

Article

Low-Cost and High-Efficiency Electromechanical Integration for Smart Factories of IoT with CNN and FOPID Controller Design under the Impact of COVID-19

Chang-Hung Hsu ^{1,*}, Shan-Jen Cheng ¹, Te-Jen Chang ², Yi-Mei Huang ³, Chin-Ping Fung ¹ and Shih-Feng Chen ⁴

¹ Department of Mechanical Engineering, Asia Eastern University of Science and Technology, New Taipei 220, Taiwan; cheng5721@gmail.com (S.-J.C.); cpfung@mail.aeust.edu.tw (C.-P.F.)

² Department of Electrical and Electronic Engineering, Chung Cheng Institute of Technology, National Defense University, Daxi District, Taoyuan 335, Taiwan; karl591218@gmail.com

³ Department of Mechanical Engineering, National Central University, Chungli District, Taoyuan 320, Taiwan; t330005@cc.ncu.edu.tw

⁴ Department of Mechanical Engineering, Lunghwa University of Science and Technology, Guishan District, Taoyuan 333, Taiwan; dave2esc@gm.lhu.edu.tw

* Correspondence: chshiu@mail.aeust.edu.tw; Tel.: +886-2-77-388-000 (ext. 3133); Fax: +886-2-77-380-145

Abstract: This study proposes a design for unmanned chemical factories and implementation based on ultra-low-cost Internet of Things technology, to combat the impact of COVID-19 on industrial factories. A safety and private blockchain network architecture was established, including a three-layer network structure comprising edge, fog, and cloud calculators. Edge computing uses a programmable logic controller and a single-chip microcomputer to transmit and control the motion path of a four-axis robotic arm motor. The fog computing architecture is implemented using Python software. The structure is integrated and applied using a convolutional neural network (CNN) and a fractional-order proportional-integral-derivative controller (FOPID). In addition, edge computing and fog computing signals are transmitted through the blockchain, and can be directly uploaded to the cloud computing controller for signal integration. The integrated application of the production line sensor and image recognition based on the network layer was addressed. We verified the image recognition of the CNN and the robot motor signal control of the FOPID. This study proposes that a CNN + FOPID method can improve the efficiency of the factory by more than 50% compared with traditional manual operators. The low-cost, high-efficiency equipment of the new method has substantial contribution and application potential.

Keywords: smart factory; IoT; machine learning; robot control; COVID-19



Citation: Hsu, C.-H.; Cheng, S.-J.; Chang, T.-J.; Huang, Y.-M.; Fung, C.-P.; Chen, S.-F. Low-Cost and High-Efficiency Electromechanical Integration for Smart Factories of IoT with CNN and FOPID Controller Design under the Impact of COVID-19. *Appl. Sci.* **2022**, *12*, 3231. <https://doi.org/10.3390/app12073231>

Academic Editors: Arkadiusz Gola, Izabela Nielsen and Patrik Grznár

Received: 8 January 2022

Accepted: 9 March 2022

Published: 22 March 2022

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Copyright: © 2022 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

The COVID-19 crisis has had a substantial effect on the economy of many countries with increased risk to the lives of many people. The Internet of Things (IoT) has the potential to improve the transformation of manufacturing technology, and it has also attracted attention from academia and industry. The IoT envisages the seamless integration of the physical world and cyberspace by being ubiquitous. Devices are widely deployed with embedded identification (ID), sensing, and driving functions, and can be extended to the physical field of the IoT [1–4]. Miniature electronic devices are embedded to interact in the physical world and are connected to the network system to make them intelligent and seamlessly integrated into the final network infrastructure. Therefore, the IoT can be extended to include manufacturing resources/capabilities in different stages of the manufacturing and business planning processes. In addition, it enables vertical integration at different hierarchical system levels. This mechanism can provide existing or new manufacturing services and applications with unprecedented opportunities [5,6].

An advanced interconnected network system includes connections between smart machines, storage systems, and production facilities. It can exchange network information independently, trigger actions, and control systems. In addition, the universal perception ability of the IoT systems generate a large and diverse amount of data, which can be used to assist manufacturing engineering in achieving the best decision-making mechanisms for all aspects of the manufacturing process. For this reason, sensors and driving devices can cover the entire range of industrial tools. They are still in the early stages of development; however, manufacturing datasets are growing rapidly, and IoT devices can be deployed quickly. In addition, cloud-based computing and big data technologies are indispensable [7,8].

These technologies play an important role in the management and large-scale manufacturing of resources. They provide users with great flexibility and service opportunities, such as signal storage, signal processing, visual manufacturing of big data, and other powerful functions. Modern factories are often equipped with robots that satisfy the requirements of the production line. The IoT enables users to confirm the position and shape of the robot's work, as well as to manage the interactive information transmission of the state of multiple objects, regardless of the situation.

The robot has a third type of perception ability, which is a sensor system platform. It is not necessary to communicate unilaterally with other robots [9,10]. To communicate the signals, the motion behavior of the robot can be monitored and understood at any time. This is the interaction between handling work and the dimensions of the working space. The robot itself has multiple signals linked to environmental signals, which poses a great challenge in obtaining greater performance stability of the robot. Therefore, for a factory with an automation system, a 3D image detection device with the ability to sense depth is required, which can be used to enable the interception of data in 3D space for detection robots. To realize the robot's operating status, it can immediately allow the robot to reflect the specific requirements of the working environment in the space, regardless of touch or the characteristics of the objects to be transported. It includes the properties of weight, shape, and size, and a suitable and optimized calculation capability for load handling can be made immediately. Industrial technology must be constructed using a collaborative system for IoT applications. The following issues must be addressed in advance to enable industrial manufacturing capabilities.

- Immediate response.

Faster connections for the control of equipment can master the complete production process and furnish customers with production history information.

- Prevent errors.

To ensure product quality, it is necessary to monitor abnormalities in the production process. To reduce the loss of defective products, possible equipment failures or errors must be monitored according to enterprise requirements.

- Ensure the manufacturing process.

The production control process must be correctly implemented, and the human error rate must be minimized to improve the consistency of the manufacturing process.

- Improve production utilization rate.

Complete equipment production accuracy must be provided to reduce the appearance of defective products and increase the utilization rate of the equipment. Modern factory equipment typically uses programmable logic controllers (PLCs) to build robotic manufacturing processes. Programming and functions can be developed to satisfy the requirements of the equipment.

With the advancement of science and technology, equipment must also improve over time to achieve the required communication technology capabilities and functions. A data-communication network linking the production equipment is needed. This enables the robot PLCs on the production line to confirm the location and interaction of multiple

surrounding objects under any circumstance. The situation message can then be delivered. In other words, the robot PLC should have a higher sensing and perception function, which ensures that the exchange of signals with the other robots is independent (and event-driven). The robot's motion and behavior in the working space is interactive.

In ref [11], a sensing method was proposed to enable the widespread adoption and development of IoT technology in the manufacturing industry to be close to the user. This has played an important role in the manufacturing industry. For example, radio frequency identification (RFID) has been applied to electromagnetic field effects to transmit data for automatic identification and tracking of tags attached to the target product. The RFID system is composed of a tag and a reader. The tag stores the information being read. In other words, tag data can also be obtained through decryption without requiring additional signals to be provided to other systems. Therefore, the RFID reader can indirectly track the physical movement state mode of the tag. Therefore, it can indirectly track the physical movement of the object to which the tag is attached. In industrial manufacturing procedures, RFID can be applied to the management of the supply chain, production scheduling, part/vehicle tracking, etc. The sensors, wireless networks, and sensor networks (wireless sensor networks, WSNs) can also be configured out of space from a node structure. The node structure is used to sense the surrounding environment, and can then be manipulated and other node communication signals obtained. Sensor nodes operate in a self-organizing and decentralized manner to maintain the best connection status as much as possible. RFID and WSNs represent two complementary technologies. RFID can be used to discover and identify the locations of objects that are difficult to detect or distinguish using traditional sensor technology. However, they cannot monitor the state of the objects. In contrast, wireless sensor networks not only provide information about objects and environmental conditions but also support multi-hop wireless communication. Some WSNs may be equipped with actuators to perform the appropriate actions. Finally, RFID and WSNs could be combined to facilitate industrial development. In this study, we propose many controllers for constructing smart factories, including sensors, RFID, and WSN control methods. However, it is unfortunate that production line process objects are not fully monitored by lens imaging technology with artificial intelligence (AI) algorithms. In addition, the color of the production line with the recognition method has rarely been mentioned.

Intelligent manufacturing technology, which is the core concept of the fourth industrial revolution (Industry 4.0), has received increasing attention worldwide. In recent years, technologies such as the IoT, big data analysis, AI, cloud computing, and cyber-physical systems, have been developed to promote intelligent manufacturing [12–14]. In this study, we discuss research related to intelligent manufacturing, including big data and AI, to optimize mechanical equipment, the integration of computer numerical controlled machine tools and robots, and the parameter design and optimization of intelligent factories. Previous studies have reported on an effective technical solution for integrating and controlling all joints, driven by hydraulic actuators for heavy robots, in which the robot controller is constructed based mainly on industrial PLC units. The PLC controller was designed to link the central control unit and all components of the entire forging station at the same time. In addition, a hybrid robot that establishes a new method for kinematics and dynamics has been proposed. The main contribution is to demonstrate the hybrid robot system with generalized velocity analytic relations, and finally, to demonstrate and verify the newly designed hybrid robot, which is feasible in kinematic and dynamic modelling results.

The main contribution of this study is the research results of the proposed AI image recognition system constructed for the PLC production line. The organizational structure of this study is as follows. Section 1 presents an introduction. Section 2 describes the construction method of the IoT in a smart factory. Section 3 describes the realization of a smart factory based on a PLC production line. Section 4 presents the experimental results and discussion, and Section 5 presents the conclusions.

