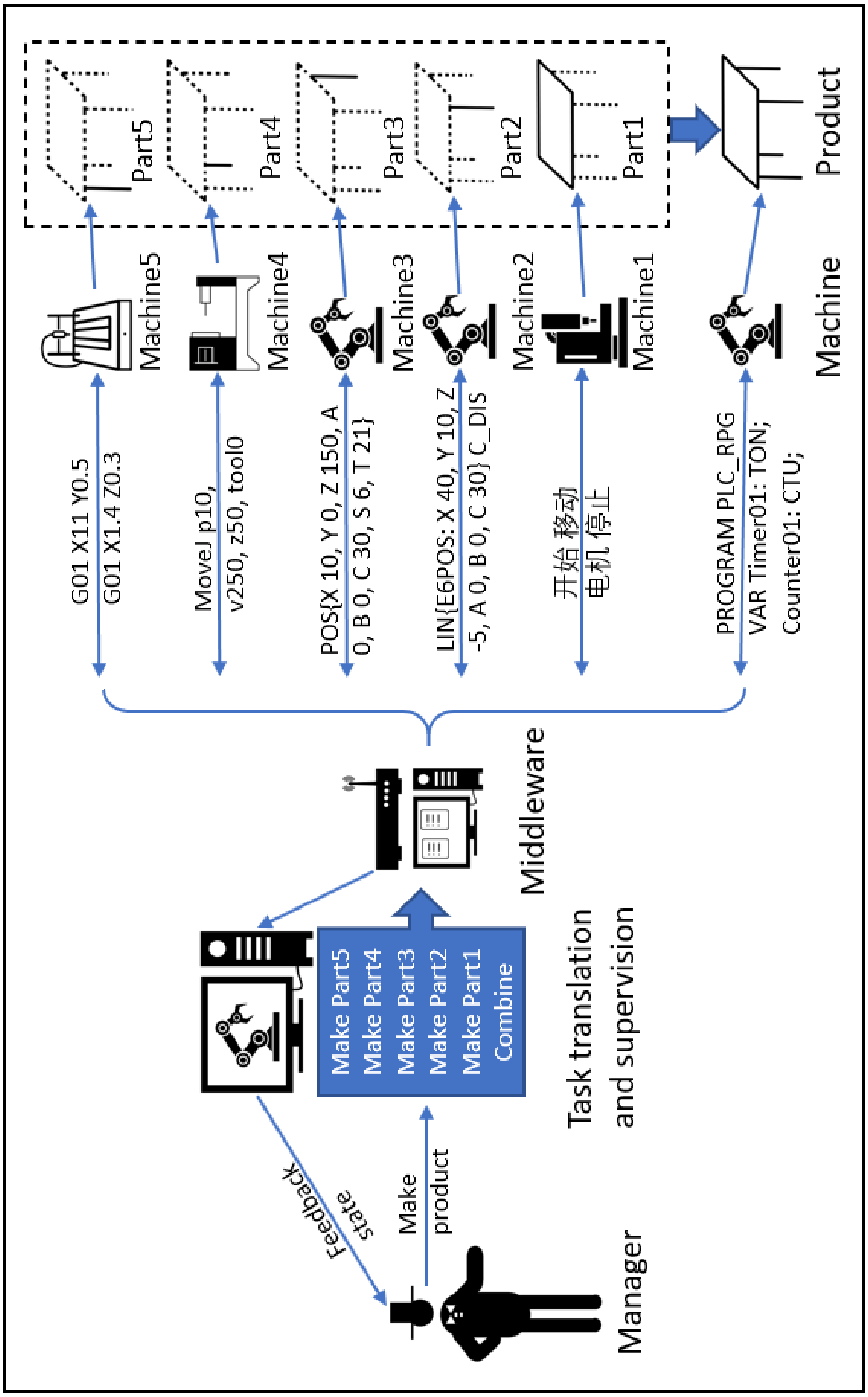


Figure 1.13: Task-dependent manufacturing

In order to promote the development of digital twin in the manufacturing industry, a few studies have been conducted in this PhD study. A real digital twin application is introduced in **paper 1**, in which a press-break line in a Korean factory with its virtual module in Gazebo [88] simulation was presented. The operator can supervise the execution of the physical production line in a virtual 3D space and configure the real robot system remotely through a graphic user interface. The results presented in **paper 1** are from a research project, where the authors took part in various research and development tasks. The robotic system was an already existing installation in a factory in Korea, and the task was to establish a cyber-physical system with it. The authors contribution is the establishment of this, mainly methodological and software development. In **paper 1**, the robot controller is connected with Robot Operating System (ROS) by personalized configuration. So, the robot controller can receive the command by ROS and then drive the robot to move. However, all the virtual models in this digital twin system are specially designed and strongly dependent on the physical system making the system migration or re-design in other types of setup is difficult.

A more general-use digital twin simulation environment and easier implementation for most of the robot systems should be proposed. Besides the open-source simulation software, i.e., Gazebo, Webots [89], most of the industrial device suppliers have their commercial simulation software, i.e., SIEMENS Plant Simulation [90], ABB RobotStudio [91], KUKA.Sim [92]. The supplier-produced simulation software has good offline programming and post-processing performance, but they typically just support their own industrial devices. Since many SMEs production lines have different industrial devices from various brands, a general simulation software can be preferred. There are several popular generic commercial simulation software available, i.e., Visual Components [93], RoboDK [94]. These software have a well-designed multi-manufacturer robot library so that the simulation design of a whole factory can be achievable.

Since the digital twin has almost the same performance and specification as the physical twin, the industrial robot system's offline programming and validation can significantly benefit from it. In more specific cases, for the robot painting/scanning tasks, the operator can test the robot program in the digital twin to check the results, e.g., time-consuming, painting/scanning coverage, and again to optimize the robot program without interrupting the production of the physical robot. For any robot tasks that require higher accuracy, e.g., robot welding, pick-and-place, the operator can also verify the robot program in the digital twin to find the acceptable moving paths and tolerance parameters. However, the accuracy of a virtual model compared with its physical representation cannot always be matched very well. Therefore, performing high accuracy robot program validation would require extra work on robot calibration and measurement. Such automatic calibration is an important research focus to resolve the major limitation of the digital twin.

Some traditional digital twin use cases are quite similar to offline programming. Operators program and verify robot motion in the virtual environment, output the robot motion code by a post-processor, and then download it to the physical robot. By combining the data acquisition platform in **Layer 2**, the digital twin can directly affect the physical robot while it is running – this results in a new type of digital twin use case, a two-way digital twin. The two-way digital twin allows the data flow to be transferred from physical to digital and digital to physical. Changing the digital robot position will directly change the physical robot and vice versa. So, the digital twin can be based on the current physical robot state to forecast the future situation and dynamically adjust the present robot motion achieving the dynamic optimization.

The human-robot collaboration (HRC) system can be regarded as a complex system, and a digital twin can benefit the HRC system in the whole lifecycle [87]. An HRC task is introduced in **paper 2**, in which a digital twin is used for task design and verification. In this experiment, Visual Components is used for the simulation environment, so that the virtual model of the physical robot can be directly imported from the software's own library. The complete test for the experiment has not been conducted, but it is partially tested in real life. As I described in the **1.2.1 Layer 1**, the direct object localization method is not ready, the indirect object localization is tested in the **2.2.3 Module 2** for tracing the human hand position. I plan to conduct the real-life experiment when the object localization solution has better accuracy and add extra safety functions to the robot (or directly use the Cobot with all safety functions embedded). However, with the software's open python API, almost all robot settings can be controlled in the virtual environment. The possibility to connect the virtual robot with the physical robot is also verified in the experiment video: https://youtu.be/o-k9ClpobJc.

The interaction between physical twin and human is intuitive since we are in the same physical world, the human can use various methods of perception to acquire the state of the physical twin, i.e., vision, tactility, and olfaction, and various methods to control the physical twin, i.e., hand guiding, buttons signals, software programs. However, the digital twin is a set of information inside the simulation software, is in the virtual world, so the interaction between physical human and digital twin can be completely different. In **paper 3**, some possible input/output methods for the HRC system in digital twin were discussed, and a more flexible user interface will be discussed in the following paragraphs in **Layer 4**. For the scenarios discussed in **paper 3**, the robot system will be dangerous without enough safety measures. For example, if the safety system fails when the nut is dropped by the operator (this is a very common mistake for a human), the robot may continue to move the TCP down and start the screwing task. This will cause the operator to get injured. The system should always trace the nut position and compare with the human hand position, if these two positions are too far away, the system should recognize the error. This, again, require higher accuracy of the direct object localization method, which need to be solved. So, the complete real-life experiment cannot start.

Robot task planning

The robot task planning will be based on a list of predefined abstract tasks that contain a series of robot movements to "understand" the manager's request. For example, the simple request from the manager, "Pick one bowl deliver to dish" in Table 1.3, contains seven sub-tasks for the robot arm. However, some specific values in the sub-tasks, i.e., position (**bowl position in the container**, **dish position on the work table**, dish surface **area A**, **idle position**), velocity and objects (**bowl gripper**, **robot arm**), can be dynamically acquired from the sensory system in **Layer 1**. The sub-tasks are not restricted to the specific robot or object, and can be independent of the hardware system, which offered the opportunity for easy system migration and required fewer instructions from the system operator — the manager.

Table 1.3: Abstract task and sub-task.

Abstract task	Sub-tasks
Pick one bowl deliver to dish	Trigger **robot arm** with **bowl gripper** : 1. Move TCP to **bowl position in the container** 2. Close gripper 3. Lift up TCP 4. Move bowl to **dish position on the work table** 5. Move down bowl to the dish surface **area A** 6. Release gripper 7. Move TCP to **idle position**

Inverse kinematics is the key element to support the independence of robot movement, it makes the mathematical connection between the workpiece, the robot Tool Center Point (TCP), and the robot joint values, while inverse kinematics itself is strongly dependent on the hardware system. Newer versions of robot controllers have already embedded the interface for the inverse kinematics calculation, but many SMEs only have old and simple manipulator systems where such an interface is not offered. As described in **paper** 4, customized inverse kinematics solver can be developed as long as the manipulator can be controlled joint by joint. In which, software packages MoveIt [95] is based on the target robot geometric module in Unified Robotic Description Format (URDF) as the parameter input to generate the IKFast [96] solver file, so that the inverse kinematics calculations can use the solver file to calculate the robot joint position from the inputted TCP position. The IKFast is an analytical solver which supports maximum 7 joints robot arm. The tutorials about the installation steps are quite detailed, however, considering the large numbers of dependencies with working environments, the implementation is not easy. IKPy [97] is another possible option for the customized inverse kinematics solver. The software packages can easily be installed by a few lines of code. Then, with a simple setup of each joints' link positions and limitations, the solver can output the robot's target joint positions from the inputted TCP position. The IKPy demonstration code shows the supported robot arm joints can be up to 7. However, from my experience, the orientation calculation is not always correct, but it is very suitable in a situation where the orientation parameters are not needed. Sns-ik [98] is also one possible option. This numerical solver was published along with a few research papers claiming that it can support more than 6 joints. The input parameters allow extra hard constraints in the robot joint space beside the robot's basic joints' link positions and limitations. Moreover, a physical 7-joint KUKA robot with a virtual 50-joint planar snake robot are demonstrated in its description. However, the use of the software package is not described in detail, and the user needs to carefully study the research papers to use the solver efficiently. An overview of these three inverse kinematics packages is shown in Table 1.4.

Table 1.4: Analysis of inverse kinematics packages

Packages	Method	Supported Joints	Implementation
IKFast	analytical	max 7	hard, but with full tutorial
IKPy	numerical	7, unclear if support higher than 7	easy, as long as python supported
sns-ik	numerical	no limited, demonstrated with 50 joints	hard, need to study research paper

During this PhD study, a demonstration for combining the customized inverse kinematics with a digital twin has been developed, in which, IKFast is used as the inverse kinematics solver, Visual Components is used as the digital twin simulation environment, and OPC-UA is used as the robot information exchange platform. See the demonstration video in **Section 2.2 Module demonstration and description**, Table 2.1, Video number 1. In the video, the python script is in the lower right window, the software to monitor OPC-UA values is in the lower left window, the view to indicate inverse kinematics is in the top left window, and the digital twin is in the top right window. The description for the demonstration follows, and also see the logic structure in Figure 1.14.

- A public variable in server works as a trigger for the inverse kinematics.
- The python script is used for updating robot TCP values.
- The execution of the script will activate the trigger and update the TCP values in server.
- Inverse kinematics solver receives the trigger signal, read the TCP values from server, calculates the robot joint values, and updates joint values to server.
- When the robot reaches the target TCP, inverse kinematics stop the calculation, and then reset the trigger.
- The digital twin constantly reads the joint values from OPC-UA server and updates the virtual robot's joint values.

