Developing a Digital Twin and Augmented Reality Monitoring System with Consideration of Cybersecurity

Keng-Hao Wang and Wei-chen Lee

Abstract—**This study proposes an Adafruit IO-based IoT platform that integrates a data storage system with a motor control device and uses augmented reality to establish a monitoring interface. The motor control device uses an Arduino Leonardo controller and a NodeMCU gateway. Node-RED is a software tool for data storage and security assurance using a self-signed certificate. In addition, a real-time monitoring system of augmented reality was established using Unity to improve data visualization and facilitate the virtual control of the motor. The IoT platform was tested, and the results confirmed its feasibility. All software used in this research is open source, making it more accessible to source-limited enterprises.**

I. INTRODUCTION

The Industry 4.0 concept – including the digital twin, the Internet of Things (IoT), and augmented reality (AR) – can help the conventional manufacturing industry in many ways. For example, many manufacturing companies still face challenges establishing a human-machine collaboration system. Hence, the digital twin has emerged as a promising solution. It creates a virtual model of a target system by collecting real-time measurements and performing analytics. This technology has tremendous potential for monitoring the production status and predicting when potential problems will occur. In the meantime, IoT plays a vital role in creating an integrated system where devices can work together to increase automation. In addition, AR can supplement reality with information, enabling user interactions and creating an automated visualization system. By integrating complex processes into an integrated system, Industry 4.0 upgrades manufacturing factories and presents information in a convenient interface to improve production efficiency and reduce downtime. Using the digital twin, IoT, and AR technologies enables the creation of a highly collaborative and smart production environment.

In recent years, industrial automation has seen these technologies emerge. Azangoo et al. [1] successfully established a digital twin of a paper process system by extracting data from engineering documents using text and image processing techniques. Wu et al. [2] used a low-cost attitude sensor to measure the rotation angle of a robotic arm, by which the data went to the robotic arm's digital twin through the IoT platform ThingJS. Similarly, Liu et al. [3] utilized computer numerical control (CNC) machining parameters to predict time-varying errors and provided automatic compensation using a digital twin based on a heat transfer model.

Several researchers have proposed different methods for establishing an IoT platform. Rajalakshami et al. [4] employed the open-source platform Node-RED and Amazon web services (AWS) to control a Raspberry Pi single-board computer. Osaretin et al. [5] utilized Arduino as the controller to connect sensors, displays, motors, and other devices with Node-RED for collecting data and presenting the data on a web client established by NGINX.

On the other hand, AR has been implemented as the human-machine interface for the data display and motion control of industrial equipment. Sun et al. [6] utilized HoloLens to detect users' gestures and transmit the corresponding messages to a Raspberry Pi using Bluetooth. Similarly, Seitz et al. [7] monitored IoT devices using AR with a smartphone as an AR device. Caiza et al. [8] sent the data of a classification station to the Google Firebase cloud database via MQTT. Then, they used the AR platform Vuforia to present the machine status on a mobile phone.

Although AR allows users to interact with devices and control them through an IoT platform, cybersecurity issues are often overlooked, with many focusing solely on achieving IoT goals without considering the IoT architecture's capability to resist malicious attacks. Bassi [9] raised concerns about IoT security in Industry 4.0 and identified areas of concern. Meanwhile, Mullet et al. [10] proposed cybersecurity guidelines for Industry 4.0. Although the IoT architecture is vulnerable to exploitation, cybersecurity is typically considered a cost rather than a revenue. The significance of cybersecurity usually becomes obvious when a cybersecurity incident occurs.

To address the cybersecurity issue, researchers developed cybersecurity requirements for Industry 4.0 smart factories, including implementing firewalls, encrypting data, and adopting secure processing techniques. Gehrmann et al. [11] proposed a digital-twin model and corresponding security architecture that meets cybersecurity requirements. Their study showed that attackers could intercept or modify communications, even issuing arbitrary messages or requests to the digital twin. Thus, they used a virtual private network (VPN) and transport layer security (TLS) to establish a secure communication channel. Their study demonstrated that IoT communication could be efficient and ensure basic communication security, and cybersecurity should be considered when creating an IoT-based digital twin.

This work was supported by the National Science and Technology Council, Taiwan [Grant number NSTC 111-2813-C-011-055-E].

K. H. Wang is with the Department of Mechanical Engineering, National Taiwan University of Science and Technology, Taipei, 106335, Taiwan, (e-mail: Scottwang0314@gmail.com).

W. C. Lee is with the Department of Mechanical Engineering, National Taiwan University of Science and Technology, Taipei, 106335, Taiwan, (phone: 886-2-2737-6478; fax: 886-2-2737-6464; e-mail: wclee@mail.ntust.edu.tw).