2. Materials and Methods

2.1. Construction of IoT in the Smart Factory

The IoT devices include hardware and software that can be used to establish and realize factories. It covers the main physical resource structure, and all manufacturing resources that are involved in the life cycle of the manufacturing process. These resources represent the realization of smart manufacturing. Based on the production line, efficient manufacturing equipment and production lines were developed using data collection for product development, as shown in Figure 1. The smart factory is a cyber-physical production system (CPPS) with the characteristics of a network entity, which integrates smart sensors, embedded terminal systems, smart control systems, and communication facilities [15]. Through a CPPS, peer-to-peer interactive transmission is realized, including transmission between users and equipment, between devices, and between services.

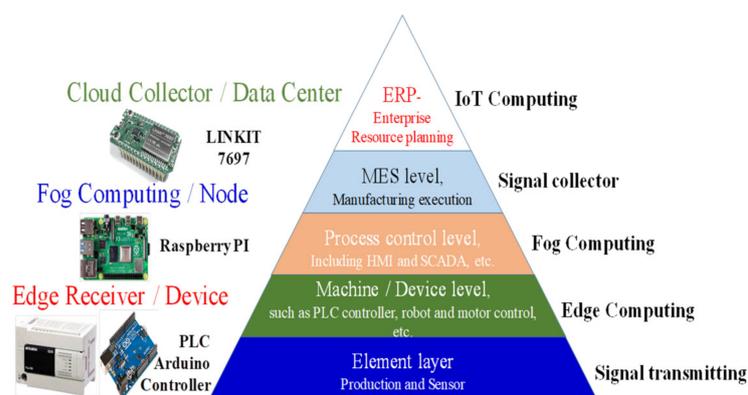


Figure 1. Schematic diagram of IoT smart-factory communication system.

Smart Factory Network Construction

The construction of smart factories should consider the characteristics of manufacturing plants to meet rapidly changing market demand. The following planning auto-mated production was used for a smart-factory laboratory platform, and many typical features of smart factories were explored. Figure 1 is a prototype mechanical electronic construction platform for industrial production lines [16]. According to the conceptual smart-factory architecture, the prototype platform of the intelligent structure has four layers: a physical resource layer, cloud service layer, terminal layer, and network layer. The prototype layer is introduced as follows.

- Physical resource layer.

This consists of basic smart sensors or equipment, conveying equipment, and pack-aging products. It is mainly responsible for performing tasks, such as processing, monitoring, and assembly. After receiving basic analogue/digital data sources, manufacturing process information is very important to enable the transfer of signals for upper-level applications.

- Cloud surface layer.

This contains a cloud platform (service cluster system based on the Hadoop architecture), which provides data storage and computing resources for data applications. The ontological model of the packaging line was constructed on a cloud platform, and the relationship between the two dimensions of structure and interaction was established objectively. The complex restricted grammar of the Web Rule Language is mainly used to upload the manufacturing data to the cloud platform and to the data model. A knowledge-based reasoning system was used for device operations. Resource allocation and factory-optimized scheduling work are supported to provide a fault alarm [17].

- Terminal layer.

This mainly includes end-user equipment such as smartphones, computers, and circuit boards. They are distributed in different manufacturing locations, monitoring centers, and other areas. Terminal devices are used to visualize the results of cloud processing and support remote monitoring operations and maintenance. In addition, customers can use smart terminals to check the order status at any time.

- Network layer.

This connects the networks in a smart factory. Following this rule, the connection between the distributed control, controller and actuator, Modbus, and EtherCAT is realized. The connection between the devices is realized through a combination of Ethernet and a data distribution service, which is a self-organizing network structure. In addition, the connection between the devices and the cloud platform can be achieved through a combination of Ethernet and Open Platform Communications (OPC, unified architecture-automation technology, and machine-to-machine network transmission protocol), which was used to provide data exchange [16,18].

The design factors of a smart factory must consider parameters such as equipment availability, equipment performance, and product qualification rate. It has a cloud-assisted manufacturing system that allows equipment to self-organize the scheduling and optimization process and propose effective solutions. Many verification experiments have proposed a specific time slot to verify the operation of production line equipment. Among them, power measurement by smart meters is used to evaluate the total power of laboratory equipment, and to calculate the power parameters and operation cycle values, and upload the data to the cloud. It is necessary to compare the efficiency improvement rate for the manufacturing and test equipment. This result verifies that the cloud-assisted manufacturing system and self-organizing scheduling both have a significant positive effect.

The network layer is an important component of a smart factory. The system devices induced many signals when it started its operation. Such data are typically unstructured and cannot be used. Big data and IoT occur [19,20]. Unstructured data can be easily analyzed. This analysis provides valuable information for the factory production lines. The equipment needs to be ready to intercept the data, transmit it to a platform, and wait for it to be analyzed. The equipment must be equipped with many sensors and support international communication standard protocols, such as the Semiconductor Equipment Materials Initiative, Equipment Communications Standard/Generic Equipment Model, OPC, and Transmission Control Protocol/Internet Protocol (TCP/IP).

3. PLC Production Line Signal and IoT Connection

A smart factory is required to establish an internal ecosystem in which devices and applications are connected through standard protocols. Key applications, such as the manufacturing execution system (MES), enterprise resource planning (ERP), and product lifecycle management tools, should be integrated with each other. The PLC equipment core is integrated into the MES to adjust each step, as shown in Figure 2. Devices, such as handheld scanners, mobile phones, and tablets, should communicate with the applications in the workshop. This ensured a closed-loop information collection and control system.

The PLC is an industrial-grade device. Therefore, it does not allow direct connection to the Internet, as it can only communicate with other devices through a special protocol developed by the PLC manufacturer. Furthermore, these protocols have limited functionality, and require specific additional knowledge and different software to be fully functional. The main purpose of this study is to recommend a simple, efficient, and low-cost technique to connect any type of PLC to the Internet, thereby moving data to high-performance servers where it can be processed quickly and easily. This adds a new dimension to data storage, processing, and interpretation as data moves from limited-resource devices such as PLCs to unlimited-capacity devices such as cloud servers. The device that plays a major role in the entire framework is the Raspberry PI, which helps to transfer data from the PLC to the cloud server by facilitating the conversion between the PLC and the server. The Raspberry PI establishes a connection with the PLC through the Modbus TCP. The main function of

the Raspberry PI is to map data from the PLC and encapsulate it in a hypertext transfer protocol (HTTP) request that the server can handle. After the connection is established, the following steps are repeated for signal processing at specific time intervals.

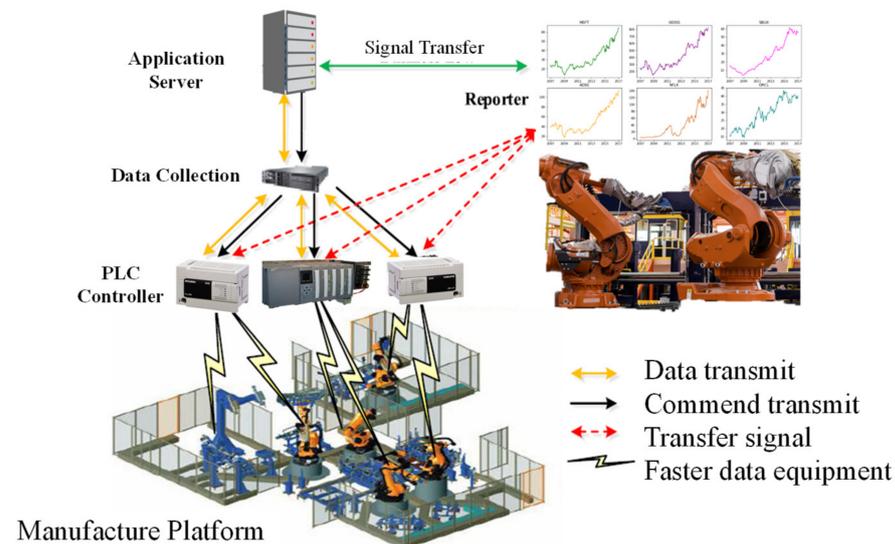


Figure 2. The smart factory uses PLC devices to monitor various equipment signals.

- Step 1: The PLC performs electromechanical integration through drivers and sensors to form the basic edge computing controller (edge layer) of the smart factory and stores parameters for electrical devices such as motor speed, current, operating temperature, machine operating status, and alarms in temporary storage in the memory.
- Step 2: Machine image recognition uses convolutional neural network (CNN) technology to detect the status of the unmanned factory, sends its information through the Raspberry PI fog computing controller (fog layer) conversion and uploads it to the cloud.
- Step 3: The Raspberry PI connects to the PLC through an analogue and digital signal converter, reads data from the I/O port of the PLC, and sends it to the server through an HTTP request.
- Step 4: Raspberry PI reads these values and wraps them into HTTP requests with a machine ID and timestamp before sending them to the cloud server. To obtain real-time information, the interval of the Raspberry PI loop is adjusted to 100 ms.
- Step 5: When the server receives a request from the Raspberry PI, it stores the received data, machine ID, and timestamp in a database. Authorised users can view the information collected in a secure network environment using an Internet-connected device.
- Step 6: The cloud controller was a dual-core MT7697 (using Cortex-M4 with a FPU Max frequency @192 MHz, with a UART/I2C/SPI/I2S/PWM/ADC/IrDA function). CNN technology passes a video signal, and a PLC intercept signal is used for terminal monitoring. A general view of the proposed system architecture is shown in Figure 2.

3.1. Realization of Smart Factory Based on PLC Production Line

The design concept of the smart factory in the production line using an automated PLC monitoring system is shown in Figure 3. The main module has many integrated systems and the smart factory has a high degree of complexity. It involves a combination of connected systems, automation, the IoT, and cloud computing. This is a key module of a smart factory, which is distinguished by the following items.