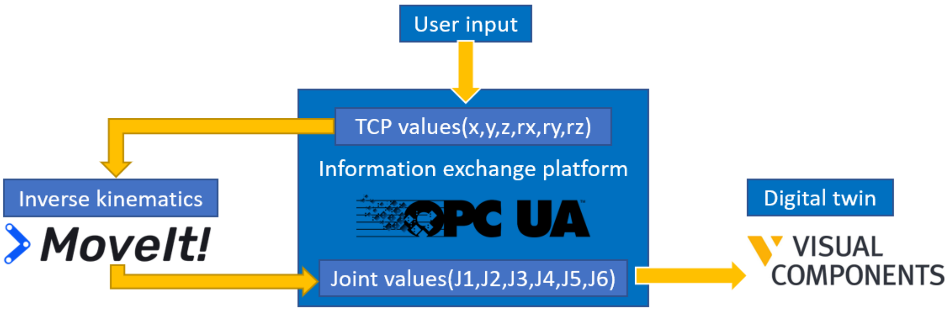


Figure 1.14: Logic structure of demonstration for combining inverse kinematics with a digital twin

The demonstration verified that the digital twin is a good simulation method in the early stage of robot system design, and the success of customized inverse kinematics offered the possibility for further integration in various physical robot systems.

In order to allow a digitized system to replace the production planner, digital twin and robot task planning are studied in this layer. A digital twin simulation environment is tested in **paper 2** and the interaction between digital twin and human is discussed in **paper 3**. The study also point out the key element of task planning is the customizable inverse kinematics solver, which enables the cross-platform of robot tasks. The demonstration described in Figure 1.14 shows that digital twin can be used for system state feedback, and robot task planning can be beneficial in controlling multiple robots. The possible improvement in this layer can be through conducting more tests on multiple robot simulations and physical platforms.

1.2.4 Layer 4 — Interaction

The fourth layer is the highest layer in the industrial system, and it is oriented to the end-user which is the manager role in our system. In this layer, a human-robot interface will be offered so the manager with no technical background can configure the whole production system with simple commands which associate with each sub-task from **Layer 3**.

The common way of interacting with an industrial robot is with a teaching pendant which can be regarded as an embedded Graphic User Interface (GUI). However, this traditional method is both difficult and time-consuming for someone not familiar with the robot controls, and the traditional industrial robot needs to have a cable connection with the teaching pendant which restricts the flexibility of the robot system.

There is, therefore, a need to propose flexible and intuitive human-robot interfaces. According to the communication typology, human-robot interfaces can be classified into four main categories [99, 100]:

- Visual displays (e.g., GUI, Augmented Reality (AR) interface [101], and Virtual Reality (VR) interface [102])
- Gestures (e.g., hand movements [103], and eye tracking [104])
- Speech and natural language (e.g., auditory speech [105])
- Physical and haptics interactions (e.g., manual lead through (by hand) programming [106, 107])

Among all these industrial robot controlling interfaces, gestures interfaces require the robot to be equipped with an extra camera detection system, AR/VR interface also requires the robot to be equipped with an additional headset for the end-user. But theoretically, both gestures and AR/VR interfaces can be applied to any industrial robot, as long as the robot can be controlled by an external signal source other than the teach pendant.

Physical and haptics interactions interfaces are currently broadly used in the collaborative robot system (Cobot). The embedded torque sensors in a Cobot system makes it possible to detect and react to forces during programming of the system. This is an important safety measure and Cobots can be programmed by the – manual lead through principle. Simply applying a force on any joint can move the robot, which is more intuitive than the traditional teach pendant programming method. Most collaborative robots in the current market offers this manual lead through programming (also referred to as programming by hand guiding), e.g., Universal robot CB3 family [108], KUKA LBR iiwa robot [109], Rethink Sawyer [110], YASKAWA HC10 [111], ABB IRB 14000 YuMi [112], Franka Emika [113]. However, in order to start the movements of the robot you first have to overcome the more or less unpredictable static friction of the motors. This will give a negative impact on smoothness and precision when programming of the robot. So, another type of manual lead through programming method is developed to achieve higher accuracy and adopted in some newer versions of collaborative robots, e.g., Universal robot e series [114], ABB SWIFTI CRB 1100 [115], KUKA LBR iisy robot[116]. Among these robots, one extra 6-axis force/torque sensor is mounted on or beside the robot TCP flange. So the operator can apply force on the robot end-effector, or the handle on the force/torque sensor, to start the robot movement. This results in higher accuracy since the force applied by the operator is directly captured by the force/torque sensor and does not need to overcome the motor's static friction. Although, there are industrial solutions like KUKA Force Torque Control [117] and research by adding extra handle with buttons or external 6-axis force/torque sensor (e.g., OnRobot's Hex force/torque sensor [118], FT 300-S force/torque Sensor [119]), to enable the general industrial robot with the functionality of manual lead through programming. Overall, the additional complex sensor system is mandatory to operate a general industrial robot with physical and haptics interactions interfaces.

Combining a simplified GUI with a speech user interface can create benefits both by using the traditional operator's existing habits and by minimizing the dependency on additional hardware (a simple microphone as the hardware add-on will be sufficient for a speech user interface). In this PhD study, therefore, GUI and speech user interface (or Voice User Interface (VUI)) are the main focus for this layer.

The GUI

The GUI should offer a flexible environment which can support multiple operation systems, and the communication protocols mentioned in **Layer 2**. In the market, the industrial oriented GUI packages can be categorized in two main groups: Application-based GUI like TeslaSCADA2 [120] and Web browser-based GUI like FlexiGUI4.0 [121] and node-red-contrib-opcua [122]. The Application-based GUI — App-GUI needs each client to install the runtime environment so that the GUI can be loaded. However, the modification of the GUI requires each of the clients to reload the application manually which can be inefficient when the client quantity is large. The Web browser-based GUI — Web-GUI allows each client to use only a web browser to open the GUI, and the modification of the GUI can automatically be synchronized to all the clients. A discussion regarding the comparison of the two GUI solutions is presented in **Paper 4**.

A demonstration by using Web-GUI to control industrial robot's Digital Twin has been developed during this PhD study. The GUI is shown in Figure 1.15 and is developed by using the open-source node-red-contrib-opcua package. The demonstration video can be found in **Section 2.2 Module demonstration and description**, Table 2.1, Video number 2. In this demonstration, the GUI is opened by a browser in PC and is controlling the robot Digital Twin through the OPC-UA server, the GUI buttons A1 - A6 are corresponding to the robot joint values J1 - J6 in degrees, and the robot joint values on the server can also be read in the GUI. A further experiment of the GUI is conducted in **paper 5**, in which the flexible GUI is loaded by operators in Finland and Norway at the same time on different platforms (e.g., PC, smartphone, tablet) to program an industrial robot. The demonstration and the experiment concluded that the GUI package node-red-contrib-opcua is a good tool in designing the industrial GUI and the Digital Twin again demonstrates its value in the robot system design.

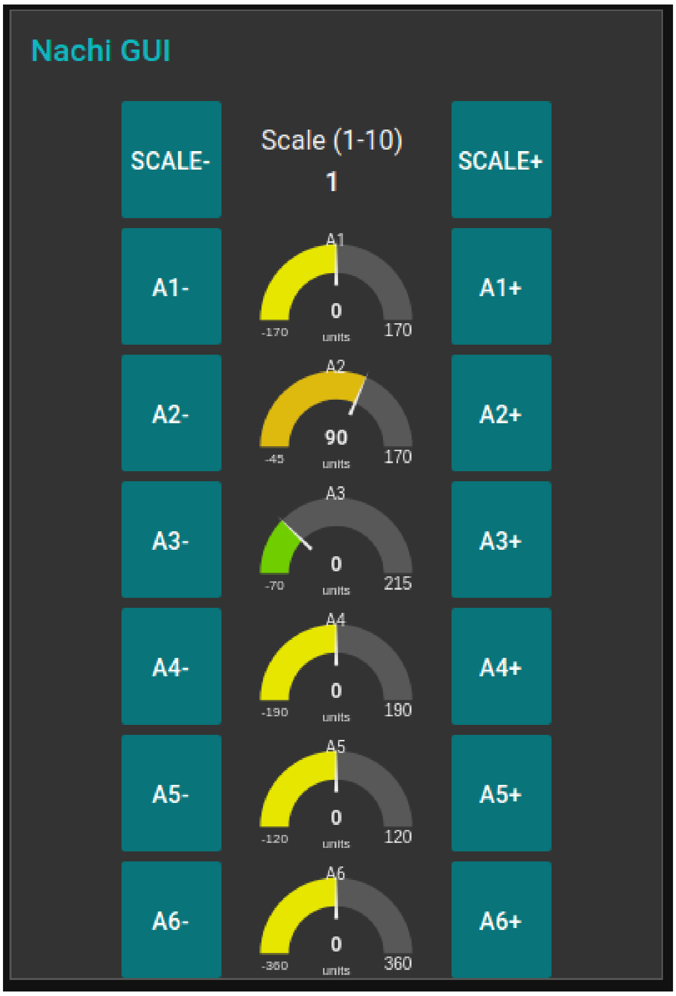


Figure 1.15: A robot joint control GUI

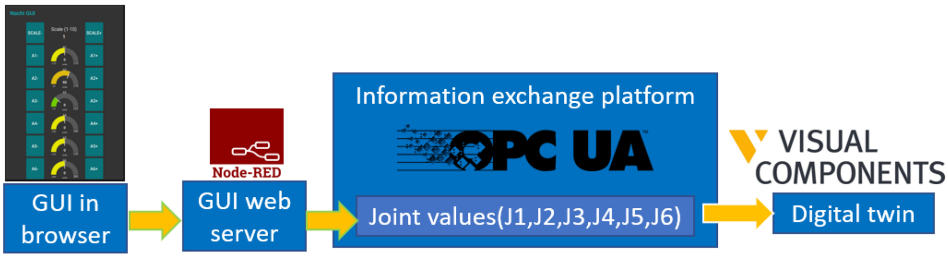


Figure 1.16: The data flow in the Web-GUI demonstration

The VUI

With the increasing numbers of scholarly articles and patents dealing with robot voice control that have been published, speech recognition has gradually become an important element in robot control and programming [123]. Many studies are using speech recognition methods to develop the robot VUI and a combination with other types of interaction for ensuring a higher system robustness.

In [124], the authors developed an interactive industrial robot interface by combining visual and voice command with an offline speech recognition library Microsoft Speech SDK [125]. In [126], the authors present an intuitive industrial robot programming method by combining human gestures and voice commands with offline speech recognition library PocketSphinx [127]. In [128], the authors designed a human-robot interaction method by combining hand gestures and voice command with online speech recognition API. In [123], the authors present a voice-based collaborative robot programming method which is using Web Speech API for speech recognition. Therefore, two user interfaces, GUI and VUI are proposed together in this PhD study, GUI and VUI are used interchangeably as the backup method for each other.

As can be seen in the previously mentioned research, there are two main speech recognition methods: offline library and online API. The main difference between these two methods is the audio data processing location — the offline library is processed locally while the online API is processed on the cloud. A benefit from the advancements of network technology is that the audio data transmission delay is decreased to an acceptable range. In addition, offline speech recognition method requires extra effort in the system implementation and higher computational capability for the hardware system. As a result, many researchers tend to use the online speech recognition method.

The online speech recognition API has various suppliers, e.g., Microsoft Azure Speech Service [129], Google Speech-to-Text [130], Houndify [131], and IBM Speech to text [132]. These online services offer quite a high recognition success rate in most office environments, however, the audio processing in industrial environments can suffers from many drawbacks, e.g., operator pronunciation, high levels of background noise.

Therefore, the robot command words pronounced by operators maybe recognized as other words. Examples are: "pick up" mis-recognized as "pick cup", "put back" mis-recognized as "pull back", or "three" mis-recognized as "tree". 1-to-1 mapping between what is being recognized and what is being coded will be insufficient. In order to increase system robustness, 1-to-multi mapping is proposed here. A group of similar sounding words can be gathered in coding and regarded as one speech command to form a "fuzzy" command group, see example in Table 1.5. Speech commands are usually based on a limited lexicon, the number of "fuzzy" groups is also limited, which allows the method in coding to be achieved. The "fuzzy" command group method is implemented and verified in the initial system experiment which will be presented in the following sections/chapters.

Table 1.5: Example of "fuzzy" group.

Recognized word	**Command group**
put back Put back put it back Put it back. Put it back Pull back.	put back
ok OK Ok okay Okay	ok
two Two to To	two
...	...

Basically, all studies mentioned above use code words to immediately call some pre-programmed functions of the robot during the interaction, see the example of code words and pre-programmed functions in Table 1.6. Although with the implementation of dynamic sub-tasks from **Layer 3** the input of coordinates (see object positions example in Table 1.7) can be omitted so the barriers for the first-time/no-engineering-background user are decreased, the user still needs to be familiar with the code words (even with small quantities of code words).

Table 1.6: Example of abstract task list and sub-task.