As previously discussed, many researchers have made significant strides in the development of digital twins, IoT platforms, and AR, which have the potential to digitally change the current industry. However, cybersecurity remains a major concern in developing such technologies. In this research, we aim to develop a monitoring system for a direct current (DC) motor based on a digital twin and AR taking cybersecurity into account.

II. Research Methods

The research framework consists of three major parts, as shown in Fig. 1. The first part involves the DC motor control device that uses an Arduino Leonardo board to control and acquire data from a DC motor (part no. GR42X25, Dunkermotoren). The data is sent to the MQTT broker through NodeMCU. The second part involves the IoT platform that uses Adafruit IO as the MQTT broker to connect data streams from various devices. Node-RED processes the data from the MQTT broker, analyzes it, stores it in a MySQL database, and issues warnings to the digital twin device if needed. The third part is the display of the digital twin that uses Unity to present the DC motor data in the AR device. AR also allows human-machine interaction to control the motor.

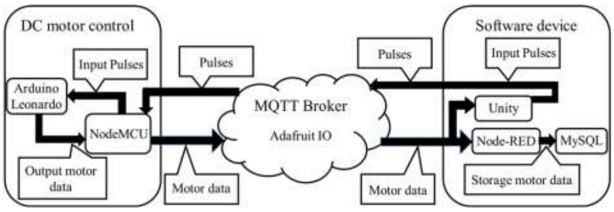

Figure 1. The research framework

The AR device with the proposed system can either start or stop the motor. This system can be utilized in factories to monitor equipment's parameters and increase monitoring efficiency, making it a cost-efficient solution for various industries.

A. DC motor control

Fig. 2 shows the wiring diagram of the DC motor control device. The Arduino Leonardo was used to control the speed of the DC motor. The motor's speed in RPM was measured by the rotary encoder (part no. RE30, Dunkermotoren) and acquired by the Arduino Leonardo. The speed of the motor was determined by the average voltage input, proportional to the duty cycle. The PWM input ranges from 0 (duty cycle 0%) to 255 (duty cycle 100%) in the Arduino IDE programming. The encoder detected the number of pulses per second (Pulse), which was used to calculate the motor's speed (RPM). The Pulse and PWM values were sent to the NodeMCU, which uploaded data to Adafruit IO for processing on the IoT platform. The NodeMCU has WiFi capability, which allows it to send the information to Adafruit IO through the internet. On the other hand, the virtual switch made using Unity can also control the on/off functions of the motor through the Ardafruit IO and NodeMCU.

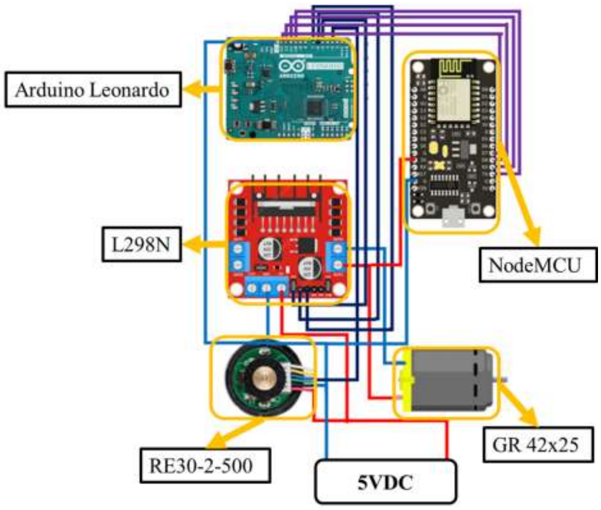

Figure 2. The hardware wiring diagram of the DC motor control device

B. IoT platform

In this study, we employed Node-RED to obtain the data from Adafruit IO for processing. Adafruit is a commercial IoT platform as a service (PaaS). Fig. 3 illustrates the framework, which achieved the following tasks: storing data and the associated timestamps in a local MySQL database, storing data and the associated timestamps as a CSV file on a local computer, and issuing a warning message using LINE Notify. In this research, the condition for issuing a warning message was when the ratio between the motor's actual speed and target speed was less than 0.2.

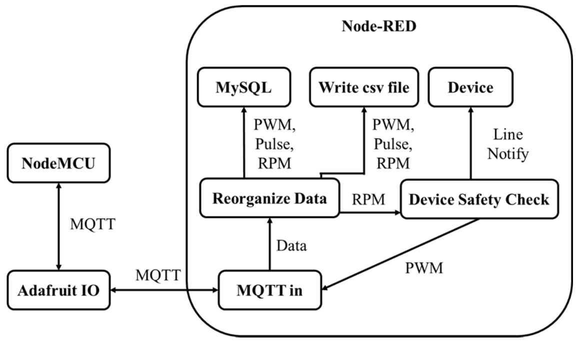

Figure 3. The IoT platform based on Adafruit IO and Node-RED

As shown in Fig. 3, data are sent from NodeMCU to Node-RED using the MQTT protocol in the node. Three parameters (PWM, Pulse, and RPM) sent from NodeMCU to the broker were combined into one, separated by commas. The first function node in Node-RED recovered the information back to three parameters, and another function node stored the data in the MySQL database with the timestamp added. The data were also saved as a CSV file on the local computer using the function node and the write file node. The other function node calculates the ratio between the actual and target motor speeds and outputs the result to the switch node. If the output result is true, the LINE Notify node will send a warning message to a specified LINE account.