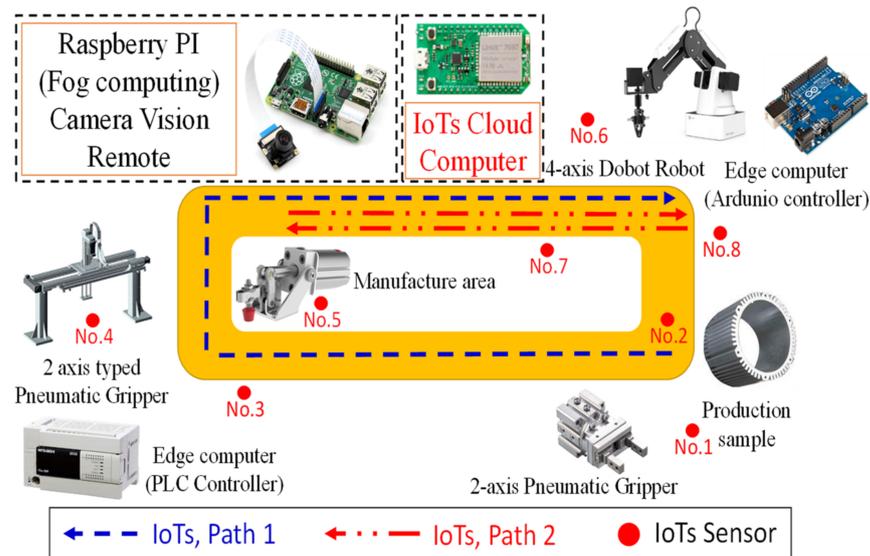


Figure 3. Schematic diagram of the physical architecture of the automated production line.

3.2. Smart-Factory with Data Security and Models

The smart-factory architecture must meet several requirements: confidentiality, integrity, and availability [21–25]. Device confidentiality ensures that only authorized users can read the emails. The integrity message ensures that the received and sent messages are unchanged. The availability of the device is for every available data service. The basic model architecture properties are defined as follows:

$$\begin{aligned}
 S &= \{S_1, S_2, S_3, \dots, S_n\}, \\
 O &= \{O_1, O_2, O_3, \dots, O_n\}, \\
 \mu &= \{M_1, M_2, M_3, \dots, M_n\}, \\
 A &= \{w, r, c\}, \\
 p &= \{l_1, l_2\},
 \end{aligned}
 \tag{1}$$

where S is the theme set; O is the set of objects; μ is the set of access matrices, which represents the subject’s access privileges to the object; A is a set of access attributes, w represents storage, r represents read, and c represents control; and p represents different privilege levels, where $l_1 < l_2$.

The architecture contains three entities: device nodes, management center, and user nodes. Device nodes and management hubs belong to l_2 ; and user nodes belong to l_1 . Data can flow from device nodes and management hubs to user nodes. The user node has no right to write or modify data. However, both the device node and the management center provide all each other’s permissions. This is an effective data interaction realization. The formula defines the determination of the current state as a safety condition [21–25].

$$V = S \times O \times A \times \mu \times p
 \tag{2}$$

where $S \times O \times A$ indicates that the object uses a certain method to access the object [24–28]. Once all elements are safe and trusted, a safe state can be ensured.

3.3. Implementation of Smart-Production Configuration in the Manufacturing Plant

For a smart factory, the architecture system is extended from the IoT. This section de-scribes the design architecture during the operation of the processed object between the processes of data interaction in the production line. Therefore, the state of the processed object must be monitored. This study provides specific electromechanical integration experimental equipment and image-recognition algorithms to validate the experimental results, as shown in Figure 3. First, the proposed architecture implements data collection,

which depends on the PLC using the edge/sensing-computing method. Normally, the PLC arranges one or more sensors and connects them to the hub management center. After receiving the data, fog computing is directly uploaded to the cloud computing system to register the ID, which is added to the whitelist of the connected management center. During the operation of a production line, management is obtained based on the essence of the IoT. According to the concept of IoT, the production line establishes edge computing for basic layer sensor data collection, such as infrared or electromagnetic sensors, and is used to control the production line scheduling signals. The fog-computing controller was applied to the CNN to analyze the signals. Based on IoT cloud system, the signal can be referred to as the basic requirement for the end-user needs [26–28].

The original production line can auto-change to a new production line when faulty equipment is on standby waiting for maintenance work. Meanwhile, the sensing (edge) computer is reconfigured with nearby production lines to reconstruct a new production capacity. To achieve continuous operation, the production capacity should be maintained until the production line breaks back to the system, as shown in the flowchart in Figure 3. The configuration of the automated production line for IoT method is presented in Table 1.

Table 1. IoT method for production line rearrangement.

IoT Interaction for Smart Factory	
1	begin
2	for $i \leftarrow 1$ to $mComputer[1 \dots a]$
3	find the connected $mComputer[j]$ for
4	$mComputer[i]$
5	register ID
6	end for
7	Wait //wait for application
8	if RequestReceived == true)
9	if compare the $mComputer$ with
10	whitelist[1 . . . a] == true)
11	if execute production work == true)
12	start the production manufacturing work and record subsequent data
13	to upload the data to the database directly, including:
14	Manufacture procedure optimization
15	Smart factory with face ID of CNN method
16	Edge-, fog-, and cloud-computing data communication
17	Else
18	Detect the whether the production status is running or break; if a break occurs, then arrange
19	other production line to continuous fabrication and upload the situation to the cloud
20	computer and record
21	end if
22	if Repair production can back to work == true); Run origin setting
23	end if
24	else
25	deny, wait for new production block to be
26	generated
27	end if
28	end if
29	end

Neural Algorithm Image-Recognition Application

A CNN is a feedforward neural network. Artificial neurons respond to some of the surrounding units in the coverage area. It exhibits excellent performance in large-scale image processing. The CNN is composed of one or more convolutional layers with the top fully connected layer (corresponding to the classic neural network). It also includes the associated weights and pooling layer, as shown in Figure 4. This structure allows CNNs to use the 2D structure of the input data. Compared with other deep learning structures, CNNs can provide better results in image and speech recognition. This model can also be

trained using back-propagation algorithms. Thus, for other methods, feed-forward neural networks and CNNs are necessary to consider certain parameters. The CNN yielded an attractive deep-learning structure in this study. Some related structures are listed below.

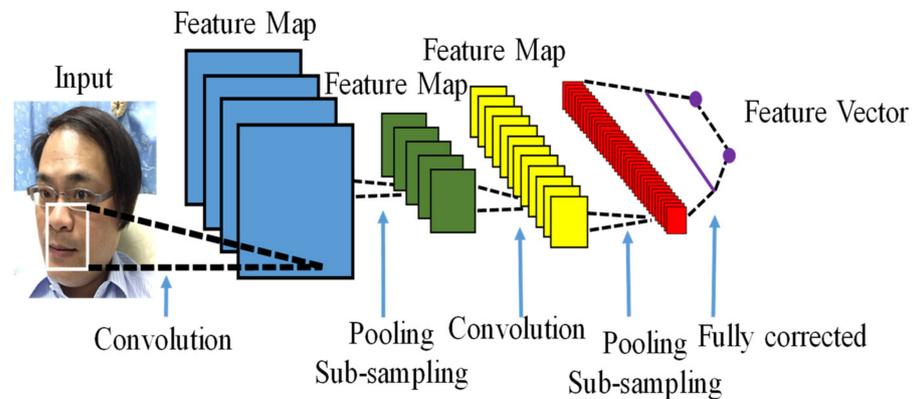


Figure 4. The proposed architecture for CNN algorithm.

- Convolutional layer.

By using a small kernel, the entire image and the intermediate feature map can be convolved. This step allows the kernel to learn the functions in the image. The convolutional layer is a set of parallel feature maps that are composed of different sliding convolution kernels on the input image and perform certain operations. In addition, an element-corresponding product and summation operation at each sliding position are performed between the convolution kernel and the input image, which sets the project information in the receptive field to an element in the feature map.

- Pooling layer.

This layer typically follows one or several convolutional layers to reduce the size of the feature map and the risk of overfitting. The pooling layer is another important concept in CNNs, which is a nonlinear form of down sampling. There are many different forms of nonlinear pooling functions, and max pooling is the most common. It divides the input image into several rectangular regions and outputs the maximum value for each sub region.

- Fully connected layer.

The final connected layer uses one or several convolution and pooling layers, and the 2D feature maps are converted into 1D vectors for classification or other processes. After several convolution and max-pooling layers, advanced reasoning in the neural network is completed by the fully connected layers.

This is like a conventional non-convolutional artificial neural network, and the neurons in the fully connected layer are connected to all activations in the previous layer. Therefore, their activation could be calculated as an affine transformation. The result was first calculated by multiplying it by a matrix. Subsequently, a bias offset was added to shift the values.

- Total connection layer.

For our application concept, CNN (machine learning) is widely used as a common image-recognition system. The image-recognition problem is much more difficult in video analysis than in still images. CNNs are often used to solve such problems. There is also natural language processing, for which CNNs are often used. The CNN model has been proven to be able to effectively deal with various natural-language-processing problems, such as semantic analysis, search result extraction, and sentence modelling.

- CNN algorithm.

Image recognition is highly challenging in smart factories. A high recognition performance is proposed, which is faced with the effects of changes in non-linear parameters, such as changing illumination, posture, facial expressions, and occlusion. This study proposes a four-layer convolutional neural network architecture. The CNN is proposed as a solution to this problem which includes face and morphological recognition of production artefacts of a smart factory. This study establishes an applicable method which has more than 1000 image recognition problems. A partial connection was introduced between the first two layers to ensure that different functions were learned during the training process. The results of the proposed CNN method. The general method requires two stages to perform convolution and subsampling, whereas the fusion method requires only one stage.