Code word	**Sub-task from Layer 3**
Pick one dish	Trigger **robot arm** with **dish gripper**: 1. Move TCP to **dish position in the container** 2. Close gripper 3. Lift up TCP 4. Move dish to **work table position** 5. Move down dish to the table surface 6. Release gripper 7. Move TCP to **idle position**
Pick one bowl deliver to dish	Trigger **robot arm** with **bowl gripper**: 1. Move TCP to **bowl position in the container** 2. Close gripper 3. Lift up TCP 4. Move bowl to **dish position on the work table** 5. Move down bowl to the dish surface **area A** 6. Release gripper 7. Move TCP to **idle position**
Pick one cup deliver to dish	Trigger **robot arm** with **cup gripper**: 1. Move TCP to **cup position in the container** 2. Close gripper 3. Lift up TCP 4. Move cup to **dish position on the work table** 5. Move down cup to the dish surface **area B** 6. Release gripper 7. Move TCP to **idle position**
Pick one spoon deliver to bowl	Trigger **robot arm** with **spoon gripper**: 1. Move TCP to **spoon position in the container** 2. Close gripper 3. Lift up TCP 4. Move spoon to **bowl position on the dish** 5. Move down spoon inside to the bowl 6. Release gripper 7. Move TCP to **idle position**
Package product	Trigger **packaging machine**: 1. Wrap up all items in the plastic package 2. Put the label with the product information 3. Conveyor move to final container
...	...

Table 1.7: Example of object positions.

Object name	Related to world	Related to self
dish	dish position in the container	/
	dish position on the work table	area A area B area C area D
bowl	bowl position in the container	/
	bowl position on the dish	/
cup	cup position in the container	/
spoon	spoon position in the container	/
work table	work table position	/
robot arm	idle position	/
...	...	...

As presented in [133], a natural language recognition method is implemented for industrial system, in which the operator can speak sentences instead of code words. However, the number of identified sentences is still limited, and only English language is supported. Thanks to the advances in the machine learning domain, more natural language recognition methods have been developed for robot VUI [134]. A deep Convolutional Neural Network (CNN) based natural language recognition method is implemented in the VUI of a PR2 robot [135], which offered high flexibility for the operator in speaking command sentences. However, after receiving one command sentence, the system will execute only one task (one pre-programmed function), which is still similar to the code words method.

A chat bot method for natural language GUI is presented in [134], in which the system can generate a series of robot tasks after the conversation, however, several iterations of ask/answer are required and the experiment with a physical robot is not presented.

In order to further increase the flexibility of the VUI, I propose that the operator should be allowed to freely speak a request in a long command sentence. The system can recognize and autonomously form a series of tasks to complete the operator's request. In the following, the two tableware cleaning company scenarios are presented to illustrate the concept (the tasks are based on Table 1.6).

Command sentence 1: "Hey factory, I want one product with one dish, one bowl, one cup and one spoon."

Automatically identified task list:

- 1: Pick one dish
- 2: Pick one bowl deliver to dish
- 3: Pick one cup deliver to dish
- 4: Pick one spoon deliver to bowl
- 5: Package product

Command sentence 2: "Hey factory, pick one dish, then add two cups, and one bowl with two spoons."

Automatically formed task list:

- 1: Pick one dish
- 2: Pick one cup deliver to dish (area B)
- 3: Pick one cup deliver to dish (area C)
- 4: Pick one bowl deliver to dish (area A)
- 5: Pick one spoon deliver to bowl (first spoon)
- 6: Pick one spoon deliver to bowl (second spoon)
- 7: Package product

These two sentences are completely natural and understandable for a human operator, and the orders and numbers of the tasks are different. The autonomous system should be able to extract the tasks, their order and the number of loops from the natural sentence. Such as, if the operator says "... I want **three** products ..." in the command sentence 1, the autonomously formed task list should be:

Loop 1

- 1: Pick one dish
- 2: Pick one bowl deliver to dish
- 3: Pick one cup deliver to dish
- 4: Pick one spoon deliver to bowl
- 5: Package product

Loop 2

- 6: Pick one dish
- 7: Pick one bowl deliver to dish

- 8: Pick one cup deliver to dish
- 9: Pick one spoon deliver to bowl
- 10: Package product

Loop 3

- 11: Pick one dish
- 12: Pick one bowl deliver to dish
- 13: Pick one cup deliver to dish
- 14: Pick one spoon deliver to bowl
- 15: Package product

These formed a robot motion program consisting of a series of abstract robot tasks. As mentioned in the **Layer 3 robot task planning** section, each sub-task is hardware independent, so the abstract tasks are hardware independent. Thus the whole motion program generated by the natural sentences can be hardware independent. By using this robot programming method, any person can program multiple robot systems by their natural language. There are several online service APIs that are focusing on analyzing the natural languages, e.g., Microsoft Azure Language Understanding [136], Google Cloud Natural Language [137], and Facebook wit.ai [138].

The intuitive interface allows no-technical background end-user to control the industrial system. In this layer, GUI and VUI are discussed as intuitive interaction methods. Furthermore, the prospect of future VUI is also proposed here. However, during this PhD study, the proposed method — autonomous generating of the task list - is not achieved, but the research focus for this layer can be natural language VUI, and the study will continue in the future.

/2

Demonstration

After the introduction of the four layers concept, a series of demonstrations constructed by multiple modules are designed to briefly show a task-dependent distributed control of a manufacturing system which is also a realization of industry 4.0 based on the layered architecture. The scope and module description are presented in the following sections.

2.1 Scope

High flexibility and modularity are the basic characteristics of the initial design of the system. The four-layer system can scale up to a factory (or multiple factories) level that contains multiple manufacturing equipment with all the functionalities and can scale down to one piece of manufacturing equipment with limited functionality. Considering the cost and complexity, this research project will focus on only one robot with some key functionalities to demonstrate the system concept. In addition, the number of technical indicators (e.g., robot tasks, object types, and GUI buttons/VUI commands) will be at a relatively limited level.

2.2 Module demonstration and description

Paper 1, **paper 2** and **paper 3** contribute to the development of modules in **paper 4**, and **paper 4** developed a set of modules that can be important building blocks for the proposed 4-layer-architecture. After the reorganization and further development of the modules from **paper 4**, there are 7 modules (0 - 6) distributed in the 4 layers that can be freely combined according to task complexity, see the structure and modules' connection in Figure 2.1. There are 4 demonstration videos listed in Table 2.1, demonstrations 1, 2, 3 used only a few modules to show the partial functions of the system, and demonstration 4 used all modules to show the complete system functions. More details about the modules will be presented in the following sections.

Table 2.1: The demonstration videos.

No.	**Task description**	**Modules**	**Video link**
1	Test kinematics	0, 3, 4	`https://youtu.be/O_4QQo7vbh4`
2	Test Node-red GUI	0, 3, 5	`https://youtu.be/NypeGnxtWu8`
3	Test body tracking	0, 2, 3, 4, 5	`https://youtu.be/7Du-PwMP_jU`
4	Complete system demo	all 7	`https://youtu.be/UwYRv4HcGLs`

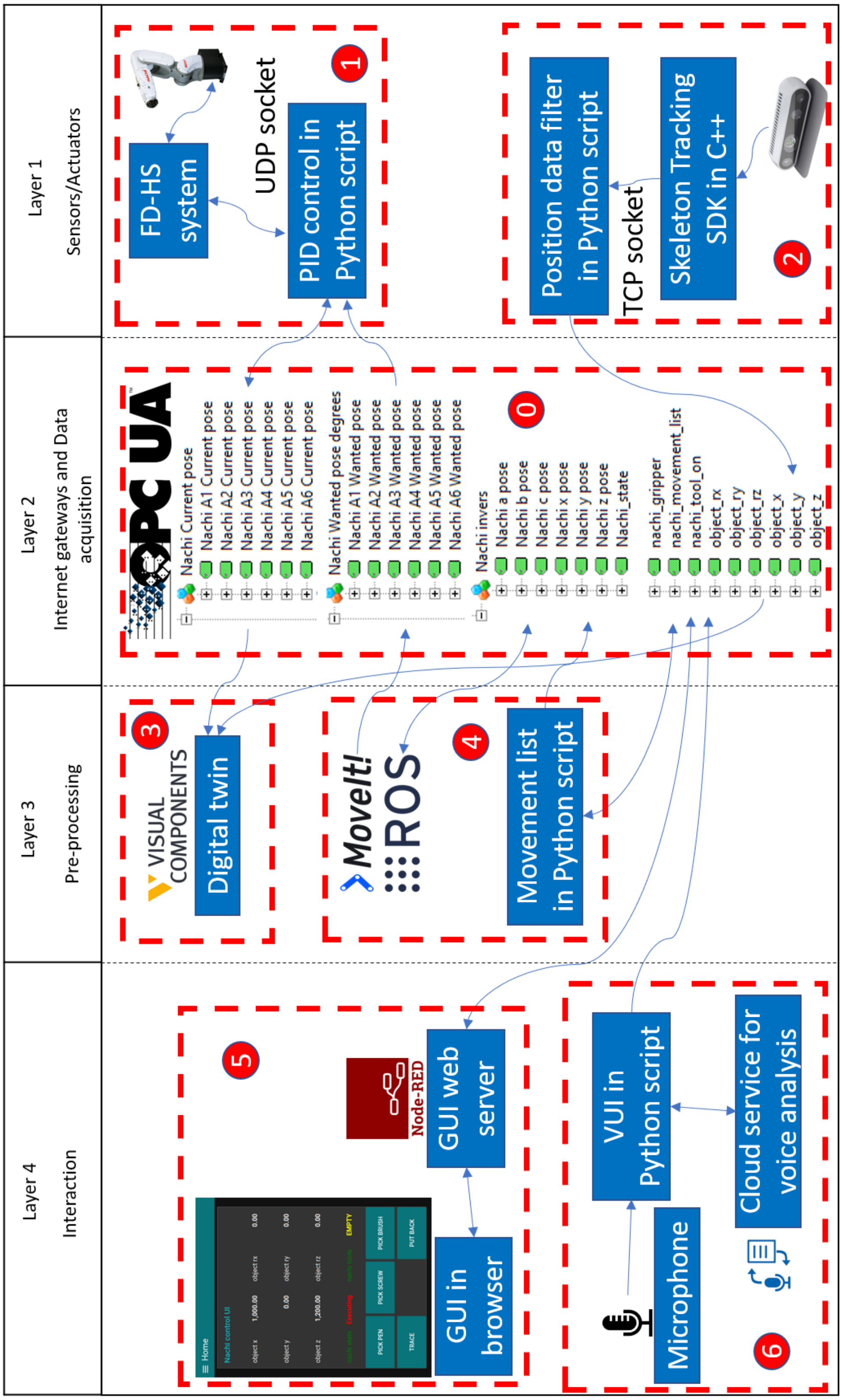


Figure 2.1: Structure of the modules and layers

2.2.1 Module 0 — Information Exchange Module

Information Exchange Module is the core module which is responsible for the information exchange between different modules. OPC-UA server is currently running in the local networks in this module, and all the necessary variables will be used by other modules are previously added, see Figure 2.1. This server can be implemented on the cloud to allow the different modules' connections from different factories, so it means these modules can be distributed anywhere in the world.

2.2.2 Module 1 — Robot Control Module

In the Robot Control Module, only one general industrial robot is implemented, the robot type is NACHI MZ07. The official inverse kinematics API from the manufacturer is not offered. However, by using the FD-HS system to modify the communication data between robot CPU board and driver board (mentioned in **paper 3**) joint control of the robot can be achieved.

A python script is developed for controlling the robot joint movements from OPC-UA server, in which a PID control method constantly reads the Wanted joint values from the server and steadily drives the robot joint to meet the Wanted position and meanwhile updates the current joint values to the server, see the data flow in Figure 2.1.

2.2.3 Module 2 — Object Localization Module

A human-robot collaboration task is designed for a complete function demonstration, in which a human is holding an object and approaching the robot workspace. The robot is able to perceive the object's position and trace it, see Figure 2.2. The direct object localization method is not ready as has been described in **1.2.1 Layer 1** section, and therefore the indirect object localization method is used to fulfill this task. Since the object is held by a human hand, first the human wrist position is detected and then an extra compensate coordinate system is added to calculate the object position. Intel Realsense depth camera D435i with Skeleton Tracking SDK [139] is used to detect the human wrist position. However, the output values from the SDK have a lot of noise and a filter method must be applied to remove this noise. The concept of the filter method is shown in Figure 2.3.