To enable an SSL protocol, OpenSSL was used to generate a self-signed certificate placed in the same location as the Node-RED program. The Node-RED configuration file was then modified to enable SSL authentication. Then, the password was required to modify the internal flow of Node-RED to protect the data and parameters of the Node-RED platform.

C. Digital twin display

The Unity program was utilized to create the digital twin display, as shown in Fig. 4. The digital twin display can be used to show data from the DC motor and provide a user-friendly interface to operate the motor. The objects in the display, such as buttons, were created, subscribed, and bound to Adafruit IO through a C# program. This mechanism enabled the interface to display real-time data from the subscription. Similarly, the buttons were bound to the project, so the commands associated with the buttons' action could be sent to the MySQL database. Node-RED then sent the MySQL data to Adafruit IO for motor control. To establish the AR display, this study utilized the Unity Vuforia library. A recognizable image was set, and the established digital twin display interface was linked to the recognizable image. Once the camera successfully detects the recognizable image, the display interface will be displayed at the position of the image on the AR device – a mobile phone, in the case of this research.

Figure 4. Digital twin display created by using Unity

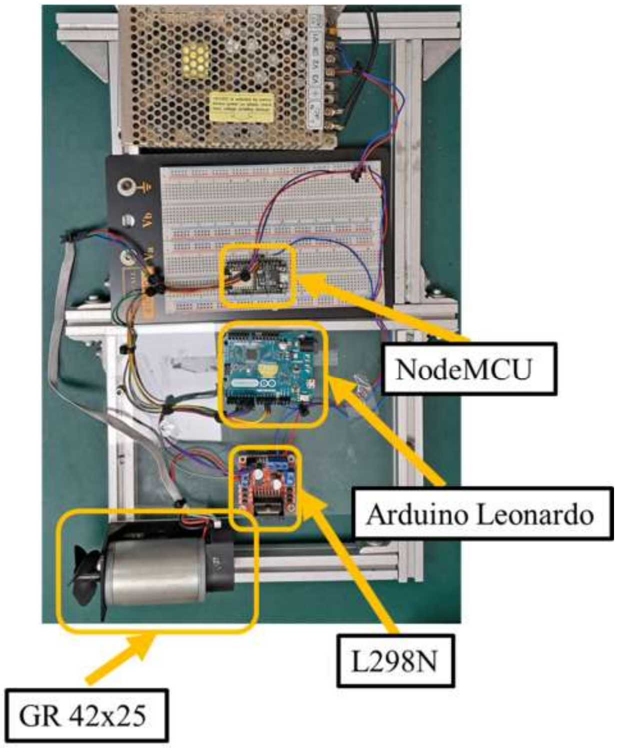

Figure 5. The completed DC motor control device in the research

TABLE I. TEST RESULTS

Time(s)	PWM	Pulse	RPM
26	255	6275	3764
28	255	42	15
40	255	38	12
42	255	34	8
44	255	7648	735
46	255	6538	3922

III. RESULTS AND DISCUSSION

The completed DC motor control device developed in this research is shown in Fig. 5. We performed the following tests on the device: (1) the DC motor ran for 25 seconds after startup; (2) a load (using fingers) was added to the motor for 15 seconds; (3) the load was removed from the DC motor, and the motor continued to run for another 20 seconds. Node-RED converted the data generated by the encoder on the motor into a CSV file and stored it on the host. The test result is listed in Table I. Since Adafruit IO has a limitation on the account we used, the data were sent every 2 s, resulting in latency in data transmission – about a one- to two-second delay – when the load changed. Nevertheless, the encoder effectively detected pulses according to the state of the motor. The motor speed was calculated within Arduino Leonardo to generate a speed in terms of RPM. It can be seen that the motor control device performed well per the target PWM requirements. The load was used to change the motor speed, and there was no compensation algorithm for the load in our program.

As mentioned, Node-RED enabled data acquisition from the motor control device and stored data in a MySQL database on a local computer. Previous tests confirmed that the stored data were correct. When the motor speed dropped due to an added load, the Node-RED program sent a notification to LINE via LINE Notify, enabling prompt notification of abnormalities. Notably, the test confirmed the proper functioning of the Node-RED platform.