The artificial neural network training process was as follows. The artificial neural network model is a multi-layer perceptron (MLP), in which a feedforward neural network is used to map the input dataset to a set of appropriate outputs. MLP is characterized by the LC2 layer of neurons (input layer, hidden layer L, and output layer), which has a non-linear activation function at the hidden layer unit. To describe the non-linear relationship between different impact factors and reflectivity, feedforward MLP is used for the impact factor (x) into a single predicted value y . In the MLP, the in-put layer is composed of the 1D vector of the region of interest in the original depth image and the authenticity image on which they are based. The characteristic of the hidden layer is that the hidden neuron has an integer linear unit function, and the output layer is composed of only one output neuron (non-linear value y). The number of hidden neurons is determined by a simple method of trial and error. The input variable vector x is mapped to the neurons in the hidden layer, as shown below:

$$\begin{cases} h_i = ReLU(W_i \times x + b_i), (i = 1) \\ h_i = ReLU(W_i \cdot h_{i-1} + b_i), (i = 2, 3 \dots, L) \end{cases} \quad (3)$$

where h_i is the output value of the i -th layer, L is the number of hidden layers, W_i is the weight matrix between the previous layer $i - 1$ and the current layer i , and b_i is the bias parameter vector of the previous layer $i - 1$ and the current layer i . Additionally, the y value represents the distance obtained from the depth image, which is also the output of each sample, which is obtained from the linear combination of the hidden neuron vector h_i , as follows:

$$y = f(x; W) = h_L(h_{L-1}(\dots h_2(h_1(x; w_1); w_2); \dots; w_{L-1}); w_L) \quad (4)$$

The final cost function can be calculated as follows:

$$\text{Loss} = \frac{1}{n} \sum_{j=1}^n [y_j - y_j^{gt}]^2 = \frac{1}{n} \sum_{j=1}^n [f(x_j; W) - y_j^{gt}]^2 \quad (5)$$

where loss function is a training set, validation set, or testing set, n is the number of samples, y_j is the output value of sample j , and y_j^{gt} is the true value of sample j (ground truth). The above formula shows the average error between the predicted value and the true value. Average-Error (AE) is used to adjust the model.

$$AE = \sqrt{\text{Loss}} \quad (6)$$

3.4. Smart Robot Control and Visual Recognition

This research is based on CNN theory. First, the MediaTek Company constructed the Linkit-7697 dual-core chip as an IoT information integration device. They also built a MediaTek Cloud Sandbox (MCS) as a cloud system, as shown in Figure 5. It is a cloud data-service platform that allows users to connect to IoT. This device uses MCS to quickly realize the prototype IoT product. Furthermore, the programming function database for machine vision through OpenCV software was obtained, including the users of OpenCV and Python/C++ to detect faces in the images. The Haar-based cascade classifier in

OpenCV was used to realize facial recognition in the code. This is an effective object-detection method. This method is a CNN algorithm in which the cascade function is trained using numerous positive and negative images. Thus, the images can be detected. OpenCV contains many retrained classifiers, such as faces, eyes, and smiles.

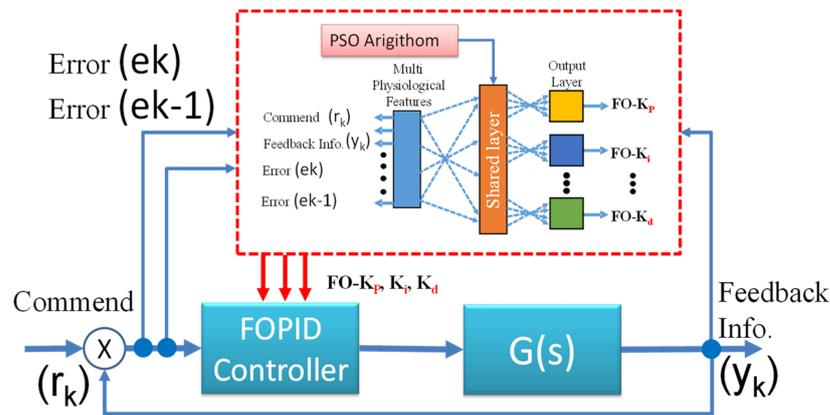


Figure 5. The neural network’s design and structure for motor control.

3.4.1. Robot Motor Control with FOPID

This study aimed to establish camera vision with a neural algorithm for image recognition, combined with the joint module of the robotic arm, to control the motor with a power feedback method. This method included two sequential processes. First, the CNN was used for image recognition to apply the output control of the signal. Second, it estimated the external torque of each joint τ and used the feedback signal for the identification and detection processes. In addition, an AI method was constructed to control the motor. The position and speed control method was based on the detection results after deep learning. To evaluate the performance of self-detection, each joint of the motor was regulated by an external force. Two performance indicators were introduced to represent safety and efficiency. The control signal was sent to the cloud system for valuable collection and observation. The design and structure of the neural network used for motor control are shown in Figure 5. Momentum-based observers were therefore widely used to estimate the torque of the external joints (the joint refers to the motor signal being measured) using the kinetic energy method. For an n degrees-of-freedom rigid-body, the dynamic equations are as follows:

$$M(q)\ddot{q} + C(q, \dot{q})\dot{q} + g(q) = \tau_m + \tau_{external} \tag{7}$$

where $M(q) \in R^{n \times n}$ is the inertia matrix, $C(q, \dot{q})\dot{q} \in R^n$ is the vector of Coriolis and centripetal torque, and $g(q) \in R$ is the gravitational vector, τ_m is the control torque, $\tau_{external}$ is the external torque. The position control applied to the model is expressed as follows:

$$\tau_m = \hat{M}(q)\ddot{q}_r + \hat{C}(q, \dot{q})\dot{q}_r + \hat{g}(q) + \tau_{reference} \tag{8}$$

The parameters of the \hat{M} , \hat{C} and \hat{g} are marked as nominal model, $\dot{q}_r = \dot{q} + K_p + K_I \int edt$ represents the reference trajectory, $e = q_d - q$ is the position error, and q_d represents the required trajectory, and $\tau_{reference}$ is the control input, which is defined as follows:

$$\tau_{reference} = -\left(K + \frac{1}{r^2}\right)\left(\dot{e} + K_{pe} + K_i \int edt\right) \tag{9}$$

where $r > 0$, and K, K_p and $K_i > 0$ in the PID controller are symmetric gain matrices and satisfy $K_p^2 > 2K_i$. The following observers are designed to evaluate the external torque as follows:

$$\hat{\tau}_{external} = L\left(-p_e(t) + p_e(0) + \int_0^t (s_e(t) - \tau_{ref.}) ds\right) \tag{10}$$

where $p_e = \hat{M}(q)\dot{e}_r$, $s_e = -\tau_{reference} + \hat{C}(q, \dot{q})\dot{e}_r$ and $\dot{e}_r = \dot{q}_r - \dot{q}$, so the dynamic equation of the observer is rewritten as

$$\hat{\tau}_{external} = L \left(-\hat{\tau}_{external} + \tau_{external} \right) - L \left(\tilde{M}\ddot{q} + \tilde{C}\dot{q} + \tilde{g} \right) \tag{11}$$

Among them, $\tilde{M} = M - \hat{M}$, $\tilde{C} = C - \hat{C}$, $\tilde{g} = g - \hat{g}$ and $\hat{\tau}_{external}$ refers to the sum of $\tau_{external}$ and τ_d . This is the recognition program constructed by the XML model, as shown in Table 2. Then, the neural network as image recognition results can be applied to the motor control of the robotic manipulator.

Table 2. Intelligent robot manipulator motor control method with object recognition for light-out factory control.

Face and Color Detection by Artificial Intelligent Method	
1	Input: Linear Velocity and Steering Angle:
2	$x_1, x_2, x_3, \dots, x_N. R_n \times n$
3	Control: Prediction of angle, torque, and direction: Y
4	Measure parameters: motor current $i(A)$, voltage (V), angle, and torque τ_m
5	Given parameters: NN method W weight matrix and b bias vector
6	Parameter Initialization of M, C, g, τ_n
7	while t > 0 do
8	Calculating $\tau_{external}$ using Equation (7)
9	Calculating position τ_m using Equation (8)
10	Calculating FOPID trajectory using Equation (9)
11	Calculating external torque $\hat{\tau}_{external}$ using Equation (10)
16	End
17	Output: motor=[current, voltage, torque, angle]
18	Y = Fully Connected to PLC controller for smart factory machine
19	Result: Prediction of current, voltage, and torque

This follows the design of a basic proportional-integral-derivative (PID) controller that is used to perform the work piece clamping work of the motor on the production line. This part can replace a manual configuration. At the same time, the visual monitoring of a lights-out smart factory can greatly improve the efficiency and reduce the total cost of the factory in terms of the workforce, and meet faster production delivery requirements. Therefore, to load the image for face detection, it must be performed as before.

First, the image must be converted into a grey image. Face detection was performed using the detectMultiscale() function of the cascade classifier: in this case, face_cascade.detectMultiScale(gray, 1.3, 5), where the scale factor (SF) is 1.3, and parameter 5 is the minNeighbors parameter. Using the SF function before and after the scan process, the search for the scale factor of the window can be done within two successive scans. The set parameter of 1.3 is expanded by a search window of 30%. In addition, for the minNeighbors function, the minimum number of adjacent rectangles can detect the target. If the number of small rectangles used to detect the target is less than that of min-Neighbors, it is excluded. After this calculation, face shape detection was completed. All face data were found in the image and stored in the system. A rectangular shape was drawn around the face. Thus, the face-recognition function was successfully performed.