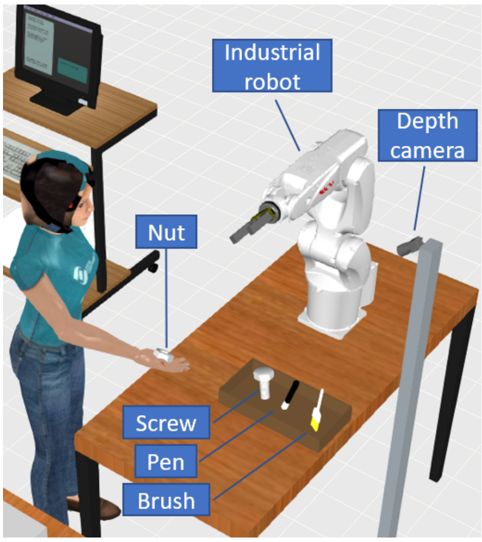


Figure 2.2: The designed human-robot collaboration task

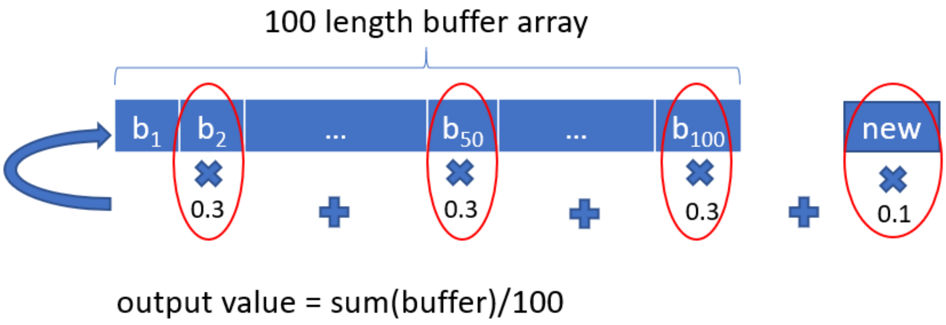


Figure 2.3: Concept of filter method

After the filtration process, the object position values will be upload to the OPC-UA server. In this module, only the position values will be updated, orientation values will be kept to 0.

2.2.4 Module 3 — Digital Twin Module

In this module, Visual Components is the simulation environment for the Digital Twin. The NACHI MZ07 robot's digital model is imported directly from the software's library. In the demonstrations 3, 4 from Table 2.1, three "tools" (pen, screw, and brush) locate in front of the robot digital model, and a nut floating around represents the human holding the object.

After enabling the "connectivity" function in Visual Components, the robot joint values and the nut position values can be paired to the corresponding values in the OPC-UA server. The change of the values in server will then change the pose of the robot and the position of the nut.

2.2.5 Module 4 — Kinematics and Tasks Module

MoveIt IKFast Kinematics Solver is used for inverse kinematics calculation. An example by testing only the inverse kinematics solver is presented in the **1.2.3 Layer 3** section, and the solver's calculation is triggered by running a python script. However, in the complete system demonstration, the kinematics solver will start the calculation by a trigger signal "Nachi_state" in the OPC-UA server. The kinematics solver will read the robot target TCP values from the server and then upload the calculated robot joint values to the robot Wanted pose in the server. After the target TCP is reached, the program will reset the trigger signal "Nachi_state".

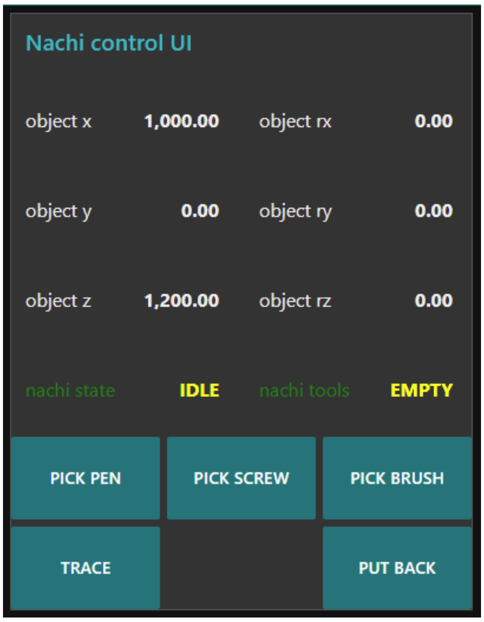
A movement list programmed by python script is predefined in the module. Due to the research scope and demonstration purpose, only five simple movements are defined in this stage, see movements in Table 2.2. Each of the movements will be based on the predefined positions and the object position on the server to update robot TCP values and trigger the "Nachi_state" signal to start the inverse kinematics calculation.

Table 2.2: The movement list.

	Movement name	**Description**
1	pick screw	move robot to screw position, close gripper, lift up, update gripper information "screw on gripper"
2	pick pen	move robot to pen position, close gripper, lift up, update gripper information "pen on gripper"
3	pick brush	move robot to brush position, close gripper, lift up, update gripper information "brush on gripper"
4	put back	put back the object on gripper (screw, pen or brush), update gripper information "gripper empty"
5	trace	read nut position and move robot TCP to nut position

2.2.6 Module 5 — GUI Module

The example of using Node-Red Web-GUI is introduced in **1.2.4 Layer 4** section, in which a robot joint control GUI is presented. A new GUI for the complete system demonstration is also designed, see Figure 2.4. There are five buttons in the GUI that represent the five movements in the **Module 4** respectively. The state of the robot, the tool on the gripper and the position of the object are displayed in the GUI. The operator can use any smartphone, tablet, or PC to open the GUI and control the robot movements.



Nachi control UI

object x 1,000.00
object rx 0.00
object y 0.00
object ry 0.00
object z 1,200.00
object rz 0.00
nachi state IDLE
nachi tools EMPTY

Figure 2.4: The GUI for the experiment

2.2.7 Module 6 — VUI Module

A VUI is also designed for the complete system demonstration. By using a simple microphone, the operator can directly speak the command to trigger the robot motions. An online voice analysis service is used here to convert the audio data from microphone to text (String format in this case), so python script can match the speech command (which is already converted to String format) with the corresponding movement in the movement list which is also in String format. In order to increase the system robustness, two voice analysis service suppliers (Azure and Google) are used together for converting the audio to text in this prototype system, see the VUI performance in the complete system demonstration in Table 2.1.

This demonstration is only one instance of using the layered architecture with one specific task, and only 7 modules are developed. The benefit of the layered architecture is that more modules can be developed separately and then connected to the same system to achieve more complex tasks. Also, it is possible to disconnect some of the modules according to the task or system requirements. The modularization and the scalability of the system are reflected here.

3

Conclusion and Future Works

3.1 Conclusion

With the continuing development of factory complexity, the need for a generalized architecture for the manufacturing system gains importance. One generalized architecture will allow multiple SMEs to use the same key technologies, which increase the speed of deployment for edge technologies. Further analysis of the manufacturing industry system is conducted, and a generalized layered architecture for the industry system is proposed in this PhD research. The proposed architecture creates a clear definition with a specific industrial task, allowing more people to understand the layered concept easily. Then, industrial system developers or SMEs can based on the same manner to develop industrial system.

Based on the proposed architecture, several edge technologies have been studied and analyzed under each of the layers:

- **Layer 1**: object classification and object localization are the two main study focus. In order to find the industrial-ready solution for direct object classification and localization, two open-source software packages are tested. The result shows the technologies of direct object classification and localization are not fully ready. Therefore, source code customization from these open source packages is mandatory to achieve the industrial-ready solution.

- **Layer 2**: robot system information communication platform is discussed. The test of hosting OPC-UA server on Cloud to achieve a longer distance communication is conducted. The result shows OPC-UA supports Cloud-enabled communication. However, more investigation should be performed to reduce the latency.
- **Layer 3**: robot digital twin and robot task planning are the two study focus. A well-designed digital twin simulation environment is tested to connect with the physical industrial robot. This PhD study point out the interaction between operator and digital twin needs some more development. An open-source general inverse kinematics package is also tested to enable the hardware independence of robot task planning. The result shows the presented open-source inverse kinematics package supports easy customization to match any industrial robot, and then the dynamic robot tasks can migrate across different robot systems.
- **Layer 4**: the GUI and VUI are the two study focus. Based on an open-source package, a web browser-based GUI is designed and demonstrated. The proposed GUI unified the user experiences when operators switch from one robot system to another, also decreased the robot system hardware dependency. The VUI allows operators to control the industrial robot system in a more intuitive method. Several VUI technologies have been studied and a prospect for the future VUI is also described.

Traditional production system configuration is on a lower robot programming level. The digitized control method, however, allows the operator to configure the production line on a higher decision level. **Paper 1** introduces a case study of digitized production system control methods with industry 4.0 enabled sensor fusion application. The novel system reduced the configuration of hardware and offered remote monitoring and supervision.

Due to safety regulations, general industrial robot's HRC tasks are difficult to design and test. **Paper 2** attempts to conquer the problem by introducing the digital twin. In **paper 2**, digital twin with sensor technologies enhanced the development of HRC, and a high level HRC task based on simulation is designed.

With the development of industry 4.0 technologies, there are increasing numbers of industrial applications using the digital twin for their system design. The lack of solutions in interactions between human and digital twin is blocking the digital twin from moving forward. **Paper 3** proposed multiple interaction methods between human and digital twin, and presents an overview study combining the virtual world with the physical world.

Many HRC systems with completely different setups have been designed in the manufacturing sector. There is a lack of architecture that can connect different emerging industry 4.0 technologies with one task and offer the possibility of system upgrade on a modular level. **Paper** 4 proposed an HRC architecture with flexible modules which can be upgraded separately, and the possible solutions for each module are also demonstrated.

Most of industrial robot systems have their own programming interface, which increases the difficulties when the operator switches between the different robot systems. The traditional robot teach pendant requires the operator to share the same physical space when the robot is being programmed. **Paper 5** presented a platform independent robot programming GUI that enables the operator to program the robot regardless of geographical distance.

Finally, in this thesis, a demonstration is presented which expresses the generalized layered architecture's flexibility in modularization and scalability. The demonstration also exhibits one actual case of multi-industry 4.0 technologies' fusion.

When SMEs try to implement industry 4.0 technologies into their factory for their system upgrading or building, it is hard to find out what technologies will be helpful and what part of their factory will benefit. This study offers an overview of different edge technologies and states how these technologies can be utilized in an industrial task. Also the availability, accessibility, limitation, current and future development state of the technologies are presented. It makes SMEs have more possibility in the fast implementation of industry 4.0 technologies. I believe, more modules developed under the proposed architecture and more SMEs understand the usage of the modules will significantly promote the implementation of industry 4.0 technologies across the industry and further benefit the whole society.

3.2 Future work

From the proposed technologies in Layer 1 and Layer 4, we can see that the Neural Network technologies benefit from both object localization and natural language recognition — one technology can benefit multiple modules in the same system. There are many edge technologies that are coming out of this rapid development era. However, not all of them can be a key technology. We therefore need to test and examine more key technologies to identify those with maximum benefits to allow the development of more industry-ready modules.

The current object localization method in Layer 1 is based on skeleton detection, and only one person is allowed to be present in front of the depth camera for wrist position detection. The advanced direct object localization and classification method should be further developed.

Cybersecurity is a hot topic in the cloud computing age. As a consequence, in Layer 2, the safety of the information exchange must be considered, especially when we plan to set up the server on the cloud.

More tests using different robot systems and more movement lists in Layer 3 are needed to verify the low hardware dependency and industrial performance.

For Layer 4, the more experimentation relating to the understanding of natural language should be implemented and the advanced GUI with more features should be developed.

Bibliography

[1] Pallavi Sethi and Smruti R. Sarangi. "Internet of Things: Architectures, Protocols, and Applications." In: *J. Electr. Comput. Eng.* (Jan. 2017). URL: `https://www.hindawi.com/journals/jece/2017/9324035` (visited on 11/27/2020).

[2] Jörg Siegert et al. "Ultra-flexible Factories: An Approach to Manage Complexity." In: *Procedia CIRP* 93 (2020). 53rd CIRP Conference on Manufacturing Systems 2020, pp. 329–334. ISSN: 2212-8271. DOI: `https://doi.org/10.1016/j.procir.2020.04.112`. URL: `https://www.sciencedirect.com/science/article/pii/S2212827120307472`.

[3] Siegert Jorg et al. "Software Control for a Cyber-Physical System in a Manufacturing Environment based on a Game Engine." In: *ResearchGate* (June 2019). URL: `https://www.researchgate.net/publication/334634161_Software_Control_for_a_Cyber-Physical_System_in_a_Manufacturing_Environment_based_on_a_Game_Engine` (visited on 12/10/2020).

[4] *CHAPTER 4 Standards and standardization processes*. Nov. 1998. URL: `http://heim.ifi.uio.no/~oleha/Publications/bok.4.html` (visited on 12/01/2020).

[5] Jochen Wolf. *STEP-NC: A new programming interface for modern nc-controls. STEP-NC supports bidirectional information flow between the shop floor and the planning*. Jan. 2007. URL: `http://www.step-nc.org/index.htm` (visited on 12/01/2020).

[6] *ISO/TS 14649-201:2011*. Dec. 2020. URL: `https://www.iso.org/standard/60042.html` (visited on 12/01/2020).

