International Journal on Robotics, Automation and Sciences

RFID and Face Mask Detector Attendance Monitoring System

Wong Yih Haw, Lee Gin Chong*, and Sim Hock Kheng

***Abstract* - The article emphasizes the significance of attendance monitoring for safety during the COVID-19 pandemic and proposes an RFID-based solution coupled with face mask detection systems to address attendance challenges. The project aims to create a contactless monitoring system that ensures face mask compliance and provides real-time attendance data for data-driven decision-making. The article also covers various technology-related topics, including the historical usage of face masks, the development of attendance systems using biometric identification and electronic methods, and facial recognition technology's applications in surveillance and finance. It introduces XAMPP, a user-friendly web application development and testing tool, and presents an overview of the IC7408 chip used in digital electronics. The study's key findings show that increasing sample size and optimizing epochs and batch size improve face mask detection accuracy, while RFID scanner distance affects scanning delay and accuracy. The research provides valuable insights into the performance of the proposed attendance monitoring system.**

Keywords— RFID, Face Mask Detection, Attendance Monitoring.

I. INTRODUCTION

The COVID-19 pandemic has highlighted the importance of attendance tracking for safety and well-being at home and work. It enables contact tracing, promotes safety measures, optimizes resource allocation, and aids in remote work monitoring. Moreover, it ensures public health compliance and facilitates proactive interventions. This article introduces a comprehensive attendance monitoring solution that integrates RFID and face mask detector technologies. RFID utilizes electronic tags on personal belongings to automatically record attendance when individuals enter or leave a building. Face mask detector cameras ensure compliance with face mask regulations. The study explores the potential of RFID technology, its affordability, and applications in various fields.

RFID technology utilizes electronic tags to track attendance without physical contact, offering advantages such as increased accuracy and reduced manipulation. Additionally, a facial mask detection system can ensure compliance with mask-wearing regulations, contributing to infection prevention. The implementation of these technologies is crucial for UOW Malaysia KDU Penang College to improve student participation during the pandemic and overcome the limitations of manual attendance monitoring methods. Contactless attendance verification systems provide a safer and more efficient approach to maintain employee well-being and productivity in various organizations,

*Corresponding author, Email: ginchong.lee@uow.edu.my, ORCID: 0009-0006-7664-5851
Wong Yih Haw is with Department of Engineering, School of Engineering, Computing and Built Environment, UOW Malaysia KDU Penang University College, 32, Jalan Anson, 10400 Georgetown, Pulau Pinang. (e-mail: 0202042@student.uow.edu.my).
Lee Gin Chong* is with Department of Engineering, School of Engineering, Computing and Built Environment, UOW Malaysia KDU Penang University College, 32, Jalan Anson, 10400 Georgetown, Pulau Pinang. (e-mail: ginchong.lee@uow.edu.my).
Sim Hock Kheng is with Department of Engineering, School of Engineering, Computing and Built Environment, UOW Malaysia KDU Penang University College, 32, Jalan Anson, 10400 Georgetown, Pulau Pinang. (e-mail: hockkheng.sim@uow.edu.my).

International Journal on Robotics, Automation and Sciences (2023) 5,2:14-24
https://doi.org/10.33093/ijoras.2023.5.2.2
Manuscript received: 23 June 2023 | Revised: 15 July 2023 | Accepted: 10 August 2023 | Published: 30 September 2023

Published by MMU PRESS. URL: http://journals.mmupress.com/ijoras

considering the challenges posed by the COVID-19 epidemic.

The article is organized in three sections: Literature Review, Methodology, and Analysis and Results, providing an overview of the study, installation procedures, and performance evaluation of the system

A. Problem Statement

Due to the COVID-19 pandemic, UOW Malaysia KDU Penang University College has faced challenges in attracting student attendance. Currently, the college lacks a digital class attendance system and relies on a handwritten method, which is prone to errors and data loss. Implementing an RFID attendance system would allow for contactless attendance tracking, improving accuracy and reducing the risk of deception.

The COVID-19 pandemic has made it challenging to collect reliable attendance data, especially when students come in contact with confirmed cases. To mitigate this, the college needs an automated face mask detection system, which uses AI and computer vision to determine if individuals are wearing masks. To address these issues, The proposed framework that integrates RFID and Face mask Identifier Participation Checking Framework.

B. Objectives

The following are the project's primary goals:

- To create a contactless RFID and face mask detection attendance monitoring system capable of properly tracking staff attendance.
- To guarantee adherence to face mask standards and to enhance employee and students' safety and health.
- To alleviate administrative burdens related with tracking attendance as well as record keeping.
- To offer management with real-time attendance statistics for enhanced decision making.

II. Literature Review

A. Background

Face masks have a long history of being used to protect against respiratory ailments and environmental contaminants. Throughout history, people used various materials such as animal skins, leaves, and fabric to cover their faces during dusty or polluted conditions. The use of face masks became more prevalent during the 1918 influenza pandemic, one of the deadliest epidemics in history, which led to the creation of contemporary face masks [1]. Face masks have also been used in recent years to guard against air pollution and other environmental issues [1].

An attendance system is a method of recording people's presence in a specific area, such as a workplace, educational institution, or conference. In the past, attendance was tracked manually using logbooks or punch cards, but with advancements in technology, digital or electronic methods are now commonly used.

Biometric identification systems, including fingerprinting and facial recognition, have gained popularity due to their precision and efficiency. Attendance systems are widely used in schools, colleges, companies, and special events to monitor attendance, ensure compliance with commitments, and enhance security by identifying unauthorized individuals [2].