3.4.2. Robotic Arm Control for Edge Computing Industrial Sensing Intercept Point

The working field node of the production line is a key factor in IoT structure. To observe the dynamic results, if the system detects the failure condition of the production line, it will provide information to the user or third-party system of the service endpoint to address production-line problems. Figure 6 illustrates the physical system used to implement the industrial network. This is the main component of technology. The system embeds the CNN model in a pre-set program. First, the physical sensing layer contains an actual measurement intercept point that is continuously collected. Industrial equipment

and systems are required. The software and hardware of the embedded system (MediaTek 7697 and Raspberry PI 4S) were developed externally. Therefore, the communication status between the physical environment and the on-site network environment is determined. Second, the fog layer contains the technical components required to accept the input information of the physical layer, which executes and returns the analysis results. Finally, the cloud layer of back-end users includes maintenance-related factories that need to deploy and receive network information for each atomization layer.

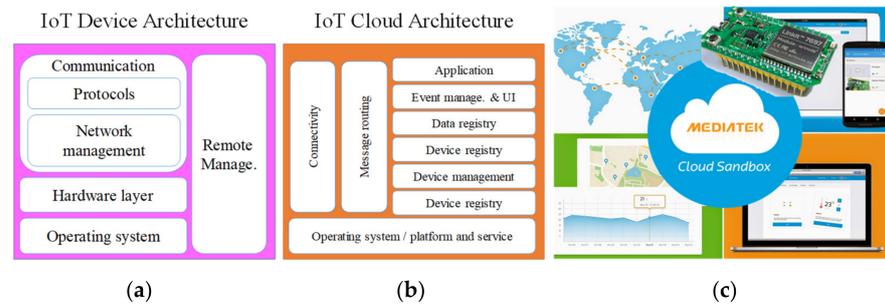


Figure 6. The sensor extension to the IoT: (a) IoT, (b) cloud, (c) Smart-chip connect to cloud.

In this study, a gripping robot with a control process is used. It remotely collects physical visual image data continuously day and night. This includes multi-sensor signal collections uploaded through the cloud-embedded system that connects to a single-chip component combined with IoT architecture platforms for signal interception and control functions, as shown in Figure 6. At present, the LinkIt series of development boards are aimed at IoT applications and mainly provide two series of development boards: LinkIt Smart 7688 and 7697 Duo, which are used for higher-level IoT nodes or gateways.

This can be performed using Linux kits with high-level languages (Python/Java script). The Linkit-7697 was positioned as a lighter node. For the application of the robot, rotation motors and linear actuators with multi-sensing signal processing were constructed. It can be shown that smart linear actuators and a wisdom factory robotic was built. The networking architecture of IoT in the device and cloud is shown in Figure 7a,b. The components of the cloud-embedded system are shown in Figure 7c. The experimental structure is shown as a diagram of a robotic arm combined with visual images to control an AI motor. Based on the above-mentioned CNN theory, facial images, objective displacement motion tracking, and color discrimination are used to control the AI motor actions, as shown in Table 3. It is an AI motor trend and control program based on the CNN theory.

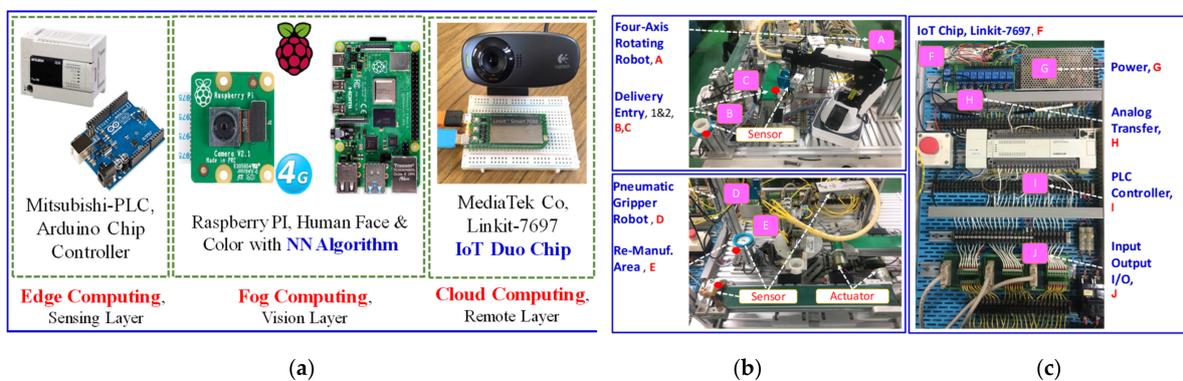


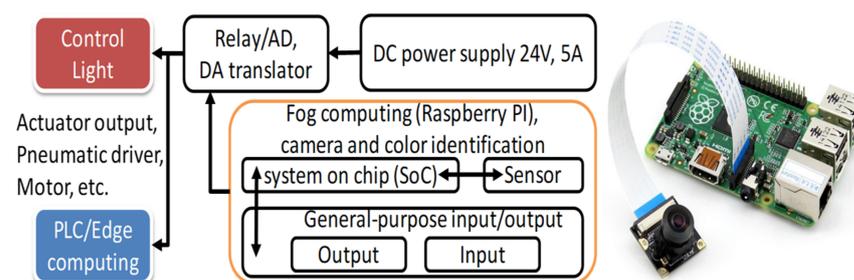
Figure 7. The platform of a physical construction for IoT of the smart factory: (a) edge, fog, and cloud controller entities; (b) gripper-type pneumatic power arm and four-dimensional robotic arm; (c) power and controller architecture.

Table 3. CNN recognition control robot AI motor.

Robot Motor Control by CNN Method	
1	Import cascade calculator
2	Input identification image
3	Conversion image to grayscale
4	Detecting the faces
5	Detect image multi scale
6	Minimize image neighbor
7	Minimize image size
8	Plot the face position shape
9	for (x, y, w, h) in face
10	Plot rectangle shape for face
11	end
12	Results:
13	Showing normal windows size
14	Show up image
15	Export image to NN parameters
16	Control FOPID motor parameters
17	Waiting next instruction
18	Close windows

3.4.3. Four-Axis Robotic Arm

The application of a PLC-gripping lightweight robot with soft material to regulate a suitable gripping force was designed, which can effectively avoid surface damage to the production parts and protect the quality of the products. In this study, the entity diagram of the system architecture is a human-machine interface, as shown in Figure 8. A module experiment was developed for the controller. After upgrading the different module signals to the control system, the experiment was continuously conducted to capture the object and adjust the air pressure and gripping force. The visual entity-monitoring lens intercepted the day and night images and uploaded them to the cloud server platform. The programmable controller was expected to be able to formulate a variety of shapes and sizes of objects to clamp for rapid production line manufacturing. Some important component interfaces are as follows:

**Figure 8.** The Raspberry Pi circuit connected to AD signal structure; 4.5 Cloud high-end system.

- Soft-touch gripper design: To establish different parameter modules of the human-machine interface, the controller parameters can be quickly adjusted.
- Gripping system design: The combination model includes a soft-touch gripper, PLC, compressed gas, and industrial gripper robot.
- Drive control interface: Standard input-output communication terminals and fast transmission program interface signal ports are applied.
- Remote visual image-monitoring lens: The visual image is sent to the cloud server through the network platform to provide users with remote observations of the factory.

3.5. Edge, Fog, and Cloud Computing Device

3.5.1. Fog System and Edge-Computing Integration

This study proposed a composition of the topology of a fog-computing embedded system. This topology is based on embedded machine learning. This was used for the physical structure of the network to interactively realize embedded machine learning. Cloud platform storage is suitable for different engineering applications, including established prediction values of the equipment, system prediction results, and processing methods.

To establish production machine learning models, these models are distributed and executed safely through fog nodes in the local network domain. In each fog node, its identity and engineering applications are linked to the cloud platform in an uploaded and synchronized manner to ensure new models and values. Once the machine-learning model is changed, it can continuously track the actual intercept point value without relying on external connections and signal services, particularly for user-operation messages and reminders.

A. Raspberry PI application of fog system.

For the fog controller, choosing a single chip, such as an Arduino computer, to perform as the fog (middle) layer of the network is not suitable because it might limit the performance of the power and storage of the computer. In this study, Raspberry PI 4 Model B was used in the fog system. This device processor was a 1.5-GHz BCM2711 (quad-core Cor-tex-A72), the storage memory capacity was increased to 4 GB, and it had Bluetooth 5.0, with four USB 2.0/3.0 interfaces, HDMI vision interfaces, and a power supply with USB-C interface. These specifications enabled the portability of this device across the entire supply chain of the factory. However, the original technical decision-making strategy may have an impact on portability. Fortunately, cloudlets or microcomputer cluster methods can also be used. It can be faster to solve engineering problems and special requirements for a larger computer stream source are not then needed. This research uses a Raspberry PI embedded system for video streaming. First, the test was carried out using a camera lens and pictures/videos. Image streaming should then be performed to connect to internet services. Finally, OpenCV was used to process the images and build a hardware development platform. The basic image technology of this system is as follows:

- Image transformation: color space and basic image processing; color space (RGB, YUV, HSV).
- Affine transformation includes translation, rotation, scaling, and cropping. Image-processing methods include (a) blur, (b) erosion, and (c) dilation.
- Optimal edge path: Canny edge detection, Hough transform, and moment invariance finish the contour.
- For image applications, except for the above-mentioned color recognition function, a face-detection (Haar classifier) method was developed as a simulate-type lights-out smart factory constructed with an application of the machine learning method. Based on its image-recognition system, Raspberry PI was used with the fog system to create a smart factory for this study.

B. Fog computing and node control.

A suitable wireless node controller needs to be developed that is used to intercept the face recognition of the neural algorithm and the color recognition of the processed product to detect product targets on the production line.

An algorithm, for example, for detecting a signal can be used to detect abnormalities in a factory. The image detector is shown in Figure 8. The embedded system of the Raspberry PI is defined as a fog-computing controller, and the general input and output of the node are extended and connected as a relay to control the terminal node to transmit the status of the control light. The output was linked to the PLC. The main purpose was to immediately stop the operation of the production line until no dangerous elimination was expected, at which point the production line could be restarted.