[7] *Industry 5.0, a transformative vision for Europe*. Jan. 2022. URL: `https://ec.europa.eu/info/publications/industry-50-transformative-vision-europe_en` (visited on 05/09/2022).

[8] Mohd Javaid et al. "Enabling flexible manufacturing system (FMS) through the applications of industry 4.0 technologies." In: (May 2022). DOI: `10.1016/j.iotcps.2022.05.005`. (Visited on 05/16/2022).

[9] Chunguang Bai, Guido Orzes, and Joseph Sarkis. "Exploring the impact of Industry 4.0 technologies on social sustainability through a circular economy approach." In: (Feb. 2022). DOI: `10.1016/j.indmarman.2021.12.004`. (Visited on 05/16/2022).

[10] Enrique Coronado et al. "Evaluating quality in human-robot interaction: A systematic search and classification of performance and human-centered factors, measures and metrics towards an industry 5.0." In: (Apr. 2022). DOI: `10.1016/j.jmsy.2022.04.007`. (Visited on 05/16/2022).

[11] Roger Nagel. “21ST Century Manufacturing Enterprise Strategy Report.” In: (Jan. 1991). URL: `https://www.researchgate.net/publication/235112061_21ST_Century_Manufacturing_Enterprise_Strategy_Report` (visited on 05/09/2022).

[12] Nour Mohammad Yaghoubi and Mahboobeh Rahat Dahmardeh. *Analytical approach to effective factors on organizational agility*. Oct. 2010. URL: `https://www.textroad.com/pdf/JBASR/JBASR%201(1)%2067-87.pdf` (visited on 05/09/2022).

[13] Rick Dove. *Lean and Agile: Synergy, Contrast, and Emerging Structure*. May 2022. URL: `https://www.academia.edu/22250454/Lean_and_Agile_Synergy_Contrast_and_Emerging_Structure` (visited on 05/09/2022).

[14] Steven L. Goldman et al. *Agile competitors and virtual organizations : strategies for enriching the customer*. New York, NY, USA: New York (N.Y.) : Van Nostrand Reinhold, 1995., 1995. (Visited on 05/09/2022).

[15] *J.P. Womack, D.T. Jones, and D. Ross, The Machine that Changed the World, Rawson Associates, New York, 1990*. Mar. 2022. (Visited on 05/09/2022).

[16] H. Sharifi and David Zhang. “Agile manufacturing in practice - Application of a methodology.” In: (May 2001). DOI: `10.1108/01443570110390462`. (Visited on 05/09/2022).

[17] Sameer Sharma et al. “Effect of agility in different dimensions of manufacturing systems: A review.” In: (Mar. 2022). DOI: `10.1016/j.matpr.2022.03.054`. (Visited on 04/05/2022).

[18] C. Hemalatha, K. Sankaranarayanasamy, and N. Durairaaj. “Lean and agile manufacturing for work-in-process (WIP) control.” In: (Jan. 2021). DOI: `10.1016/j.matpr.2020.12.473`. (Visited on 03/14/2022).

[19] Bouchra Abdelilah, Akram El Korchi, and Mohammed Amine Balambo. “Flexibility and agility: evolution and relationship.” In: (June 2018). DOI: `10.1108/JMTM-03-2018-0090`. (Visited on 04/29/2022).

[20] Guangxi Wan et al. “Design and implementation of agent-based robotic system for agile manufacturing: A case study of ARIAC 2021.” In: (Oct. 2022). DOI: `10.1016/j.rcim.2022.102349`. (Visited on 04/05/2022).

[21] Aakanksha Tewari and B. B. Gupta. “Security, privacy and trust of different layers in Internet-of-Things (IoTs) framework.” In: *Future Gener. Comput. Syst.* (June 2020). URL: `http://www.sciencedirect.com/science/article/pii/S0167739X17321003` (visited on 11/11/2020).

[22] Fatos Xhafa, Burak Kilic, and Paul Krause. “Evaluation of IoT stream processing at edge computing layer for semantic data enrichment.” In: *Future Gener. Comput. Syst.* (Apr. 2020). URL: `http://www.sciencedirect.com/science/article/pii/S0167739X19321296` (visited on 11/11/2020).

[23] Hugh Boyes et al. “The industrial internet of things (IIoT): An analysis framework.” In: *Comput. Ind.* (Oct. 2018). URL: `http://www.sciencedirect.com/science/article/pii/S0166361517307285` (visited on 11/11/2020).

[24] Mahmoud Parto, Christopher Saldana, and Thomas Kurfess. “A Novel Three-Layer IoT Architecture for Shared, Private, Scalable, and Real-time Machine Learning from Ubiquitous Cyber-Physical Systems.” In: *Procedia Manuf.* (Jan. 2020). URL: `http://www.sciencedirect.com/science/article/pii/S2351978920315870` (visited on 11/12/2020).

[25] Muhammad Habib ur Rehman et al. “The Role of Big Data Analytics in Industrial Internet of Things.” In: *ResearchGate* (Apr. 2019). URL: `https:`

//www.researchgate.net/publication/332368604_The_Role_of_Big_Data_Analytics_in_Industrial_Internet_of_Things (visited on 11/12/2020).

[26] Marc-Fabian Körner et al. "Extending the Automation Pyramid for Industrial Demand Response." In: *Procedia CIRP* 81 (2019). 52nd CIRP Conference on Manufacturing Systems (CMS), Ljubljana, Slovenia, June 12-14, 2019, pp. 998–1003. ISSN: 2212-8271. DOI: https://doi.org/10.1016/j.procir.2019.03.241. URL: https://www.sciencedirect.com/science/article/pii/S2212827119305463.

[27] Tiago Coito et al. "A Novel Framework for Intelligent Automation." In: *IFAC-PapersOnLine* 52.13 (2019). 9th IFAC Conference on Manufacturing Modelling, Management and Control MIM 2019, pp. 1825–1830. ISSN: 2405-8963. DOI: https://doi.org/10.1016/j.ifacol.2019.11.501. URL: https://www.sciencedirect.com/science/article/pii/S2405896319314855.

[28] Yazen Alsafi and Valeriy Vyatkin. "Ontology-based reconfiguration agent for intelligent mechatronic systems in flexible manufacturing." In: *Rob. Comput. Integr. Manuf.* (Aug. 2010). URL: http://www.sciencedirect.com/science/article/pii/S0736584509001239 (visited on 11/13/2020).

[29] Jiewu Leng et al. "Digital twin-driven rapid reconfiguration of the automated manufacturing system via an open architecture model." In: *Rob. Comput. Integr. Manuf.* (June 2020). URL: http://www.sciencedirect.com/science/article/pii/S073658451930167X (visited on 11/13/2020).

[30] Ali Ahmad Malik and Alexander Brem. "Digital twins for collaborative robots: A case study in human-robot interaction." In: *Rob. Comput. Integr. Manuf.* (Apr. 2021). URL: http://www.sciencedirect.com/science/article/pii/S0736584520303021 (visited on 12/15/2020).

[31] R. Harikrishnan et al. "Productivity improvement in poly-cover packing line through line balancing and automation." In: *Mater. Today:. Proc.* (Jan. 2020). URL: http://www.sciencedirect.com/science/article/pii/S2214785320320332 (visited on 12/15/2020).

[32] Tiago Coito et al. "A Middleware Platform for Intelligent Automation: An Industrial Prototype Implementation." In: *Computers in Industry* 123 (2020), p. 103329. ISSN: 0166-3615. DOI: https://doi.org/10.1016/j.compind.2020.103329. URL: https://www.sciencedirect.com/science/article/pii/S0166361520305637.

[33] Daniel Stock, Daniel Schel, and Thomas Bauernhansl. "Middleware-based Cyber-Physical Production System Modeling for Operators." In: *Procedia Manuf.* (Jan. 2020). URL: http://www.sciencedirect.com/science/article/pii/S2351978920305783 (visited on 11/19/2020).

[34] Georgios Bouloukakis et al. "Automated synthesis of mediators for middleware-layer protocol interoperability in the IoT." In: *Future Generation Computer Systems* 101 (2019), pp. 1271–1294. ISSN: 0167-739X. DOI: https://doi.org/10.1016/j.future.2019.05.064. URL: https://www.sciencedirect.com/science/article/pii/S0167739X18323586.

[35] Charles Steinmetz et al. "Using Ontology and Standard Middleware for integrating IoT based in the Industry 4.0." In: *IFAC-PapersOnLine* 51.10 (2018). 3rd IFAC Conference on Embedded Systems, Computational Intelligence and Telematics in Control CESCIT 2018, pp. 169–174. ISSN: 2405-8963. DOI:

https://doi.org/10.1016/j.ifacol.2018.06.256. URL: https://www.sciencedirect.com/science/article/pii/S2405896318305767.

[36] *E3S-DC*. [Online; accessed 30. Dec. 2020]. Dec. 2020. URL: https://industrial.omron.eu/en/products/e3s-dc.

[37] *MFS211 - Full-metal magnetic sensor - ifm electronic*. [Online; accessed 30. Dec. 2020]. Dec. 2020. URL: https://www.ifm.com/de/en/product/MFS211.

[38] *TMP235A4DCKR Texas Instruments | Sensors, Transducers | DigiKey*. [Online; accessed 30. Dec. 2020]. Dec. 2020. URL: https://www.digikey.no/product-detail/en/texas-instruments/TMP235A4DCKR/296-50016-2-ND/8597481.

[39] *HS1101LF HUMIDITY SENSOR*. [Online; accessed 30. Dec. 2020]. Dec. 2020. URL: https://www.te.com/global-en/product-HPP801A031.html.

[40] *Barcode Readers | Sortation & Routing Readers | Cognex Bar Code Reading Systems*. Jan. 2021. URL: https://www.cisco-eagle.com/catalog/category/9151/barcode-readers (visited on 01/05/2021).

[41] Cheng-Jin Du and Da-Wen Sun. “Shape extraction and classification of pizza base using computer vision.” In: *J. Food Eng.* (Oct. 2004). URL: http://www.sciencedirect.com/science/article/pii/S0260877403004655 (visited on 01/04/2021).

[42] An-An Liu et al. “Hierarchical multi-view context modelling for 3D object classification and retrieval.” In: *Inform. Sci.* (Feb. 2021). URL: http://www.sciencedirect.com/science/article/pii/S0020025520309671 (visited on 01/05/2021).

[43] Kyungpyo Kim et al. “Deep learning-based dynamic object classification using LiDAR point cloud augmented by layer-based accumulation for intelligent vehicles.” In: *Expert Syst. Appl.* (Aug. 2020). URL: http://www.sciencedirect.com/science/article/pii/S0957417420306710 (visited on 01/05/2021).

[44] Leandro B. Marinho et al. “A novel mobile robot localization approach based on classification with rejection option using computer vision.” In: (2018). ISSN: 0045-7906. DOI: https://doi.org/10.1016/j.compeleceng.2018.03.047. URL: https://www.sciencedirect.com/science/article/pii/S0045790617337758.

[45] Wei Ding et al. “Semantic recognition of workpiece using computer vision for shape feature extraction and classification based on learning databases.” In: *Optik* (Feb. 2017). URL: http://www.sciencedirect.com/science/article/pii/S0030402616314917 (visited on 01/05/2021).

[46] *Categorization and Product Sortation – Logistics | Cognex*. Jan. 2021. URL: https://www.cognex.com/en-no/industries/logistics/sortation/categorizing-packages-prior-to-shipment (visited on 01/05/2021).

[47] Andrew D. O’ Riordan et al. “Object recognition within smart manufacturing.” In: *Procedia Manuf.* (Jan. 2019). URL: http://www.sciencedirect.com/science/article/pii/S2351978920300536 (visited on 01/05/2021).

[48] Kirti Chawla, Gabriel Robins, and Liuyi Zhang. “Object localization using RFID.” In: *IEEE 5th International Symposium on Wireless Pervasive Computing 2010* (May 2010). URL: https://ieeexplore.ieee.org/abstract/document/5483750 (visited on 01/07/2021).

[49] Klaas Gadeyne and Herman Bruyninckx. “Markov techniques for object localization with force-controlled robots.” In: (2001).

[50] Stéphane Bazeille et al. "Model based object localization and shape estimation using electric sense on underwater robots." In: (July 2017). DOI: 10.1016/j.ifacol.2017.08.941. (Visited on 05/16/2022).

[51] N. Strobel, S. Spors, and R. Rabenstein. "Joint audio-video object localization and tracking." In: *IEEE Signal Process. Mag.* (Jan. 2001). URL: https://ieeexplore.ieee.org/abstract/document/911196 (visited on 01/07/2021).

[52] Leandro B. Marinho et al. "A novel mobile robot localization approach based on classification with rejection option using computer vision." In: (2018). ISSN: 0045-7906. DOI: https://doi.org/10.1016/j.compeleceng.2018.03.047. URL: https://www.sciencedirect.com/science/article/pii/S0045790617337758.