Facial recognition is a technology that identifies and verifies individuals based on their facial features using artificial intelligence and computer vision. It has gained significant attention and acceptance in recent years due to its applications in safety, monitoring, authorization, and personalization. Early developments in face recognition date back to the 1960s, but recent advancements in AI and computer vision have significantly improved its accuracy [3]. Deep learning algorithms, which use massive datasets of labeled facial photos, are commonly used in modern face recognition systems. This technology has various uses in sectors such as surveillance, security, finance, immigration, and marketing, but it also raises concerns about security, privacy, and biases [3].

Face mask detection is a technology that determines the likelihood of individuals wearing face masks in a specific area. It has become particularly relevant during the COVID-19 pandemic, as face masks are crucial in preventing virus transmission[4]. Face mask detection typically involves the integration of cameras and AI algorithms to analyze facial features and identify whether someone is wearing a mask [4]. This technology is used in airports, hospitals, schools, and corporations for purposes such as enforcing mask-wearing laws and monitoring compliance. While face mask detection technology is an important tool in preventing the spread of infectious diseases, it should be used alongside other preventive measures and is not 100% effective on its own [4].

B. Similar Projects

1) Detection of Facial Masks Using a Depthwise Separable Convolutional Neural Network Model During the COVID-19 Pandemic

The article focuses on data collection and preparation for a face mask identification research study. Two datasets, AIZOO and Moxa3K, were used. Preprocessing techniques like normalization, scaling, and cropping improved photo quality, while picture labeling highlighted foreground objects. A Depthwise Separable Convolutional Neural Network (DWS-CNN) based on MobileNet was employed to classify the photos. This design optimized computational efficiency without sacrificing accuracy [8]. The article explains traditional and depthwise separable convolutions used in MobileNet and their computational requirements.

Implementing MobileNet with depth-separable filters achieved high accuracy in categorizing masked and unmasked faces. The DWS-CNN model outperformed Full Convolution MobileNet, increasing accuracy by 0.6% and reducing computation time and trainable parameters [5].

C. Machine Learning Platform

Artificial intelligence and machine learning heavily rely on free and open-source libraries for model building and execution. Prominent libraries in this domain include TensorFlow, Keras, PyTorch, scikit-learn, Theano, Caffe, Torch, MXNet, Hugging Face Transformers, and DeepLearning4j [4].

TensorFlow is a widely-used open-source library by Google for machine learning and AI. It offers a versatile framework for constructing and training neural networks, with support for various programming languages and deployment on different platforms. Keras is a popular framework that works seamlessly with TensorFlow, CNTK, or Theano. It provides a user-friendly API for creating complex neural networks, including convolutional neural networks (CNNs), recurrent neural networks (RNNs), and autoencoders [4].PyTorch, developed by Facebook, is known for its dynamic computing graph and simple API. It allows developers to use Python to create and refine neural networks, making it suitable for computer vision and natural language processing tasks.

Scikit-learn is a Python library that offers efficient data analysis and machine learning capabilities. It provides a range of supervised and unsupervised learning methods, making it ideal for classification, regression, clustering, and dimensionality reduction tasks [4]. Theano is a numerical computing toolkit for Python, often used for deep learning models like CNNs and recurrent networks. It is recognized for its speed and efficiency, making it suitable for large-scale AI workloads. Caffe, developed by Berkeley AI Research (BAIR), is known for its speed and modularity. It supports multiple programming languages and is widely used in academia and industry for deep learning applications [4].

Torch is a machine learning toolkit for Lua, favored for its simplicity and versatility in areas such as natural language processing and computer vision. MXNet, an Apache project, is a scalable machine learning platform compatible with multiple programming languages. It offers fast distributed training and is suitable for large-scale operations. DeepLearning4j is a Java-based deep learning library. It provides a comprehensive framework for building and deploying neural networks and excels in scalability and integration with Spark and Hadoop for large data applications [4].

1) TensorFlow

TensorFlow is a powerful open-source library for machine learning and artificial intelligence. Developed by Google Brain in 2015, it supports various algorithms like deep neural networks, convolutional neural networks, and recurrent neural networks [4]. Key advantages of TensorFlow include its ability to handle large datasets using computational graphs and its compatibility with CPUs, GPUs, and TPUs (tensor processing units). It is widely used for image classification, object detection, audio recognition, natural language processing, and recommendation systems. TensorFlow offers both supervised and unsupervised learning, with various optimization techniques, loss functions, and activation functions [4].

Keras, an API built on TensorFlow, simplifies neural network design and programming with pre-built layers like convolutional and recurrent layers. TensorFlow supports distributed computation, enabling faster training on multiple machines. It provides tools and techniques for efficient distributed learning. With a large community and extensive resources, TensorFlow offers ample learning and debugging support. It is widely adopted in various industries, thanks to its flexibility and scalability, particularly in healthcare, finance, and transportation [4].

2) Keras

Keras is a high-level neural network API for deep learning, built on TensorFlow, Theano, and Microsoft Cognitive Toolkit. It offers an easy interface for designing and training neural networks, supporting various layers and allowing customization of model design [4]. Keras supports multiple backends like TensorFlow, Theano, and Microsoft Cognitive Toolkit, facilitating flexibility and model transfer. It includes default optimization techniques, loss functions, and activation functions that can be tailored to specific applications, supporting both supervised and unsupervised learning [4]. Keras provides additional utilities for deep learning, such as data augmentation, model visualization, and debugging. It has a vibrant community with abundant resources, documentation, and pre-existing models for various applications.

D. RFID Attendance Monitoring System

RFID attendance monitoring systems provide precise and efficient monitoring, saving time and reducing errors. They are adaptable and customizable to meet organizational needs. Real-time monitoring helps identify patterns and intervene promptly. Integration with other systems simplifies data management and reporting. These systems enhance security by preventing unauthorized access [6]. However, challenges include implementation costs and data security concerns. In summary, RFID attendance monitoring systems offer precision, efficiency, adaptability, real-time monitoring, integration, and security benefits, simplifying attendance monitoring and administration [6].