3.5.2. Cloud High-End System

A high-level IoT platform can extract data from multiple types, such as application procedures, sensors, and equipment. It conducts complexation analysis through connectors, including assuming analysis procedures, and can provide more valuable data. This is because the numerical value is used to improve the efficiency and reduce the cost of the optimization design, as shown in Figure 9. This is a simple diagram of IoT architecture. The foundation of the cloud system is mainly based on IoT chip development by MediaTek. Meanwhile, MediaTek has established a cloud-data service platform, MCS, that can be connected to different IoT devices. This is used in the MCS system to quickly realize IoT prototype. Thus, each test of the prototype device provides multiple definitions of the data channels.

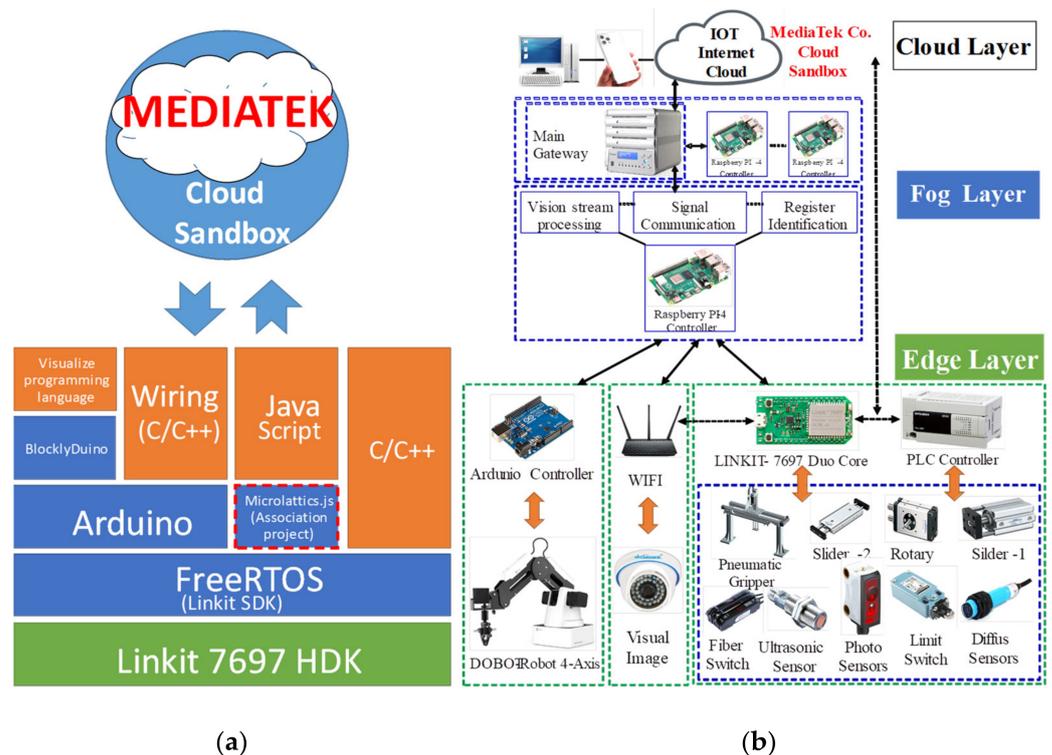


Figure 9. Low-cost IoT architecture: (a) Cloud computing MediaTek Linkit-7697 structure, (b) diagram of complete communication link.

Each test device had an independent device ID and device password (device key) to provide external connections and ID functions. To connect the MCS server, the development device communicates with the MCS server through HTTP Restful APIs and other communication protocols. The MCS library was provided by Linkit-7697 Arduino BSP, encapsulating these communication protocols into a simple and easy-to-use interface, as shown in Figure 9. This allows Arduino developers to connect easily to MCS services. The Arduino BSP of Linkit-7697 provides an MCS library for developers to implement applications using MCS services. The MCS library encapsulates the operation and communication between Linkit-7697 and the MCS server, including: (a) connecting to the specified MCS server; (b) creating a data channel (except for the gamepad controller); (c) specifying the data channel, and the data point is transmitted to the MCS server; (d) the data point of the specified data channel is received from the MCS server; and (e) the current communication protocol is supported by the library. The TCP and HTTP communication links can be connected.

Based on the uncertain nonlinear characteristics of the robot parameters, a robot controller is needed to adjust the motor and follow a predetermined trajectory. The algorithm tuned the gain using the FOPID controller parameters. The robot manipulator was devel-

oped to replace the traditional program of the designer’s expertise. Figure 10 shows the control architecture of the manipulator. In addition, the controller gains need to be properly optimized to achieve better performance. In this paper, the FOPID with a modified neural network algorithm is proposed as a novel adaptive adjustment algorithm to optimize the gain of the controller.

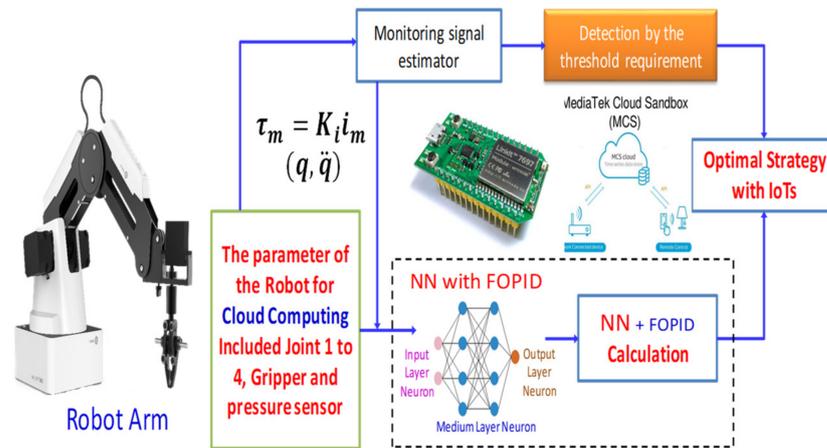


Figure 10. Neural network regulated the fractional order PID controller to feedback the signal of the motor torque, and transmit the upload of each joint datum to the IoT.

4. Results and Discussion

In this study, an IoT solution based on the establishment, installation, testing, and implementation of experiments in the Robot and Motor Control Laboratory at the Asia Eastern University of Science and Technology, Mechanical Engineering (AEUST-ME, Taiwan), was built. Related experiments to create a solution using IoT are proposed to monitor the status of workpieces on the production line. As shown in Figure 11, AEUST proposed a physical map of the electromechanical integration production line platform. The quality of the product is improved through the relationship between the data of various types of sensors and the color recognition of the image.

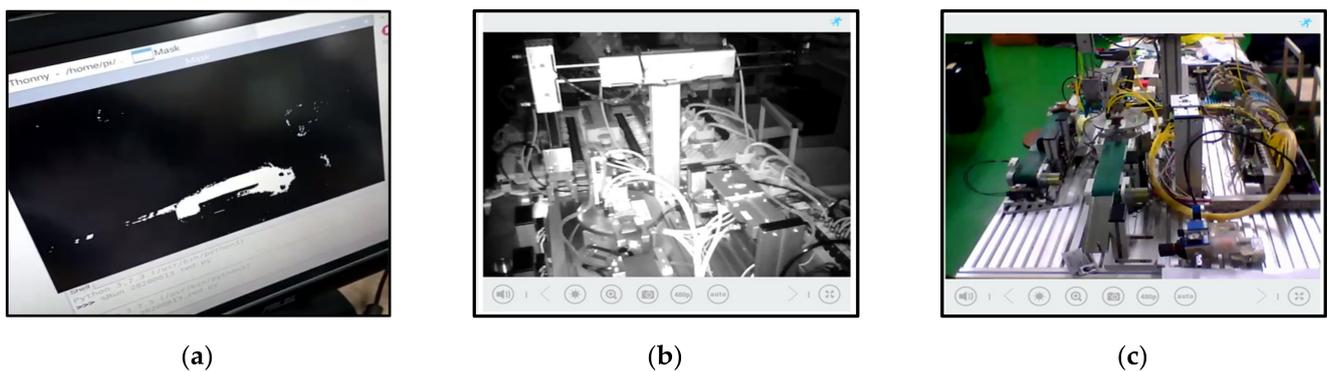


Figure 11. Physical visual image results of the cloud servo platform to remote and monitor: (a) color-recognition image of the production object, (b) night monitoring image: black and white image, (c) day monitoring image: color image.

In the first step of the initial construction, the original machine is based on the traditional PLC architecture, and its speed of operation, product color recognition, self-inspection, and repair or learning mechanisms are lacking. Therefore, the establishment of a new IoT in the signal of the existing network communication interception sensor is integrated, which is unified for analogue/digital signal conversion. To avoid excessive surge signals, the impact of the electronic circuit on the burning out of the electronic components

is clear. Adding a rectifier and limit-function circuit for signal interception can avoid this issue. Thus, the converted signal can be sent to the cloud network control system with assured integrity. Furthermore, the construction and functionality of the PLC equipment and machines are enhanced. A supervised machine learning mechanism was proposed.

This is a general supervised learning process used to generate a global model that corresponds to the input objects and the expected output. Based on the case in this study, the optimized production line transfer mode of similar processes has been solved in the past, and new inference methods are sought to solve the best path of the current production line. This is the process of the machine learning method used to solve current production line problems. Figure 12 shows the face recognition results constructed by applying the neural theory. The Raspberry PI system is detected by fog computing that outputs the recognition signal, which includes the human wear mask and motion-tracking detection. By synchronously uploading data to the cloud computing controller for calculation and monitoring, the functions of factory population monitoring are achieved. During the visualization process to repair a rejected product, pre-shipment inspection is necessary. The system uses real-time location data and shipment urgency to track progress. It can repair and reduce human costs after required additional delivery procedures. In the future, these challenges will include image analysis of production-line workers and equipment during the test and analysis of the relationship between these data and test log data. To use these results, it is necessary to further reduce the scrap rate of finished products. In addition, it is difficult to reduce indirect costs and extend the visualization plan to the entire supply chain and between other factories. To create a solution using an IoT structure, both design and experiments are proposed to monitor the status of workpieces on a smart factory platform. As shown in Figure 13, we propose a physical map of the electromechanical integration platform. The quality of the product is improved through the relationship between the data of various types of fluxgate sensors, color recognition of the image, and acoustic and vibration signals.