[53] Yasushi Sumi, Yutaka Ishiyama, and Fumiaki Tomita. "Robot-vision architecture for real-time 6-DOF object localization." In: *Comput. Vision Image Understanding* (Mar. 2007). URL: http://www.sciencedirect.com/science/article/pii/S1077314206001755 (visited on 01/07/2021).

[54] Fanxiu Chen et al. "Full-field 3D measurement using multi-camera digital image correlation system." In: *Opt. Lasers Eng.* (Sept. 2013). URL: http://www.sciencedirect.com/science/article/pii/S0143816613000869 (visited on 01/08/2021).

[55] Jong-Kyu Oh, Sukhan Lee, and Chan-Ho Lee. "Stereo vision based automation for a bin-picking solution." In: *Int. J. Control Autom. Syst.* (Apr. 2012). URL: https://doi.org/10.1007/s12555-012-0216-9 (visited on 01/08/2021).

[56] Paulo Ricardo Marques de Araujo and Romulo Gonçalves Lins. "Cloud-based approach for automatic CNC workpiece origin localization based on image analysis." In: *Robotics and Computer-Integrated Manufacturing* 68 (2021), p. 102090. ISSN: 0736-5845. DOI: https://doi.org/10.1016/j.rcim.2020.102090. URL: https://www.sciencedirect.com/science/article/pii/S0736584520303008.

[57] Gergely Horváth and Gábor Erdős. "Object localization utilizing 3D point cloud clustering approach." In: *Procedia CIRP* 93 (2020). 53rd CIRP Conference on Manufacturing Systems 2020, pp. 508–513. ISSN: 2212-8271. DOI: https://doi.org/10.1016/j.procir.2020.04.132. URL: https://www.sciencedirect.com/science/article/pii/S2212827120307745.

[58] *find_object_2d - ROS Wiki*. [Online; accessed 8. Jan. 2021]. Jan. 2021. URL: http://wiki.ros.org/find_object_2d.

[59] Hedi Harzallah, Frédéric Jurie, and Cordelia Schmid. "Combining efficient object localization and image classification." In: *2009 IEEE 12th International Conference on Computer Vision*. 2009, pp. 237–244. DOI: 10.1109/ICCV.2009.5459257. URL: https://ieeexplore.ieee.org/abstract/document/5459257.

[60] intel. *ros_object_analytics*. Jan. 2021. URL: https://github.com/intel/ros_object_analytics (visited on 01/11/2021).

[61] *Guide your robot with Pickit 3D vision solutions - Pickit 3D - robot vision made easy*. Mar. 2022. URL: https://www.pickit3d.com/en (visited on 03/24/2022).

[62] ISRA VISION GmbH. *ISRA VISION - Leading Machine Vision Systems*. Mar. 2020. URL: https://www.isravision.com/en (visited on 03/24/2022).

[63] *SICK USA*. Feb. 2022. URL: https://www.sick.com/us/en (visited on 03/24/2022).

[64] *Robot Programming Software - fast, easy, automatic: AUTOMAPPPS*. Mar. 2022. URL: http://convergent-it.com (visited on 03/24/2022).

[65] Zivid. *Industrial 3D cameras for random bin picking, logistics, machine tending - Zivid*. Mar. 2022. URL: https://www.zivid.com (visited on 03/14/2022).

[66] *Machine Vision and Automation Solutions | Photoneo Focused on 3D*. Mar. 2022. URL: https://www.photoneo.com (visited on 03/24/2022).

[67] Magnus Åkerman, Åsa Fast-Berglund, and Sven Ekered. "Interoperability for a Dynamic Assembly System." In: *Procedia CIRP* (Jan. 2016). URL: http://www.sciencedirect.com/science/article/pii/S2212827116002961 (visited on 01/18/2021).

[68] *MQTT - The Standard for IoT Messaging*. Jan. 2021. URL: https://mqtt.org (visited on 01/19/2021).

[69] *Unified Architecture - OPC Foundation*. Sept. 2019. URL: https://opcfoundation.org/about/opc-technologies/opc-ua (visited on 01/19/2021).

[70] Tsubasa Kubota, Reza Hamzeh, and Xun Xu. "STEP-NC Enabled Machine Tool Digital Twin." In: *Procedia CIRP* (Jan. 2020). URL: http://www.sciencedirect.com/science/article/pii/S2212827120310155 (visited on 01/20/2021).

[71] Gustavo Caiza et al. "Human Machine Interfaces Based on Open Source Web-Platform and OPC UA." In: *Procedia Manufacturing* 42 (2020). International Conference on Industry 4.0 and Smart Manufacturing (ISM 2019), pp. 307–314. ISSN: 2351-9789. DOI: https://doi.org/10.1016/j.promfg.2020.02.089. URL: https://www.sciencedirect.com/science/article/pii/S2351978920306545.

[72] Carlos A. Garcia, William Montalvo-Lopez, and Marcelo V. Garcia. "Human-Robot Collaboration Based on Cyber-Physical Production System and MQTT." In: *Procedia Manuf.* (Jan. 2020). URL: http://www.sciencedirect.com/science/article/pii/S2351978920306533 (visited on 01/20/2021).

[73] Gustavo Caiza et al. "Flexible robotic teleoperation architecture for intelligent oil fields." In: *Heliyon* (Apr. 2020). URL: http://www.sciencedirect.com/science/article/pii/S2405844020306782 (visited on 01/20/2021).

[74] Sebastian Thiede et al. "Integrative simulation of information flows in manufacturing systems." In: *Procedia CIRP* 81 (2019). 52nd CIRP Conference on Manufacturing Systems (CMS), Ljubljana, Slovenia, June 12-14, 2019, pp. 647–652. ISSN: 2212-8271. DOI: https://doi.org/10.1016/j.procir.2019.03.170. URL: https://www.sciencedirect.com/science/article/pii/S2212827119304755.

[75] Zongmin Jiang, Yan Chang, and Xuefen Liu. "Design of software-defined gateway for industrial interconnection." In: *Journal of Industrial Information Integration* (June 2020). URL: http://www.sciencedirect.com/science/article/pii/S2452414X20300054 (visited on 01/20/2021).

[76] Hadil Abukwaik et al. "OPC UA Realization for simplified commissioning of adaptive sensing applications for the 5G IIoT." In: *Internet of Things* (Sept. 2020). URL: http://www.sciencedirect.com/science/article/pii/S2542660520300548 (visited on 01/27/2021).

[77] Yu-Ju Lin, Ci-Bin Lan, and Chin-Yin Huang. “A Realization of Cyber-Physical Manufacturing Control System Through Industrial Internet of Things.” In: *Procedia Manuf.* (Jan. 2019). URL: `http://www.sciencedirect.com/science/article/pii/S2351978920305217` (visited on 01/27/2021).

[78] Mina Younan et al. “Challenges and recommended technologies for the industrial internet of things: A comprehensive review.” In: *Measurement* (Feb. 2020). URL: `http://www.sciencedirect.com/science/article/pii/S0263224119310644` (visited on 01/27/2021).

[79] Bufan Liu et al. “Edge-cloud orchestration driven industrial smart product-service systems solution design based on CPS and IIoT.” In: *Adv. Eng. Inf.* (Oct. 2019). URL: `http://www.sciencedirect.com/science/article/pii/S1474034619305579` (visited on 01/27/2021).

[80] Erik Kajati et al. “Cloud based cyber-physical systems: Network evaluation study.” In: *Adv. Eng. Inf.* (Oct. 2019). URL: `http://www.sciencedirect.com/science/article/pii/S1474034619305610` (visited on 01/27/2021).

[81] *opcua*. Feb. 2021. URL: `https://pypi.org/project/opcua` (visited on 02/01/2021).

[82] *OPC UA Clients - Unified Automation*. Feb. 2021. URL: `https://www.unified-automation.com/downloads/opc-ua-clients.html` (visited on 02/01/2021).

[83] Michael Grieves and John Vickers. “Digital Twin: Mitigating Unpredictable, Undesirable Emergent Behavior in Complex Systems.” In: *Transdisciplinary Perspectives on Complex Systems: New Findings and Approaches*. Cham, Switzerland: Springer, Aug. 2016. URL: `https://doi.org/10.1007/978-3-319-38756-7_4` (visited on 02/04/2021).

[84] Benjamin Schleich et al. “Shaping the digital twin for design and production engineering.” In: *CIRP Ann.* (Jan. 2017). URL: `http://www.sciencedirect.com/science/article/pii/S0007850617300409` (visited on 02/04/2021).

[85] Thijs Defraeye et al. “Digital twins are coming: Will we need them in supply chains of fresh horticultural produce?” In: (2021). ISSN: 0924-2244. DOI: `https://doi.org/10.1016/j.tifs.2021.01.025`. URL: `https://www.sciencedirect.com/science/article/pii/S092422442100025X`.

[86] Rainer Stark, Carina Fresemann, and Kai Lindow. “Development and operation of Digital Twins for technical systems and services.” In: *CIRP Ann.* (Jan. 2019). URL: `http://www.sciencedirect.com/science/article/pii/S0007850619300502` (visited on 02/04/2021).

[87] Ali Ahmad Malik and Alexander Brem. “Digital twins for collaborative robots: A case study in human-robot interaction.” In: *Rob. Comput. Integr. Manuf.* (Apr. 2021). URL: `https://www.sciencedirect.com/science/article/pii/S0736584520303021` (visited on 02/04/2021).

[88] Osrf. *Gazebo*. Feb. 2021. URL: `http://gazebosim.org` (visited on 02/04/2021).

[89] *Webots: robot simulator*. Feb. 2021. URL: `https://cyberbotics.com` (visited on 02/05/2021).

[90] *Plant Simulation and Throughput Optimization | Siemens Software*. [Online; accessed 25. Mar. 2022]. Apr. 2020. URL: `https://www.plm.automation.siemens.com/global/en/products/manufacturing-planning/plant-simulation-throughput-optimization.html`.

[91] *RobotStudio | ABB Robotics*. [Online; accessed 25. Mar. 2022]. Mar. 2022. URL: `https://new.abb.com/products/robotics/robotstudio`.

[92] *KUKA AG*. [Online; accessed 25. Mar. 2022]. Mar. 2022. URL: `https://www.kuka.com/en-de/products/robot-systems/software/planning-project-engineering-service-safety/kuka_sim`.

[93] *Visual Components: 3D manufacturing simulation and visualization software - Design the factories of the future*. Feb. 2021. URL: `https://www.visualcomponents.com` (visited on 02/05/2021).

[94] *Simulator for industrial robots and offline programming - RoboDK*. Feb. 2021. URL: `https://robodk.com` (visited on 02/05/2021).

[95] *Getting Started — moveit_tutorials Kinetic documentation*. Mar. 2021. URL: `http://docs.ros.org/en/kinetic/api/moveit_tutorials/html/doc/getting_started/getting_started.html` (visited on 03/21/2022).

[96] *IKFast Kinematics Solver — moveit_tutorials Kinetic documentation*. May 2020. URL: `http://docs.ros.org/en/kinetic/api/moveit_tutorials/html/doc/ikfast/ikfast_tutorial.html` (visited on 02/11/2021).

[97] Phylliade. *ikpy*. Feb. 2021. URL: `https://github.com/Phylliade/ikpy` (visited on 02/11/2021).

[98] RethinkRobotics-opensource. *sns_ik*. Feb. 2021. URL: `https://github.com/RethinkRobotics-opensource/sns_ik` (visited on 02/11/2021).

[99] Michael A. Goodrich and Alan C. Schultz. “Human–Robot Interaction: A Survey.” In: *Foundations and Trends® in Human–Computer Interaction* 1.3 (2008), pp. 203–275. ISSN: 1551-3955. DOI: `10.1561/1100000005`. URL: `http://dx.doi.org/10.1561/1100000005`.

[100] Elisa Prati et al. “How to include User eXperience in the design of Human-Robot Interaction.” In: *Rob. Comput. Integr. Manuf.* (Apr. 2021). URL: `https://www.sciencedirect.com/science/article/pii/S0736584520302805` (visited on 02/22/2021).

[101] Antti Hietanen et al. “AR-based interaction for human-robot collaborative manufacturing.” In: *Rob. Comput. Integr. Manuf.* (June 2020). URL: `https://www.sciencedirect.com/science/article/pii/S0736584519307355` (visited on 02/17/2021).

[102] Christos Maragkos, George-Christopher Vosniakos, and Elias Matsas. “Virtual reality assisted robot programming for human collaboration.” In: *Procedia Manuf.* (Jan. 2019). URL: `https://www.sciencedirect.com/science/article/pii/S2351978920301104` (visited on 02/17/2021).

[103] Eva Coupeté, Fabien Moutarde, and Sotiris Manitsaris. “Gesture Recognition Using a Depth Camera for Human Robot Collaboration on Assembly Line.” In: *Procedia Manuf.* (Jan. 2015). URL: `https://www.sciencedirect.com/science/article/pii/S2351978915002176` (visited on 02/17/2021).