E. Web Based Server

XAMPP is a popular open-source program for designing and testing web applications locally. It includes Apache, PHP, MySQL, and Perl, simplifying setup and deployment. The XAMPP control panel is a user-friendly GUI for managing services and modules. Users can start, stop, and customize Apache, MySQL, and more with ease. The control panel also allows configuration and management of installed modules. It provides a log window for debugging and offers options for upgrading XAMPP. Overall, the XAMPP control panel is a vital tool that simplifies managing and overseeing XAMPP services and modules for developers.

F. Integration IC

The IC7408 chip is commonly used in digital electronics and belongs to the 7400 series of integrated circuits. It functions as a quadruple two-input AND gate, with four separate AND gates, each having two inputs. This versatile 14-pin DIP IC operates within a voltage range of 4.75 to 5.25 volts and is widely employed due to its simplicity, affordability, and availability.

III. Methodology

A. Introduction to Methodology

The RFID and face mask detection attendance monitoring system uses RFID tags, scanners, and mask detection software to efficiently manage attendance data. Here's how it works:

1. Tagging: Employees or students receive RFID tags linked to their personal information, which can be attached to ID cards.

2. RFID reader installation: RFID readers at entry and exit points interact with the RFID tags, collecting attendance data.

3. Face mask detection: Facial mask identification software captures images and alerts if a mask is not detected, ensuring compliance with mask-wearing protocols.

4. Data capture: Attendance data is stored in a centralized database accessible to authorized HR personnel.

5. Reporting: Reports highlight attendance patterns, helping identify individuals with frequent absences or tardiness.

This system streamlines attendance management, improves accuracy, and promotes compliance with mask-wearing rules.

B. System Overview

The RFID and face mask detection system tracks attendance reliably using RFID tags, readers, and face mask detection. Here's an overview:

1. RFID Tags: Small devices on ID cards provide unique codes for attendance monitoring.

2. RFID Readers: Scanners at entry/exit points collect real-time attendance data.

3. Face Mask Detection: Software captures images to verify mask compliance and trigger alerts.

4. Centralized Database: Attendance data is securely stored and accessible to authorized personnel.

5. Reporting: Generates reports on attendance patterns, aiding HR and academic departments.

This system streamlines attendance management, ensures mask compliance, and promotes a safe environment.

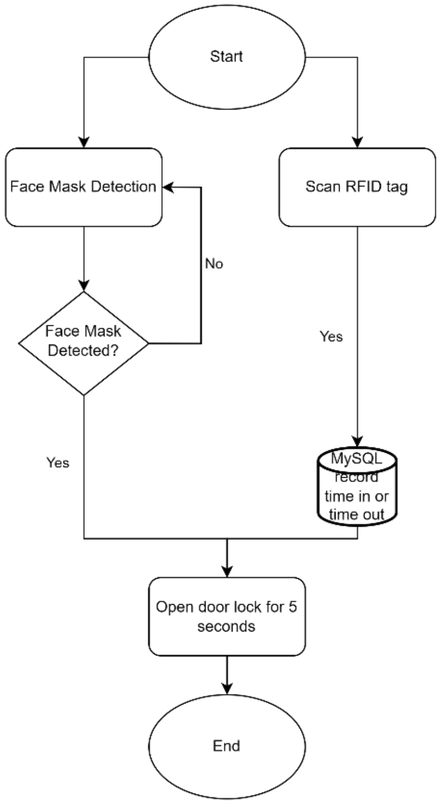

FIGURE 1. General flow of the system.

The flowchart in Figure 1 illustrates the system's overall flow, combining face mask detection and RFID attendance monitoring. Both systems start simultaneously. The face mask detection system checks if a person is wearing a mask. If not, they cannot proceed to the next step. Simultaneously, the RFID attendance monitoring system begins. Once both requirements are met (mask detection and attendance data collection), the solenoid door lock activates for 5 seconds. This ensures that only individuals wearing masks and registering attendance can enter the premises.

C. Hardware Methodology

1) Hardware Overview

The flowchart in Figure 2 shows the hardware implementation of the system, combining face mask detection using a webcam and laptop, and RFID attendance monitoring using ESP8266. Both systems start together.

The face mask detection system uses a webcam and laptop. The webcam detects faces, and the video stream is sent to the laptop. The detection result is transmitted to Arduino Uno via HC05 Bluetooth module. Simultaneously, the RFID attendance monitoring system is initiated. Individuals scan their RFID tags with the RFID scanner. ESP8266 detects the tag and sends the information to the MySQL server.

Once both face mask detection and attendance data collection are successful, the IC7408 (AND gate logic IC) activates the solenoid door lock. This ensures that only individuals wearing masks and registering attendance can enter the premises.

2) Face mask Detection Hardware

The hardware implementation of the Python face mask detection system involves the following steps: component selection, camera deployment, camera-to-laptop connection, programming, LED connection, system testing, and integration with other systems.

The webcam is used for its non-intrusive, real-time, and cost-effective benefits in detecting face masks. The HC05 Bluetooth module transmits data from Python to Arduino, facilitating communication between the two. Data transmitted includes '1' for a detected face mask and '0' for no mask. Arduino Uno and HC05 are chosen for their ease of use and compatibility. Arduino Uno provides multiple input/output ports, while the HC05 module enables wireless connectivity with external devices. The integrated setup of Arduino Uno and HC05 wirelessly communicates Python data to the Arduino board, controlling the LED based on the received data. This setup offers a simple and cost-effective solution for LED operation using Python.