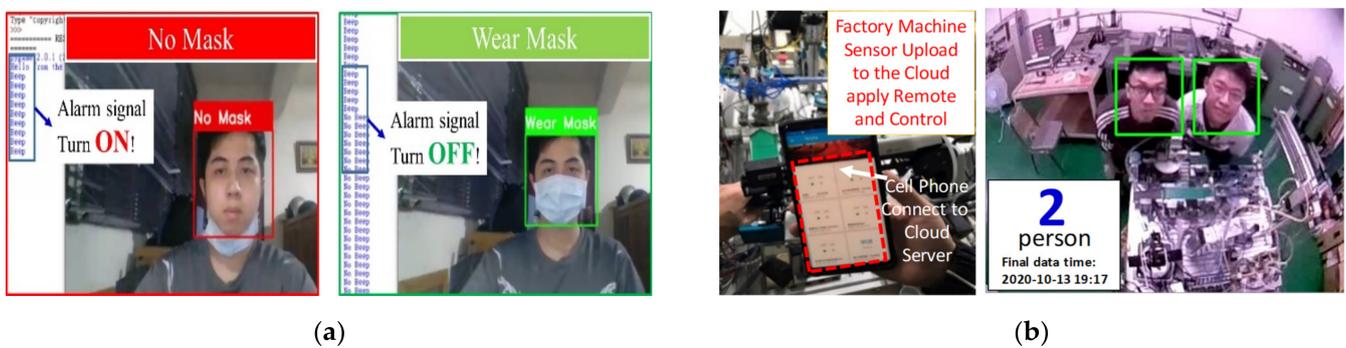


Figure 12. IoT PLC smart-factory-monitoring field platform: (a) physical layer sensing numerical platform, (b) detected signal upload to cloud platform.

The converted signal can be sent to the cloud network control system with assured integrity. Furthermore, the construction and functionality of PLC equipment and machines are enhanced. A supervised machine-learning mechanism was proposed. This is a general supervised learning process used to generate a global model that corresponds to the input objects and expected output, including the flux signal, color vision, and motor-moving trajectory path.

Optimized PLC production line transfer processes have been previously solved, and new inference methods have been sought. These methods can provide a better path for the smart factory of a PLC production line. This is a process of the machine learning method used to solve current PLC production line problems. Figure 14 shows the face recognition results constructed by applying the neural theory. The Raspberry PI system is detected by fog computing, which outputs a recognition signal. We achieved synchronous uploading of the data to the cloud computing controller for the calculation and monitoring functions

of factory population monitoring. In this study, the intelligent robot arm integrates the neural network theory and FOPID controller to evaluate the operating efficiency of the smart factory in terms of magnetic sensor signals, product image recognition, and motor signal control, thereby reducing the total cost by at least 50% and measuring the operation duration of each motor, as shown in Figure 15.

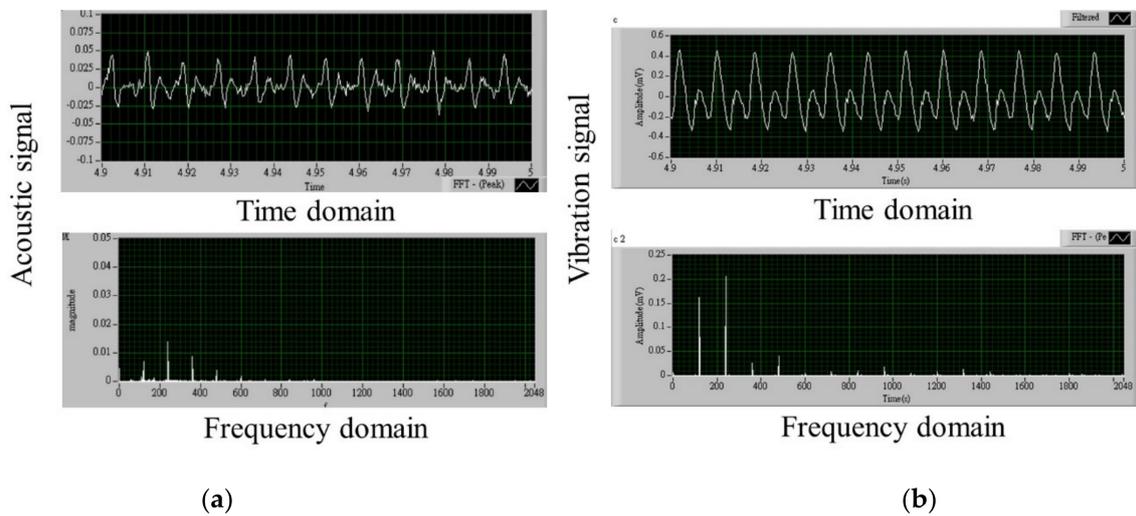


Figure 13. IoT PLC smart-factory-monitoring field in acoustic and vibration with Fog computing platform: (a) acoustic signal and (b) vibration signal.

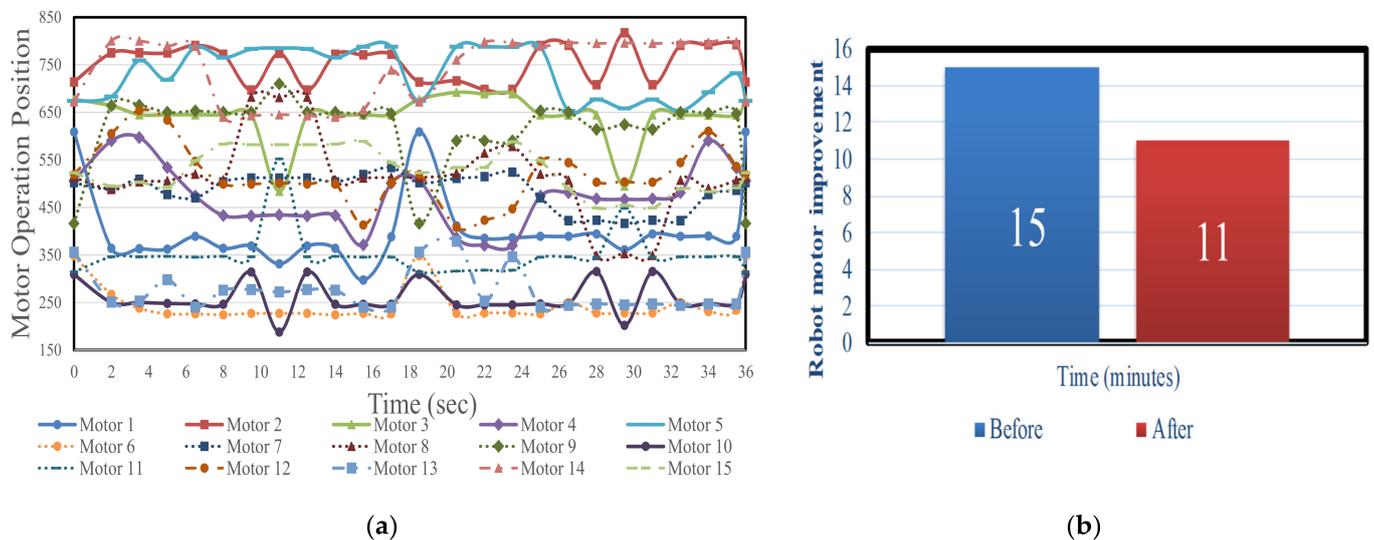


Figure 14. Robot motor control performance by using FOPID+NN algorithm: (a) motor operation duration, (b) operation efficiency.

In this study, the data were collected in the laboratory for more than half a year for numerical analysis and function adjustment. New research results in IoT and image transmission have been realized, which can smoothly achieve higher production efficiency and high quality in a shorter time. At the same time, the installation of the device has a multifunctional and responsive man-machine-coordinated production system. Figure 15 shows the results of these studies. The lead-time for manufacturing was reduced by 50%. In addition, the product can be visualized, and the elimination of defects helps to reduce defective products by 73% in the production of workpieces, which reduces the total cost of factory work, and enables a faster CNN method to improve the production delivery, respectively. These improvements have helped reduce the production area and inventory

by at least 50%. This research has some relative benefits: for example, it can reduce the material storage space of upstream manufacturers. It can also use the excess space in the factory for service, business, and production line research and development.

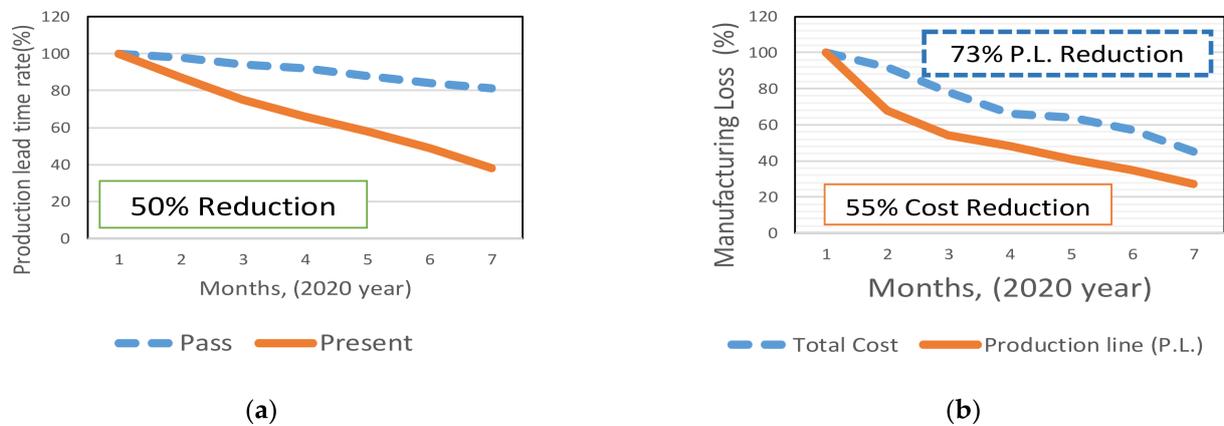


Figure 15. Statistics of benefits and results brought by the IoT in smart factories: (a) Product lead time, including design, production, logistics and sales process, reduced time, (b) product manufacturing process and assembly, its failed product components, and reduction result.