[104] Gilbert Tang, Phil Webb, and John Thrower. “The development and evaluation of Robot Light Skin: A novel robot signalling system to improve communication in industrial human–robot collaboration.” In: *Rob. Comput. Integr. Manuf.* (Apr. 2019). URL: `https://www.sciencedirect.com/science/article/pii/S0736584518300553` (visited on 02/17/2021).

[105] Patrik Gustavsson et al. “Human-robot Collaboration Demonstrator Combining Speech Recognition and Haptic Control.” In: *Procedia CIRP* (Jan. 2017). URL: `https://www.sciencedirect.com/science/article/pii/S2212827117302755` (visited on 02/17/2021).

[106] Federica Ferraguti et al. “Walk-through Programming for Industrial Applications.” In: *Procedia Manuf.* (Jan. 2017). URL: `https://www.sciencedirect.com/science/article/pii/S235197891730330X` (visited on 02/17/2021).

[107] A. Al-Yacoub et al. “Improving human robot collaboration through Force/-Torque based learning for object manipulation.” In: *Rob. Comput. Integr. Manuf.* (June 2021). URL: `https://www.sciencedirect.com/science/article/pii/S0736584520303215` (visited on 02/17/2021).

[108] *CB3 Family | Collaborative robots by Universal Robots*. Mar. 2022. URL: `https://www.universal-robots.com/cb3` (visited on 03/30/2022).

[109] *KUKA AG*. Mar. 2022. URL: `https://www.kuka.com/en-de/products/robot-systems/industrial-robots/lbr-iiwa` (visited on 03/30/2022).

[110] *Sawyer Collaborative Robots for Industrial Automation*. Mar. 2022. URL: `https://www.rethinkrobotics.com/sawyer` (visited on 03/30/2022).

[111] *Motoman HC Series Collaborative Robots for Assembly & Handling*. Mar. 2022. URL: `https://www.motoman.com/en-us/products/robots/collaborative/hc-series` (visited on 03/30/2022).

[112] *ABB’s Collaborative Robot -YuMi*. Mar. 2022. URL: `https://new.abb.com/products/robotics/collaborative-robots/irb-14000-yumi` (visited on 03/30/2022).

[113] *Robot System - Franka Emika*. Mar. 2022. URL: `https://www.franka.de/robot-system` (visited on 03/30/2022).

[114] *Collaborative robots from UR | Start your automation journey*. Mar. 2022. URL: `https://www.universal-robots.com/products` (visited on 03/30/2022).

[115] *CRB 1100*. Mar. 2022. URL: `https://new.abb.com/products/robotics/collaborative-robots/crb-1100` (visited on 03/30/2022).

[116] *KUKA AG*. Mar. 2022. URL: `https://www.kuka.com/en-de/products/robot-systems/industrial-robots/lbr-iisy` (visited on 03/30/2022).

[117] *KUKA AG*. Mar. 2022. URL: `https://www.kuka.com/en-de/products/robot-systems/software/application-software/kuka-forcetorquecontrol` (visited on 03/30/2022).

[118] *Hex 6 Axis Force Torque Sensor*. Mar. 2022. URL: `https://onrobot.com/en/products/hex-6-axis-force-torque-sensor` (visited on 03/30/2022).

[119] *Robotiq Force Torque Sensor*. Aug. 2018. URL: `https://robotiq.com/products/ft-300-force-torque-sensor` (visited on 03/30/2022).

[120] *Product TeslaSCADA2 - Multi-platform SCADA system*. Feb. 2021. URL: `https://teslascada.com/products/teslascada2` (visited on 02/19/2021).

[121] Ppm-Robotics-As. *flexgui4.0*. Feb. 2021. URL: `https://github.com/PPM-Robotics-AS/flexgui4.0` (visited on 02/19/2021).

[122] *node-red-contrib-opcua*. Feb. 2021. URL: `https://flows.nodered.org/node/node-red-contrib-opcua` (visited on 02/19/2021).

[123] Tudor B. Ionescu and Sebastian Schlund. “Programming cobots by voice: A human-centered, web-based approach.” In: *Procedia CIRP* (Jan. 2021). URL: `https://www.sciencedirect.com/science/article/pii/S2212827120314347` (visited on 02/23/2021).

[124] Sebastian van Delden et al. “Pick-and-place application development using voice and visual commands.” In: *Industrial Robot: An International Journal* (Oct. 2012). URL: `https://www.emerald.com/insight/content/doi/10.1108/01439911211268796/full/html` (visited on 02/24/2021).

[125] trevorbye. *About the Speech SDK*. Feb. 2021. URL: `https://docs.microsoft.com/en-us/azure/cognitive-services/speech-service/speech-sdk?tabs=windows%2Cubuntu%2Cios-xcode%2Cmac-xcode%2Candroid-studio` (visited on 02/24/2021).

[126] Sotiris Makris et al. “Intuitive dual arm robot programming for assembly operations.” In: *CIRP Ann.* (Jan. 2014). URL: `https://www.sciencedirect.com/science/article/pii/S0007850614000201` (visited on 02/23/2021).

[127] cmusphinx. *pocketsphinx*. Feb. 2021. URL: `https://github.com/cmusphinx/pocketsphinx` (visited on 02/24/2021).

[128] Junhong Meng, Zhiquan Feng, and Tao Xu. *A Method of Fusing Gesture and Speech for Human-robot Interaction*. New York, NY, USA: Association for Computing Machinery, Jan. 2020. URL: `https://dl.acm.org/doi/abs/10.1145/3379247.3379261` (visited on 02/23/2021).

[129] *Cognitive Speech Services | Microsoft Azure*. Feb. 2021. URL: `https://azure.microsoft.com/en-us/services/cognitive-services/speech-services` (visited on 02/24/2021).

[130] *Speech-to-Text: Automatic Speech Recognition | Google Cloud*. Dec. 2020. URL: `https://cloud.google.com/speech-to-text` (visited on 02/24/2021).

[131] *Houndify | Add voice enabled, conversation interface to anything*. Feb. 2021. URL: `https://www.houndify.com` (visited on 02/24/2021).

[132] *Speech to Text - IBM Cloud*. May 2016. URL: `https://cloud.ibm.com/catalog/services/speech-to-text` (visited on 02/24/2021).

[133] Adam Rogowski. “Industrially oriented voice control system.” In: *Rob. Comput. Integr. Manuf.* (June 2012). URL: `https://www.sciencedirect.com/science/article/pii/S0736584511001189` (visited on 02/24/2021).

[134] Sara Beschi, Daniela Fogli, and Fabio Tampalini. “CAPIRCI: A Multi-modal System for Collaborative Robot Programming.” In: *End-User Development*. Cham, Switzerland: Springer, July 2019. URL: `https://link.springer.com/chapter/10.1007/978-3-030-24781-2_4` (visited on 02/24/2021).

[135] Jinpeng Mi et al. “Object affordance based multimodal fusion for natural Human-Robot interaction.” In: *Cognit. Syst. Res.* (May 2019). URL: `https://www.sciencedirect.com/science/article/pii/S1389041718302018` (visited on 02/25/2021).

[136] *Language Understanding | Microsoft Azure*. Feb. 2021. URL: `https://azure.microsoft.com/en-us/services/cognitive-services/language-understanding-intelligent-service` (visited on 02/26/2021).

[137] *Cloud Natural Language | Google Cloud*. Feb. 2021. URL: `https://cloud.google.com/natural-language` (visited on 02/26/2021).

[138] *Wit.ai*. Feb. 2021. URL: `https://wit.ai` (visited on 02/24/2021).

[139] *Skeleton Tracking SDK by cubemos™ - Intel RealSense Technologies*. Feb. 2021. URL: `https://www.intelrealsense.com/skeleton-tracking` (visited on 03/11/2021).

Paper 1

Introduction of cyber-physical system in robotized press-brake line for metal industry

Beibei Shu, Gabor Sziebig, Bjørn Solvang

DOI: 10.1007/978-981-10-5768-7_20

Author's Contribution

Beibei Shu has contributed substantially in the proposal of research idea, concept, literature review, graphics produce and writing of the paper.

The paper published version has errors at the end of Introduction section.

As shown in the following:

..."This is achieved with the following tools: system based components are connected to the network with the usage of Robot Operating System (introduced in Chapter ~~“Application of Long Short-Term Memory Neural Network to Sales Forecasting in Retail—A Case Study”~~), supervision of the robotized solution is solved by a software solution (called FlexGui 4.0), which is also introduced in Chapter ~~“Application of Long Short-Term Memory Neural Network to Sales Forecasting in Retail—A Case Study”~~. In Chapter ~~“”~~, the application of the solutions mentioned above will be presented, while Chapter ~~“A Coarse-to-Fine Matching Method in the Line Laser Scanning System”~~ provides the conclusion."...

Then the correct version is enclosed in this thesis.

Introduction of cyber-physical system in robotized press-brake line for metal industry

Beibei Shu (✉), Gabor Sziebig, Bjørn Solvang

UiT, campus Narvik, Lodve Langesgate 2, 8514 Narvik, Norway
`e-mail: beibei.shu@uit.no`

Abstract. Bin picking is a typical work, which is easy to automate up to a given complexity of the work-piece dimensions. In case of casted work-pieces, the dimensions are most of the time not accurate enough for an industrial robot to be able to pick it up without additional sensors/intelligence. In this paper we introduce a cyber-physical system, where all sensors, actuators, machines and industrial robot is connected to a local network, where they share information easily with each other. The novelty of the system that the proposed solution is achieved on the software side, with minimum hardware reconfiguration need. We keep the flexibility of the industrial robot, but extend its' understanding with sensor fusion on a higher decision level, rather than on low robot programming level. The system also allows remote monitoring and supervision of the production plant.

Keywords: industrial robot, industry 4.0, cyber-physical system, ROS

1 Introduction

In recent years, industry 4.0 has become a very popular topic in industry-related fairs, conferences, or call for public-funded projects. The concept of industry 4.0 was to develop from Internet of Things, which was created by British technology pioneer Kevin Ashton in 1999 (1). Then in April 2013, the Industry 4.0 Working Group present a final report to the German federal government. The report categorizes human industry into four stages, from steam engine stage – first industrial revolution, electrically-powered mass production stage – second industrial revolution and electronics-information stage – third industrial revolution, to cyber-physical based production stage – fourth industrial revolution (2). The fourth stage is the Industry 4.0 which includes cyber-physical system, Internet of things and cloud computing (3). As for now, we are in the industrial revolution but not finish this revolution. Germany is planning to promote the revolution process and Germany is not the only country planning to upgrade their industry. The industry upgrading becomes a trend across the world recently, the representative countries' plan including: Japan – Super Smart Society (4), Norway – Norge 6.0 (5), United States – The Advanced Manufacturing Partnership (6), China – Made in China 2025 (7).

Cyber-physical systems (CPS) are enabling technologies, which bring the virtual and physical worlds together to create a truly networked world in which intelligent objects communicate and interact with each other (8). In this paper, we will be introducing a solution for a robotized press-brake line, where all system are connected to the network (a true cyber-physical system) and the progress of the production can be either supervised on-site, in a web browser or in a simulation. This is achieved with the following tools: system based components are connected to the network with the usage of Robot Operating System (introduced in Chapter 2), supervision of the robotized solution is solved by a software solution (called FlexGui 4.0), which is also introduced in Chapter 2. In Chapter 3, the application of the solutions mentioned above will be presented, while Chapter 4 provides the conclusion.

2 Software based control

The Robot Operating System (ROS) is an open source, multi-platform robot controlling software, which is a collection of variety tools and libraries (9). ROS is widely used among the researchers, but it is also developed together with industry and is highly promoted between industrial partners, as all results that are achieved in research can be implicitly implemented in industrial version. In ROS, there is a very basic concept – *Nodes*, which are processes that perform computation. For example, one node controls a servo motor, one node control optical sensors and one node control electro-pneumatic valves, see in **Fig. 1**. Multiple nodes can run in a same ROS device, and multiple ROS devices can connect together. In between all the connected ROS device or inside a ROS device, each node can communicate with each other by passing *Messages*, which is simply a data structure. Under nodes, there are *Topics* and *Services* and a node can contain many topics and services. A topic is like a message bus, see in **Fig. 2**. You can get messages from a subscribed topic if someone publishes data in it, or send messages to other parties by yourself. A service is like a function, see in **Fig. 3**. You can call it to run a specific script and wait for a call-back function, but the service can only response one request each time.