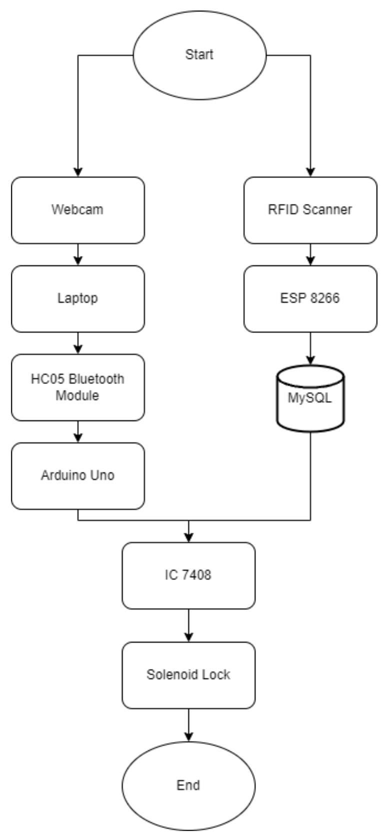

FIGURE 2. Block diagram for hardware implementation.

3) RFID Attendance Monitoring System Hardware

The RFID attendance system uses the ESP8266 and RFID RC522 reader module. The RFID reader is connected to the ESP8266 board for power and serial communication.

To set up the system, turn on the ESP8266 board and install the required code libraries using the Arduino IDE.

Create an Arduino program to initialize the RFID reader, retrieve RFID tag IDs, and store attendance data in the ESP8266's database. The ESP8266 is an affordable Wi-Fi microcontroller that is easy to program and connect to the internet. The RC522 RFID module is a cost-effective reader for reading and writing RFID tag information. Together, they provide a simple and low-cost solution for remote RFID attendance monitoring [7]. Both modules have a strong developer community with available libraries and code samples, allowing for easy customization and scalability.

4) Hardware For Integration Of Systems

In this project, an IC7408 chip is used as a versatile integrated circuit from the 7400 series, functioning as a quadruple two-input AND gate. It plays a key role in obtaining the output from the face mask detection system and RFID attendance system, ensuring that the solenoid is triggered only when both conditions are met.

A junction box is used to protect electrical connections and provide a safe and organized space for wire splices. It offers durability and resistance to heat and moisture. A 5V relay is an electrical device that controls high voltage and current applications. It provides separation between the control circuitry and the device being controlled. In this project, it controls the solenoid valve for simulating door locking/unlocking. A solenoid lock is an electromechanical mechanism that engages or disengages the locking bolt or latch when an electrical current is applied. It is commonly used in security systems for access control. A LiPo battery, with its high energy density, low self-discharge rate, and fast discharge rate, is a rechargeable power source. It powers the ESP8266, Arduino Uno, and solenoid lock in this project, while another LiPo battery acts as a power bank for the Arduino Uno and ESP8266.

D. Software Methodology

1) Software Overview

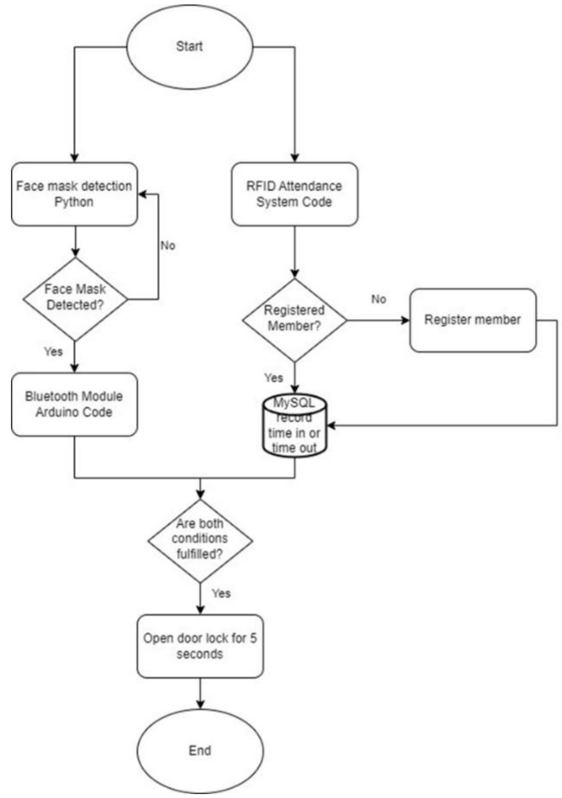

FIGURE 3: Block Diagram of software implementation.

An RFID and face mask detection attendance system combines RFID and facial recognition technologies to monitor attendance and enforce face mask compliance. It includes RFID Reader Software, Facial Recognition Software, Attendance Monitoring Software, Mask Detection Software, and a Dashboard.

The RFID Reader Software reads RFID tags linked to ID cards, which are then processed by the Attendance Monitoring Software. This central hub monitors attendance, tracks late arrivals and early departures, and manages attendance records. The Facial Recognition Software scans faces and determines if a person is wearing a mask, while the Mask Detection Software analyzes video data to identify mask non-compliance. A Dashboard provides a user interface for real-time attendance monitoring and reporting, enabling managers to monitor attendance and ensure safety compliance.

The flowchart (Figure 3) illustrates the system's software implementation, with both face mask detection and RFID attendance monitoring operations running concurrently. The face mask detection system checks for mask presence, while the RFID attendance system registers users before collecting their attendance data using MySQL. Once both requirements are met, indicating mask compliance and attendance registration, the solenoid door lock activates for 5 seconds. This ensures that only individuals who meet the mask and attendance criteria can enter the premises.