5. Conclusions

This research successfully implemented a low-cost small smart factory with CNN, IoT, and a cloud servo-control experimental system. The core architecture is image recognition technology and a neural network system, namely, machine learning for face recognition in lights-out smart factories. The research results included the implementation of machine learning rules in the production line fog-computing image-monitoring system to establish the edge-computing sensing signal architecture and synchronously receive the edge- and fog-computing device modules in the cloud system. We verified the image recognition of the CNN and the robot motor signal control of the FOPID. This study proposes that a CNN + FOPID method can improve the efficiency of the factory by more than 50% compared with traditional manual operators. The low-cost, high-efficiency equipment of the new method has substantial contribution and application potential.

Author Contributions: Conceptualization, C.-H.H. and S.-J.C.; methodology, C.-H.H. and S.-J.C.; software, C.-H.H. and C.-P.F.; resources, C.-H.H.; writing—original draft preparation, Y.-M.H.; writing—review and editing, C.-H.H. and C.-P.F.; writing—review & editing; Methodology; validation, T.-J.C.; project administration, C.-P.F. and S.-F.C. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Data are contained within the article.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Siriwardhana, Y.; de Alwis, C.; Gür, G.; Ylianttila, M.; Liyanage, M. The Fight Against the COVID-19 Pandemic with 5G Technologies. *IEEE Eng. Manag. Rev.* **2020**, *48*, 72–84. [[CrossRef](#)]
2. Fatima, Z.; Tanveer, M.H.; Zardari, S.; Naz, L.F.; Khadim, H.; Ahmed, N.; Tahir, M. Production Plant and Warehouse Automation with IoT and Industry 5.0. *Appl. Sci.* **2022**, *12*, 2053. [[CrossRef](#)]
3. Pierucci, L. Hybrid Direction of Arrival Precoding for Multiple Unmanned Aerial Vehicles Aided Non-Orthogonal Multiple Access in 6G Networks. *Appl. Sci.* **2022**, *12*, 895. [[CrossRef](#)]

4. Chang, R.I.; Lee, C.Y.; Hung, Y.H. Cloud-Based Analytics Module for Predictive Maintenance of the Textile Manufacturing Process. *Appl. Sci.* **2022**, *11*, 9945. [[CrossRef](#)]
5. Cassagnau, P.; Bounor-Legaré, V.; Vergnes, B. Experimental and Modelling Aspects of the Reactive Extrusion Process. *Mech. Ind.* **2019**, *20*, 803. [[CrossRef](#)]
6. Tao, F.; Qi, Q. New IT driven service-oriented smart manufacturing: Framework and characteristics. *IEEE Trans. Syst. Man Cybern. Syst.* **2019**, *49*, 81–91. [[CrossRef](#)]
7. Alrubei, S.M.; Ball, E.; Rigelsford, J.M. A Secure Blockchain Platform for Supporting AI-Enabled IoT Applications at the Edge Layer. *IEEE Access* **2022**, *10*, 18583–18593. [[CrossRef](#)]
8. Zhu, L.; Li, P.; Shen, G.; Liu, Z. A Novel Service Composition Algorithm for Cloud-Based Manufacturing Environment. *IEEE Access* **2020**, *8*, 18583–18593. [[CrossRef](#)]
9. Feng, J.; Li, F.; Xu, C.; Zhong, R.Y. Data-Driven Analysis for RFID-Enabled Smart Factory: A Case Study. *IEEE Trans. Syst. Man Cybern. Syst.* **2020**, *50*, 81–88. [[CrossRef](#)]
10. Chuang, L.; Lee, Y.; Yao, F. Intelligent Machinery Product Service Blueprint Development and Verification: An Empirical Study of Machine Tool Industry. *IEEE Access* **2022**, *16*, 19796–19811. [[CrossRef](#)]
11. Yu, W.; Liu, Y.; Dillon, T.; Rahayu, W.; Mostafa, F. An Integrated Framework for Health State Monitoring in a Smart Factory Employing IoT and Big Data Techniques. *IEEE Internet Things J.* **2022**, *9*, 2443–2454. [[CrossRef](#)]
12. My, C.A. The Role of Big Data Analytics and AI in Smart Manufacturing: An Overview. *Research in Intelligent and Computing in Engineering*. In *Research in Intelligent and Computing in Engineering; Advances in Intelligent Systems and Computing*; Springer Nature: New York, NY, USA, 2021; Volume 1254.
13. Anh-My, C. Effective Solution to Integrate and Control a Heavy Robot Driven by Hydraulic Actuators. In *Further Advances in Internet of Things in Biomedical and Cyber Physical Systems*; Springer Nature: New York, NY, USA, 2021; Volume 193.
14. Chu, A.M.; Nguyen, C.D.; Vu, M.H.; Duong, X.B.; Nguyen, T.A.; Le, C.H. Kinematic and Dynamic Modelling for a Class of Hybrid Robots Composed of m Local Closed-Loop Linkages Appended to an n-Link Serial Manipulator. *Appl. Sci.* **2020**, *10*, 2567. [[CrossRef](#)]
15. Ye, X.; Hong, S.H.; Song, W.S.; Kim, Y.C.; Zhang, X. An Industry 4.0 Asset Administration Shell-Enabled Digital Solution for Robot-Based Manufacturing Systems. *IEEE Access* **2022**, *9*, 154448–154459. [[CrossRef](#)]
16. Ahmed, I.; Jeon, G.; Piccialli, F. A Deep-Learning-Based Smart Healthcare System for Patient’s Discomfort Detection at the Edge of Internet of Things. *IEEE Internet Things J.* **2021**, *8*, 10318–10326. [[CrossRef](#)]
17. Savaglio, C.; Ganzha, M.; Paprzycki, M.; Badic, C.; Ivanovic, M.; Fortino, G. Agent-based Internet of Things: State-of-the-art and Research Challenges. *Future Gener. Comput. Syst.* **2020**, *102*, 1038–1053. [[CrossRef](#)]
18. Alam, M.G.R.; Hassan, M.M.; Uddin, M.Z.; Almogren, A.; Fortino, G. Autonomic Computation Offloading in Mobile Edge for IoT Applications. *Future Gener. Comput. Syst.* **2019**, *90*, 149–157. [[CrossRef](#)]
19. Piccialli, F.; Casolla, G.; Cuomo, S.; Giampaolo, F.; di Cola, V.S. Decision Making in IoT Environment through Unsupervised Learning. *IEEE Intell. Syst.* **2020**, *35*, 27–35. [[CrossRef](#)]
20. Gandhi, V.; Joo, Y.H. T-S Fuzzy Sampled-Data Control for Nonlinear Systems with Actuator Faults and Its Application to Wind Energy System. *IEEE Trans. Fuzzy Syst.* **2022**, *30*, 462–474. [[CrossRef](#)]
21. Wan, J.; Li, J.; Imran, M.; Li, D. A Blockchain-Based Solution for Enhancing Security and Privacy in Smart Factory. *IEEE Trans. Ind. Inform.* **2019**, *15*, 3652–3660. [[CrossRef](#)]
22. Lee, J.S.; Cho, I.S. Extracting the Maritime Traffic Route in Korea Based on Probabilistic Approach Using Automatic Identification System Big Data. *Appl. Sci.* **2022**, *12*, 635. [[CrossRef](#)]
23. Wan, J.; Tang, S.; Li, D.; Wang, S.; Liu, C.; Abbas, H.; Vasilakos, A.V. A Manufacturing Big Data Solution for Active Preventive Maintenance. *IEEE Trans. Ind. Inform.* **2017**, *13*, 2039–2047. [[CrossRef](#)]
24. Park, Y.-S.; Yoo, D.-Y.; Lee, J.-W. Programmable Motion-Fault Detection for a Collaborative Robot. *IEEE Access* **2021**, *9*, 133123–133142. [[CrossRef](#)]
25. Liu, Y.; Yu, W.; Dillon, T.; Rahayu, W.; Li, M. Empowering IoT Predictive Maintenance Solutions with AI: A Distributed System for Manufacturing Plant-Wide Monitoring. *IEEE Trans. Ind. Inform.* **2022**, *18*, 1345–1354. [[CrossRef](#)]
26. Colombo-Mendoza, L.O.; Paredes-Valverde, M.A.; Salas-Zárate, M.D.; Valencia-García, R. Internet of Things-Driven Data Mining for Smart Crop Production Prediction in the Peasant Farming Domain. *Appl. Sci.* **2022**, *12*, 1940. [[CrossRef](#)]
27. Knittel, D.; Makich, H.; Nouari, M. Milling Diagnosis using Artificial Intelligence Approaches. *Mech. Ind.* **2019**, *20*, 809. [[CrossRef](#)]
28. Zhang, H.; Wang, H.; Li, J.; Gao, H. A Generic Data Analytics System for Manufacturing Production. *Big Data Min. Anal.* **2018**, *1*, 160–171.