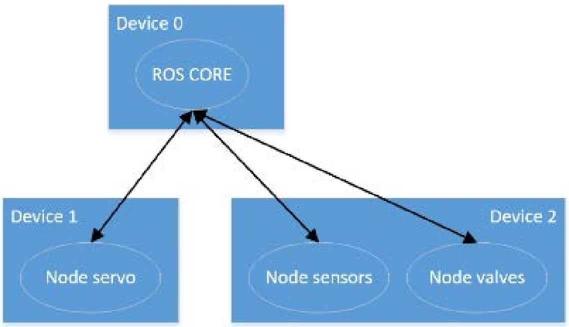


Fig. 1. ROS node

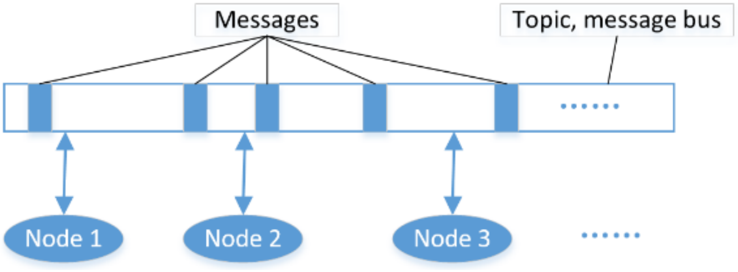


Fig. 2. ROS topic

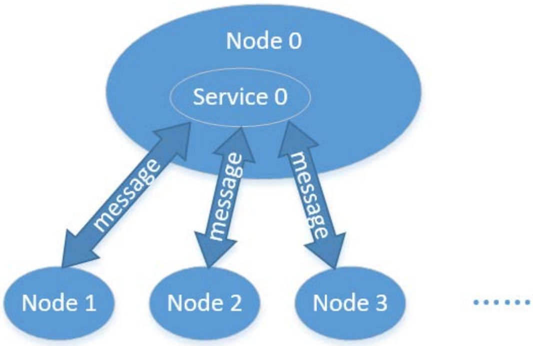


Fig. 3. ROS service

FlexGui 4.0 is an advanced open source user friendly industrial robot controlling interface, which is based upon popular web technologies enabling the possible to run it on almost every modern browser supported device. The user will have exactly the same user experience on each device. FlexGui 4.0 is used to show information and send user inputs to the server, which is the ROS device. Since FlexGui 4.0 is based on ROS, after connected to ROS, FlexGui 4.0 will display all the nodes in the system, and offer a graphic interface for user to read or set variables. For customized variables, user can publish them to topics or send request to services. Based on this mechanism, FlexGui 4.0 is not suit for real-time control functions. While, it is possible for ROS connected to multiple FlexGui 4.0 clients, each FlexGui 4.0 client will synchronize with the server and show the same project, even if it changes only one of them. This makes it is possible for users to develop a project simultaneously (10).

3 Application

Gazebo is a well-designed free robot simulator, and has already embedded into ROS. When the Gazebo is running, ROS server will generate a ROS node called “gazebo”. Meanwhile, FlexGui 4.0 is connect to ROS server, and it will recognize the node gazebo. User can change a robot-moving project without interfere the process line, and test it virtually in Gazebo. If the project passes the test, user can connect physical robot with ROS server, conduct actual test or introduce it into processing line. When the physical robot is running, user can see a simulated robot additionally in Gazebo, **Fig. 4**.

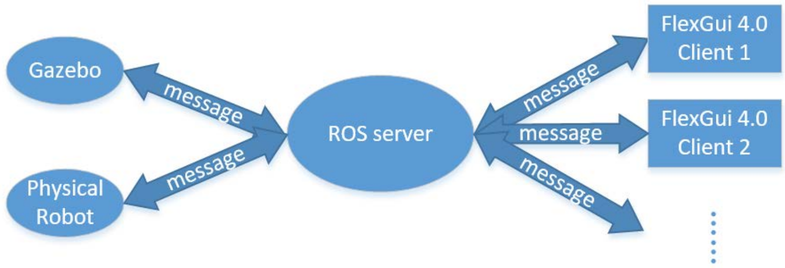
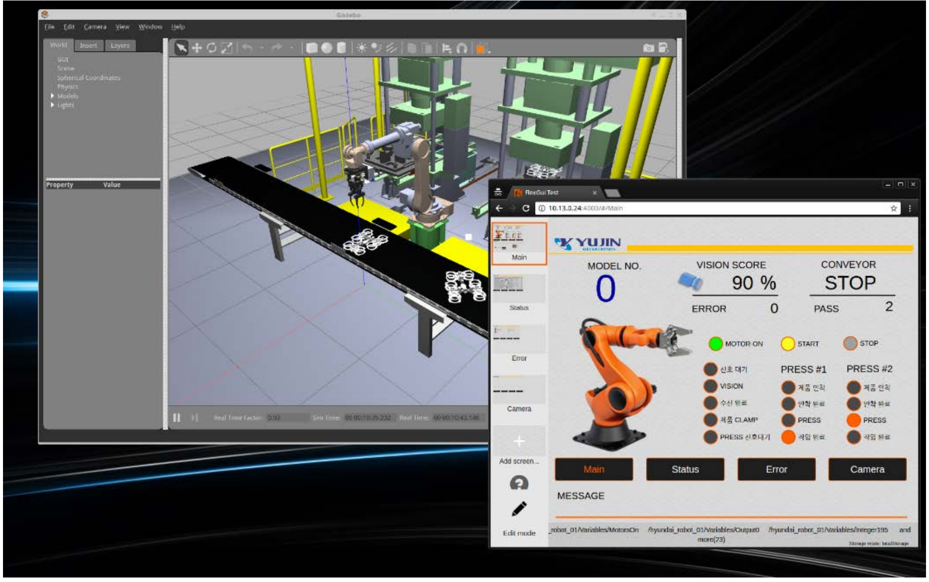


Fig. 4. Connect Gazebo and Physical robot

In **Fig. 5,** we can see a virtual press-brake line in Gazebo and a robot control interface in FlexGui 4.0. During the simulation or real manufacturing process, the Gazebo will show the robot's activity and FlexGui 4.0 will offer a detail state of each sensor. And for each element on the FlexGui 4.0 interface such as button in this case, we can change the number, the name, the location, the size or the background function of the button. Multiple robot instructions can be integrated in one button. So, the robot will execute a series of actions just after one button clicked.



MODEL NO. 0

VISION SCORE 90 %

ERROR 0

CONVEYOR STOP

PASS 2

_robot_01/Variables/MotorsOn /hyundai_robot_01/Variables/Output0 /hyundai_robot_01/Variables/Integer195 and more(23)

Fig. 5. Gazebo simulation and FlexGui 4.0 interface

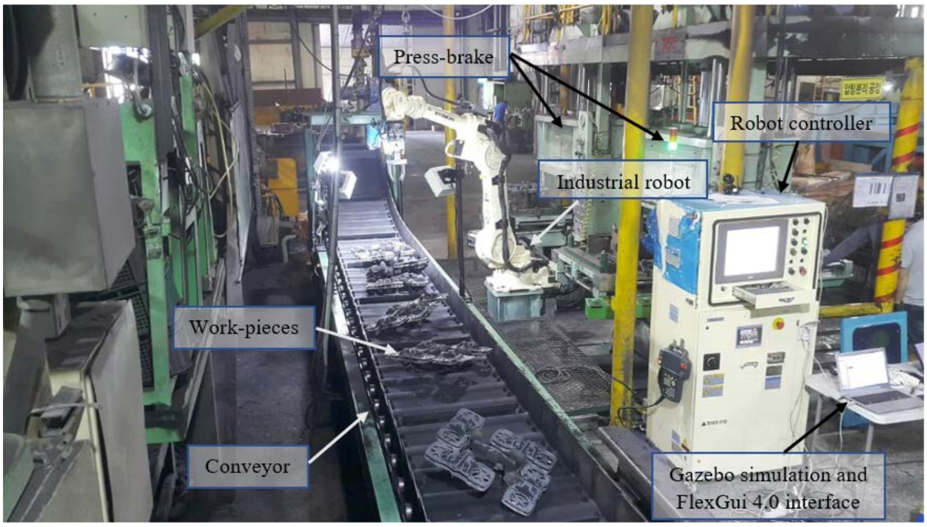


Fig. 6. Industry robot and press-break line in real case

In **Fig. 6**, that is a real case in physical world using this ROS, FlexGui 4.0 and Gazebo solution. Work-pieces that are coming from the moulding machine, robot picks these up and sets on the press-brake, then two pieces running the production parallel. The conveyor, press-brake, robot controller and all sensors connect to ROS sever, each device will be assigned a node name in ROS. With the help of ROS network, all the objects can easily share information with each other including Gazebo and FlexGui 4.0, achieving Gazebo simulation and FlexGui 4.0 interface control. A demonstration of this communication and supervision for the application can be viewed on the following place: http://t.cn/RSVQGgz

4 Conclusion

In this paper, we introduce a cyber-physical system, where all sensors, actuators, machines and industrial robot is connected to a local network – ROS network, where they share information easily with each other by ROS message. The novelty of the system that the proposed solution is achieved on the software side, with minimum hardware reconfiguration need. The only hardware configuration is to connect all the present signal to a ROS supported computer, so ROS can access all the devices in the production line. With the aid of FlexGui 4.0, we keep the flexibility of the industrial robot, but extend its' understanding with sensor fusion on a higher decision level, rather than on low robot programming level. Gazebo synchronously simulate production line with the physical world that allows remote monitoring and supervision of the production plant.

Acknowledgement

The research was supported by Korea Institute for Advancement of Technology (Kiat) grant funded by the Korea government (N0001698), Eurostars programme (E-9692) and YUJIN Mechatronics Co., Ltd.

References

1. Werr P. IoT Evolution World. [Online].; 2015 [cited 2017 May 29. Available from: http://www.iotevolutionworld.com/m2m/articles/401292-how-industry-40-the-internet-things-connected.htm.
2. Kagermann H, Wahlster W, Helbig J. Final report of the Industrie 4.0 Working Group. ; 2013.
3. Morgan J. A Simple Explanation Of 'The Internet Of Things'. [Online].; 2014 [cited 2017 May 29. Available from: http://www.forbes.com/sites/jacobmorgan/2014/05/13/simple-explanation-internet-things-that-anyone-can-understand/#3f87e8a66828.
4. Harayama Y. Society 5.0. [Online].; 2016 [cited 2017 May 29. Available from: http://fpcj.jp/wp/wp-content/uploads/2016/07/f2d3eec7bf7678840f8adf2ca8000b05.pdf.
5. Qvam W. Norge 6.0. [Online].; 2015 [cited 2017 May 29. Available from: https://khrono.no/sites/default/files/walter_qvam_hjernekraft2015.pdf.
6. The White House Office of the Press Secretary. President Obama Launches Advanced Manufacturing Partnership Steering Committee “2.0”. [Online].; 2013 [cited 2017 May 29. Available from: https://obamawhitehouse.archives.gov/the-press-office/2013/09/26/president-obama-launches-advanced-manufacturing-partnership-steering-com.
7. Wübbeke J, Meissner M, Zenglein MJ, Ives J, Conrad B. MADE IN CHINA 2025: The making of a high-tech superpower and consequences for industrial countries. ; 2016.
8. Germany Trade & Invest. INDUSTRIE 4.0 - Smart Manufacturing for the Future. [Online].; 2016 [cited 2017 May 29. Available from: https://www.gtai.de/GTAI/Content/EN/Invest/_SharedDocs/Downloads/GTAI/Brochures/Industries/industrie4.0-smart-manufacturing-for-the-future-en.pdf?v=8.
9. Thomas D. ROS/Introduction. [Online].; 2014 [cited 2017 May 29. Available from: http://wiki.ros.org/ROS/Introduction.
10. PPMAS. www.ppm.no. [Online].; 2016 [cited 2017 May 29. Available from: https://www.ppm.no/flexgui4-Home/Index/downloads.

Paper 2

Human-Robot Collaboration: Task sharing through Virtual Reality

Beibei Shu, Gabor Sziebig, Sakari Pieska

DOI: 10.1109/IECON.2018.8591102

Author's Contribution

Beibei Shu has contributed substantially in the proposal of research idea, concept, literature review, graphics produce, programming, experimental analysis and writing of the paper.

Paper 3

Architecture for Safe Human-Robot Collection: Multi-Modal Communication in Virtual Reality for Efficient Task Execution

Beibei Shu, Gabor Sziebig, Roel Pieters

DOI: 10.1109/ISIE.2019.8781372

Author's Contribution

Beibei Shu has contributed substantially in the proposal of research idea, concept, literature review, graphics produce, programming, experimental analysis and writing of the paper.

Paper 4

Architecture for task-dependent human-robot collaboration

Beibei Shu, Bjørn Solvang

DOI: 10.1109/IEEECONF49454.2021.9382703

Author's Contribution

Beibei Shu has contributed substantially in the proposal of research idea, concept, literature review, graphics produce, programming, experimental analysis and writing of the paper.

Paper 5

Platform independent interface for programming of industrial robots

Beibei Shu, Halldor Arnarson, Bjørn Solvang, Tero Kaarlela, Sakari Pieskä

DOI: 10.1109/SII52469.2022.9708905

Author's Contribution

Beibei Shu has contributed substantially in the proposal of research idea, concept, literature review, graphics produce, programming, experimental analysis and writing of the paper.