2) Face Mask Detection Model Training Code

The Python code segment imports various packages and modules required for creating a deep learning algorithm for image categorization using Keras and TensorFlow libraries. These imports include packages for image preprocessing, pre-trained models, layers, models, optimizers, data preparation, data splitting, model evaluation, image processing, data visualization, numerical computations, and operating system interaction. These packages provide the necessary tools and functionalities for building and training the deep learning model.

The code then sets up variables and loads the image data into the script. It initializes variables such as the initial learning rate (INIT_LR), number of epochs (EPOCHS), and batch size (BS). It also specifies the directory containing the image data and the categories or labels associated with the images. The code initializes empty lists for storing the image data and labels. The code then loops through the categories, accesses the image files in the corresponding directory, loads and preprocesses each image, and appends the processed image data to the data list and the category label to the labels list. This prepares the image data and labels for further processing and model training.

3) Face Mask Detection Code

The code imports the required libraries and establishes a serial connection with the HC05 Bluetooth module for communication. It defines a function called 'detect_and_predict_mask', which takes an input frame, faceNet, and maskNet (pre-trained models). Within the function, the frame is preprocessed by transforming it into a blob and fed into the faceNet model to identify faces.

The function outputs the identified faces, their locations, and mask detection predictions for each face. After that, the code includes a loop that iterates over the face detections in the frame. If the confidence score of a detection is above a threshold, the bounding box coordinates of the face are extracted and scaled. The face area is then processed and added to the 'faces' and 'locs' lists. After iterating through all detections, if any faces are found, the 'faces' list is sent through the maskNet model to predict if masks are worn (stored in 'preds'). Finally, the 'detect_and_predict_mask' function returns the 'locs' and 'preds' lists. The code also loads the weights and architecture files for the faceNet and maskNet models, starts the video stream, and captures real-time video frames.

And then, the code is used to determine if a person in a live video feed is wearing a mask. It utilizes pre-trained neural network models to recognize faces and predict mask-wearing likelihood. The code interacts with an Arduino board through the serial port, sending signals ('1' or '0') indicating if a person is wearing a mask or not. It displays a bounding box around the face and a label showing the mask-wearing probability. The script runs indefinitely until the user presses the 'q' key, and it shuts down the video stream and cleans up the environment at the end.

4) RFID attendance monitoring system code

The Arduino code is used to run the ESP8266 which in turn run the attendance monitoring system. The system incorporates RFID and WiFi libraries. It initializes the MFRC522 RFID module and sets the pins accordingly. Additionally, it configures the WiFi connection credentials and device token.

The code reads RFID tags/cards using an RFID reader (MFRC522) and sends the tag/card ID to a server via WiFi. It includes the necessary libraries for the MFRC522 and ESP8266 WiFi modules. The code sets constant variables for pins and WiFi credentials. In the setup function, it establishes a serial connection, configures an LED pin, and initializes the MFRC522 module. The loop function checks for WiFi connectivity, detects new cards, reads the card ID, and sends it to the server. Another function shown is responsible for transmitting the card ID to the website's server using HTTP GET. It checks WiFi connectivity, creates objects for WiFi and HTTP clients, generates the necessary URL and data variables, and sends the HTTP request. It then examines the response code and content, performs actions based on the response, and terminates the HTTP connection.

There is also code for connecting the NodeMCU to a WiFi network. The connectToWiFi() function disables the WiFi mode, waits for it to exit, sets the mode to STA (station mode), and attempts to connect to the specified network. It displays the connection status on the serial monitor and waits until the connection is established. Apart from the code, the system employs the XAMPP

Control Panel software as shown in Figure 4 to host the Apache web server and MySQL database for the RFID attendance system. The attendance system includes a website hosted on the Apache web server, where attendance records can be monitored.

The website's attendance monitoring section displays the time in and time out when a user scans their RFID card or tag as shown in Figure 5. Lastly, there is a feature called "Log Filter/Export To Excel" that allows the moderator of the attendance monitoring system to obtain specific data.

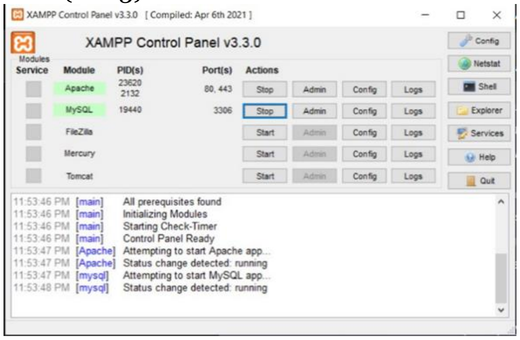

FIGURE 4. XAMPP Control Panel.

FIGURE 5. Attendance system website.

E. *Integration of System*

In this project, the IC7408 is used to combine the outputs from the face mask detection system and RFID attendance system. By doing so, it ensures that the solenoid is activated only when both face masks are worn and attendance is recorded, making it an ideal logic gate for this purpose. The project also incorporates a junction box which serves the purpose of housing and organizing the components for safety and aesthetics.

IV. RESULT, ANALYSIS & DISCUSSION

A. *Introduction to Result, Analysis & Discussion*

This section presents the results of experiments on face mask detection and RFID scanning, investigating the impact of sample size, epochs, batch size, ambient luminosity, and scanning range on model precision. Findings revealed that increasing sample size improved face mask identification up to a certain point, while exceeding a specific number of epochs led to overfitting and decreased accuracy. Batch size had an optimal value, and well-lit areas improved face mask identification. The scanning delay increased with greater distance between the RFID scanner and tag, and scanning accuracy decreased with larger scanning range. Using 3000 samples yielded the best accuracy, and training for at least 30 epochs was necessary for optimal performance. A batch size of 40 achieved the highest accuracy, and epochs had the greatest influence on performance. Brightness levels affected face mask detection, and scanning latency varied with attempts. RFID tag detection was limited to specific distances.