When Node-RED runs on the host computer, it does so under an SSL certificate encryption system. An SSL self-signed certificate was generated by an algorithm based on the password set when the certificate was signed, ensuring that attackers could not access the content during data transmission. To access the content of the system, users need to input the username and password in the certificate to satisfy the security requirement of the device.

During motor operation, the Unity software displayed its AR interface on a mobile phone, as shown in Fig. 6. The software detected the pre-defined recognizable image through the camera of the AR device and displayed the interface on top of the image, with motor data displayed in real time. If an object, such as a finger, is detected between the image and the camera at the position of the button, the button will be triggered, and the motor will stop or start depending on which button is triggered. When testing the AR interface, a delay of about 2 s was observed between the displayed interface and the actual motor status due to data latency. It was confirmed that the AR interface could successfully monitor and control a device in real time.

Figure 6. Unity software displays the AR interface on a mobile phone

IV. CONCLUSION

The proposed IoT platform integrated a motor control system through WiFi and established a monitoring interface using the AR technique. The motor control system used Arduino Leonardo to control a DC motor. NodeMCU was the gateway between the control system and the IoT platform Adafruit IO for data acquisition and transmission. Node-RED was used to establish communication protocols, send alarms, store data, and perform other functions. Additionally, OpenSSL was adopted to generate self-signed certificates to ensure data transmission security using SSL. Unity was used to create an AR monitoring system that provided real-time data visualization and allowed control of the DC motor.

Various open-source software was used in this research to develop an IoT platform with AR and cybersecurity capability, which allowed operators to remotely inspect and control multiple devices separately and securely. The proposed system can also store the data in a database for analytics. Moreover, the IoT architecture and the associated important techniques related closely to smart manufacturing demonstrated feasibility in this research.

REFERENCES

[1] Azangoo, M., et al., A Methodology for Generating a Digital Twin for Process Industry: A Case Study of a Fiber Processing Pilot Plant. IEEE Access, 2022. 10: p. 58787-58810.

[2] Wu, Z., et al., A Low-Cost Digital Twin-Driven Positioning Error Compensation Method for Industrial Robotic Arm. IEEE Sensors Journal, 2022. 22(23): p. 22885-22893.

[3] Liu, K., et al., Time-Varying Error Prediction and Compensation for Movement Axis of CNC Machine Tool Based on Digital Twin. IEEE Transactions on Industrial Informatics, 2022. 18(1): p. 109-118.

[4] Rajalakshmi, A. and H. Shahnasser. Internet of Things using Node-Red and alexa. in 2017 17th International Symposium on Communications and Information Technologies (ISCIT). 2017.

[5] Osaretin, C.A., et al. Open Source IoT-Based SCADA System for Remote Oil Facilities Using Node-RED and Arduino Microcontrollers. in 2020 11th IEEE Annual Information Technology, Electronics and Mobile Communication Conference (IEMCON). 2020.

[6] Sun, Y., et al. MagicHand: Interact with IoT Devices in Augmented Reality Environment. in 2019 IEEE Conference on Virtual Reality and 3D User Interfaces (VR). 2019.

[7] Seitz, A., et al. Augmenting the industrial Internet of Things with Emojis. in 2018 Third International Conference on Fog and Mobile Edge Computing (FMEC). 2018.

[8] Caiza, G. and R. Sanz. Digital Twin for Monitoring an Industrial Process Using Augmented Reality. in 2022 17th Iberian Conference on Information Systems and Technologies (CISTI). 2022.

[9] Bassi, L. Industry 4.0: Hope, hype or revolution? in 2017 IEEE 3rd International Forum on Research and Technologies for Society and Industry (RTSI). 2017.

[10] Mullet, V., P. Sondi, and E. Ramat, A Review of Cybersecurity Guidelines for Manufacturing Factories in Industry 4.0. IEEE Access, 2021. 9: p. 23235-23263.

Keng-Hao Wang is a senior student studying mechanical engineering at the National Taiwan University of Science and Technology. Currently, he is interning at Cimforce Corp. and focusing his research on the field of Industry 4.0.

Wei-chen Lee received the B.S. degree from the National Taiwan University, in 1989, and the M.S. and Ph.D. degrees from the University of California at Berkeley, Berkeley, in 1993 and 1995, respectively, all in mechanical engineering. He studied part-time and received the second M.S. degree in electrical engineering from The Pennsylvania State University, Harrisburg, in 2004. After his Ph.D. degree, he joined Foxconn (also known as Hon Hai) as a CAE Engineer and became a Development Engineer later. In 2004, he returned to Taiwan and started teaching at the Department of Mechanical Engineering, National Taiwan University of Science and Technology. His research interests include CAE, CAM, signal integrity, AOI, industry 4.0, and mechatronics.