B. *Model Accuracy In Relation To Sample Size*

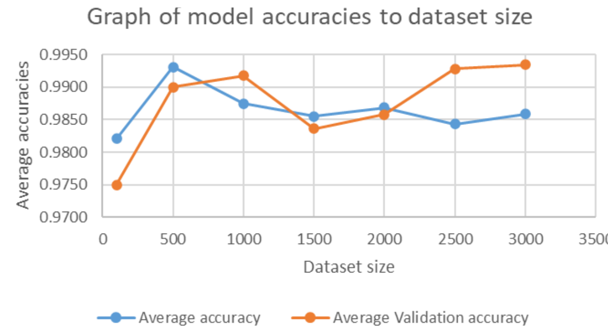

FIGURE 6. Graph of model accuracies to dataset size.

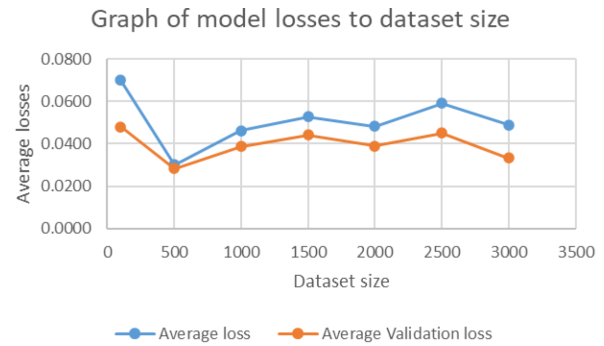

FIGURE 7. Graph of model losses to dataset size.

AI model training relies on dataset size, with larger datasets improving performance by providing more information. However, the relationship between dataset size and performance is not always linear, with diminishing returns. Large datasets reduce overfitting, while small datasets can lead to poor generalization [8]. Model complexity also affects the required dataset size. Experimentation is necessary to find the optimal size. Performance indicators include average loss, validation loss, accuracy, and validation accuracy, aiming for lower loss and higher accuracy with larger datasets, although exceptions can occur [8].

According to the results shown in Figure 6 and 7, increasing the dataset size generally leads to a decrease in average loss and average validation loss, indicating improved performance. However, there are exceptions, such as a higher average validation loss for a dataset size of 1000 compared to 500, highlighting the need for experimentation to determine the most effective dataset size. Regarding accuracy, both average accuracy and average validation accuracy improve with larger dataset sizes, except for a slightly lower average validation accuracy for a dataset size of 1500 compared to 1000.

This emphasizes the importance of exploring different dataset sizes to achieve optimal performance.

The findings indicate that a dataset size of 2500 is the most optimal for the model's performance, achieving the highest average validation accuracy of 0.9858 among all dataset sizes. Additionally, this dataset size exhibits the lowest average validation loss, demonstrating effective generalization by the model. Therefore, the model performs better with this particular dataset size compared to others.

C. Model Accuracy In Relation To Number Of Epochs

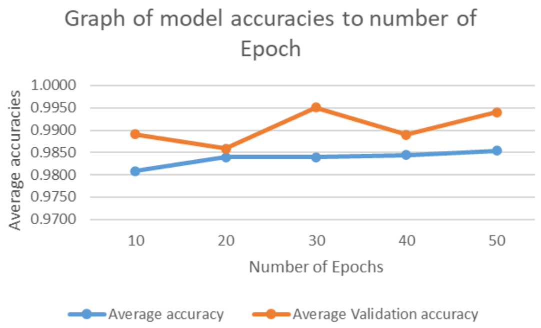

FIGURE 8. Graph of model accuracies to number of Epoch.

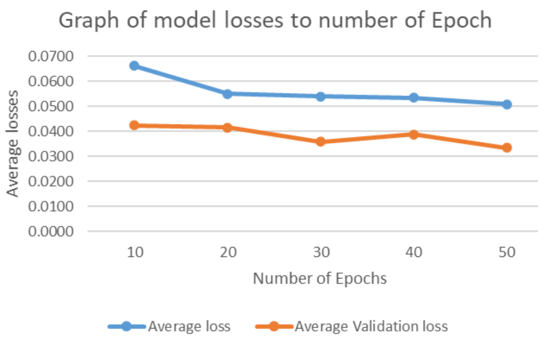

FIGURE 9. Graph of model losses to number of Epoch.

Machine learning depends on training accuracy and epochs. Increasing epochs improves accuracy, but too many can cause overfitting. In picture categorization, layers extract features and a final layer creates a loss function. Training adjusts parameters using stochastic gradient descent. Instances are analyzed, weights updated based on gradients during training. More epochs allow better parameter adjustments. Monitoring validation accuracy identifies overfitting. Epochs depend on model complexity, dataset size, and data quality. An epoch is a full dataset run, improving performance. Excessive training causes overfitting, performing poorly on new data.

As shown in Figure 8, the highest average accuracy achieved was 0.9854 at 50 epochs in the training set, and 0.9951 at 30 epochs in the validation set, indicating effective generalization to new data. As shown in Figure 10, average validation loss decreased from 0.0424 to 0.0334 across epochs, suggesting improved prediction accuracy during training. However, excessive epochs can cause overfitting and poor performance on new data. Monitoring validation performance and choosing the optimal number of epochs is crucial.

Overall, increasing epochs enhances model performance. The ideal number appears to be 30, with acceptable average losses (0.0540) and validation losses (0.0358), as well as high accuracy (0.9840) and validation accuracy (0.9951).

D. Model Accuracy In Relation To Number Of Batch Size.

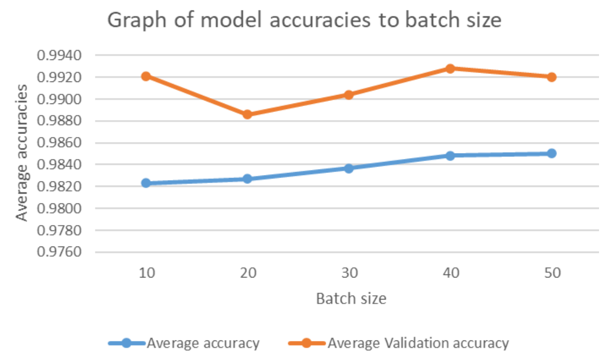

FIGURE 10. Graph of model accuracies to batch size.

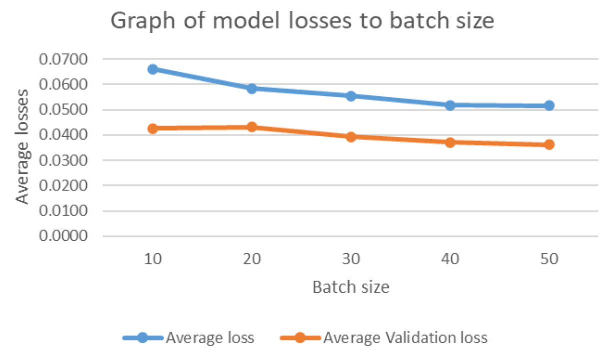

FIGURE 11. Graph of model losses to batch size.

The batch size is a crucial hyperparameter in machine learning, impacting speed, efficiency, and effectiveness. Larger sizes yield faster convergence, while smaller sizes prevent overfitting in image categorization. The suitable size depends on complexity, dataset size, and other factors [9]. Experimenting and selecting the best size using validation data is common. Batch size choice should align with memory limitations and other hyperparameters. Ultimately, it significantly affects training outcomes [9].

Figure 10 demonstrates the rise in average and validation accuracy. However, the rate of improvement slows with larger batch sizes. It also has a low loss of validation as shown in Figure 11 and a good validation accuracy, indicating that it generalizes effectively to unseen data. Choosing the appropriate batch size requires considering both accuracy and speed. Smaller sizes offer more precise weight updates but longer training times due to more iterations. Larger sizes may reduce training time but could lead to poorer generalization with fewer samples.

Based on the data, a batch size of 40 strikes a balance between accuracy and speed. It achieves high precision (0.9848) with fewer iterations, resulting in shorter training time. It also exhibits low validation loss and good validation accuracy, indicating effective generalization to new data.

E. Face Mask Detection Accuracy In Relation To Surrounding Brightness

TABLE 1. Face mask detection capability with surrounding brightness.

Surrounding brightness, E_V (Lux)	Face Mask Detected
1	No
3	Yes
5	Yes
10	Yes
32	Yes
97	Yes
126	Yes

Face mask detection software is crucial for monitoring compliance with mask regulations during COVID-19. However, accuracy can be affected by environment brightness. Low-light conditions make it challenging to distinguish faces from surroundings, while glare in bright light can hinder mask recognition [10]. Table 1 shows the system performs well across various lighting conditions, except in near darkness. This versatility allows effectiveness in different scenarios, aiding in COVID-19 transmission control. High accuracy is achieved through modern algorithms and adaptable webcams that adjust settings based on brightness, minimizing lighting impact. Powerful computer vision technology recognizes facial features and detects masks, reducing the influence of external factors like surrounding light. Note that accuracy may vary based on mask type and face position, and changes in brightness due to motion or environment alterations can affect performance.

F. RFID Scanning Latency

TABLE 2. RFID scanning latency.

Attempts	Latency Time (seconds)
1	1.26
2	2.29
3	3.83
4	4.64
5	2.60
6	1.48
7	6.14

Scanning latency is critical in RFID usage, representing the delay between tag detection and data acquisition. Factors like distance, signal intensity, and tag type influence scanning delay. Passive RFID tags used here have longer delays than active tags, which have their power source for faster communication. The operating environment and wireless interference can also affect scanning delay.

Considering scanning latency is crucial for optimal RFID performance. Table 2 shows delay times in a seven-trial operation, with an average latency of 3.18 seconds. Variations exist among attempts, with some showing reduced latency and others higher delays. Attempts 2 and 5 had notably longer latency times, suggesting potential processing limitations for high-volume scanning. Further investigation is needed to determine the cause of these anomalies. Analyzing latency provides insights into system performance and areas for improvement.

G. RFID Card Or Tag Scanning Distance

TABLE 3. RFID card or tag scanning distance.

Scanning distance, L (cm)	RFID tag or card detection
0.5	Yes
1.0	Yes
1.5	No
2.0	No
2.5	No
3.0	No

RFID enables wireless reading of unique identification on tags or cards. The scanning range, the maximum distance between the reader and the RFID tag, is crucial for data retrieval. Table 3 shows RFID tag detection at different scanning distances. Tags were identified at 0.5 cm and 1.0 cm, but not beyond. The optimal scanning distance for this system is around one cm, beyond which performance constraints may arise due to frequency, reader power, or antenna type. Environmental factors can affect tag detection. The experiment conducted in a disruptive setting may have impacted the maximum scanning distance. Cross-interference with devices like WIFI or Bluetooth, sharing frequency bands, could be a potential cause.

V. CONCLUSION

In conclusion, contactless RFID and face mask detection attendance monitoring devices have successfully achieved their objectives, improving attendance management, safety, and reducing administrative burdens. These technologies have enhanced attendance monitoring in various organizations.

Regarding dataset size, 3000 samples produce optimal accuracy and validation results, although smaller datasets show slightly higher average loss, affecting model performance. The number of epochs significantly impacts model performance, with accuracy increasing until 30 epochs, while validation loss decreases steadily. Training for at least 30 epochs is necessary for optimal performance. Batch size has a modest influence, with a size of 40 yielding the best results.

Brightness in the environment affects face mask detection, with increasing brightness improving mask identification. However, masks are not recognized at the lowest brightness level of 1 Lux. Scanning RFID tags or cards is successful only at distances of 0.5 cm and 1.0 cm, suggesting scanning range affects detection.

Inconsistencies in latency times occur as scanning attempts increase, indicating limitations in high-volume scanning with the current setup. Further investigation is needed to understand additional factors influencing system and model performance. Overall, dataset size, epoch number, batch size, brightness, and scanning range impact system performance.

VI. Recommendation

Data Encryption: Attendance data security in RFID and face mask detection systems requires encryption during transmission and storage. TLS or SSL protocols ensure secure transmission, while AES encryption protects data in storage. Regular security audits by impartial third parties enhance system safety.

Analytics and Reporting: Effective attendance monitoring systems with RFID and face mask detection offer analytics and reporting. Analyzing attendance data reveals insights into patterns, timeliness, and mask compliance, enabling improvement and corrective actions. Attendance trends and punctuality reports address absenteeism and optimize schedules and resources.

Real-Time Notifications: Efficient attendance monitoring systems provide real-time alerts through web portals, mobile apps, or email. Instant messages upon attendance recording address mask non-compliance and attendance issues promptly. They ensure appropriate actions like providing masks, contacting absent employees, and maintaining a safe environment while ensuring complete attendance records.

Acknowledgment

Deep appreciation for unwavering guidance and support, and heartfelt thanks to research participants, family, friends, and UOW Malaysia KDU Penang University College staff for their invaluable contributions.

Funding Statement

No funding was funded for this paper.

Author Contributions

Wong Yih Haw: Conceptualization, Data Curation, Methodology, Validation, Writing – Original Draft Preparation;

Lee Gin Chong: Project Administration, Supervision, Writing – Review & Editing;

Sim Hock Kheng: Project Administration, Supervision, Writing – Review & Editing.

Conflict of Interests

No conflict of interests were disclosed.

Ethics Statements

Our publication ethics follow The Committee of Publication Ethics (COPE) guideline. https://publicationethics.org/

References

[1] B. J. Strasser and T. Schlich, "A history of the medical mask and the rise of throwaway culture," The Lancet, vol. 396, no. 10243, pp. 19–20, Jul. 2020. DOI: https://doi.org/10.1016/S0140-6736(20)31207-1

[2] A. Singh Upreti, P. Kumar, A. Kumar, and A. Kumar, "Attendance Monitoring System Using Face Recognition," *International Journal of Information Technology*, vol. 1, no. 3, p. 5343, 2022. URL: https://ijitra.com/index.php/ijitra/article/view/41 (Accessed: 9 July 2023)

[3] I. Adjabi, A. Ouahabi, A. Benzaoui, and A. Taleb-Ahmed, "Past, present, and future of face recognition: A review," Electronics (Switzerland), vol. 9, no. 8, pp. 1–53, Aug. 2020. DOI: https://doi.org/10.3390/electronics9081188

[4] Vibhuti, N. Jindal, H. Singh, and P. S. Rana, "Face mask detection in COVID-19: A strategic review," Multimedia Tools and Applications, vol. 81, no. 28, pp. 40013–40042, Nov. 2022. DOI: https://doi.org/10.1007/s11042-022-12999-6

[5] M. Z. Asghar, A. Rahman, F. Ahmad, A. Khan, and M. S. M. Saad, "Facial Mask Detection Using Depthwise Separable Convolutional Neural Network Model During COVID-19 Pandemic," Frontiers in Public Health, vol. 10, Mar. 2022. DOI: https://doi.org/10.3389/fpubh.2022.855254

[6] O. T. Arulogun, A. Olatunbosun, F. O. Fakolujo, and O. M. Olaniyi, "RFID-Based Students Attendance Management System," International Journal of Scientific and Engineering Research, vol. 4, no. 2, pp. 1–9, Feb. 2013. URL:https://www.ijser.org/researchpaper/rfid-based-students-attendance-management-system.pdf (Accessed: 10 July 2023)

[7] S. Amsler and S. Shea, "What is RFID and how does it work?," TechTarget, 2021. URL:https://www.techtarget.com/iotagenda/definition/RFID-radio-frequency-identification (Accessed: 10 July 2023)

[8] C. A. Ramezan, T. A. Warner, A. E. Maxwell, and B. S. Price, "Effects of training set size on supervised machine-learning land-cover classification of large-area high-resolution remotely sensed data," Remote Sensing, vol. 13, no. 3, pp. 1–27, Feb. 2021. DOI: https://doi.org/10.3390/rs13030368

[9] P. M. Radiuk, "Impact of Training Set Batch Size on the Performance of Convolutional Neural Networks for Diverse Datasets," Information Technology and Management Science, vol. 20, no. 1, pp. 20–24, Jan. 2018. DOI: https://doi.org/10.1515/itms-2017-0003

[10] N. M. Athif, F. Sthevanie, and K. N. Ramadhan, "Face Mask Detection Under Low Light Condition Using Convolutional Neural Network (CNN)," JIPI (Jurnal Ilmiah Penelitian dan Pembelajaran Informatika), vol. 8, no. 1, pp. 281–290, Feb. 2023. DOI: https://doi.org/10.29100/jipi.v8i1.3324