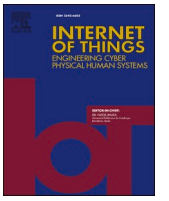
Internet of Things 25 (2024) 101089

Contents lists available at ScienceDirect

Internet of Things

journal homepage: www.sciencedirect.com/journal/internet-of-things



Lightweight face recognition-based portable attendance system with liveness detection

Nico Surantha [a,b,*], Boy Sugijakko [a,*]

[a] Computer Science Department, BINUS Graduate Program - Master of Computer Science, Bina Nusantara University, Jakarta, 11480, Indonesia
[b] Department of Electrical, Electronics and Communication Engineering, Faculty of Engineering, Tokyo City University, 1-28-1 Tamazutsumi, Setagaya-ku, Tokyo 158-8557, Japan

ARTICLE INFO

Keywords:
Liveness detection
Face recognition
Portable attendance system

ABSTRACT

Face recognition systems that do not implement liveness detection are susceptible to face spoofing attacks. This vulnerability implies that an attacker could disguise themselves as another individual and the system would falsely take the attendance of that other individual. To prevent these attacks, a liveness detection step can be implemented before recognizing subjects. Face recognition-based attendance system devices are typically installed at the entrance to an event or space, so having a portable device that can be easily relocated is practical and efficient. Hence, face recognition systems should be lightweight enough to be able to run on portable devices with limited computational power. Implementing liveness detection will increase the system's processing time. Therefore, this study aims to develop a lightweight liveness detection method that can be run on a Raspberry Pi. To achieve this, several pre-trained models were evaluated and MobileNetV2 was chosen based on the results. The MobileNetV2 model was then trained using transfer learning method. The proposed attendance system achieved an average processing time below 0.6 s and 96 % accuracy for live subjects, 79 % accuracy for level A spoof attacks, 83.7 % accuracy for level B spoof attacks, and 70 % accuracy for level C spoof attacks.

1. Introduction

There are two types of attendance systems, manual attendance systems, and automated attendance systems [1]. With manual attendance system, the attendance of each individual present in an event is recorded manually by someone in charge. In an automated attendance system, the tedious process of manually recording everyone's attendance is replaced with an automated system. One way to achieve this is by using an authentication method to verify and record individuals' attendance. Fingerprint, palm veins, face, and iris recognition are often used due to their false rejection rate and higher acceptance rate [2]. Face recognition is preferred over other biometric authentication methods because of its inherent benefits such as non-intrusive interaction and accessibility [1].

Biometric authentication methods are robust and convenient because it verifies the identity of the subject by using their physiological and/or behavioral characteristics and not the subject's knowledge or possession. Password-based authentication is based on knowledge, which implies that if someone knows the knowledge, they can authenticate as someone else. Ownership-based authentication is based on item possession, which means that someone who owns the object can authenticate as someone else. While password-based authentication and ownership-based authentication have their own security concerns, biometric authentication such

* Corresponding authors.
E-mail addresses: nico.surantha@binus.ac.id (N. Surantha), boy.sugijakko@binus.ac.id (B. Sugijakko).

https://doi.org/10.1016/j.iot.2024.101089

Available online 26 January 2024

as face recognition also has its own security concerns.

Face recognition, one of the biometric identification techniques, uses an analysis of an individual's distinctive facial features to confirm their identity. In most cases, this involves capturing an image or a video of the person's face, after which algorithms are used to extract and evaluate particular facial features, such the distance between the eyes, the curves of the nose, and the jawline. To see if there is a match, these features are then put in comparison with a database of recognized faces. Face recognition systems are vulnerable to face spoofing attacks, where the attacker would try to gain illegitimate access by presenting a fake face of an authorized individual [2].

Studies have shown that, because traditional face recognition does not account for an attacker, those systems are vulnerable to spoofing attacks in which an attacker impersonates an authorized individual to gain unauthorized access to a system [3]. According to Schuckers [4], face spoofing attacks could be categorized into three levels. Level A is face spoofing using a paper printout of a face image or mobile phone display of a face photo, where it requires little to no preparation by the attacker. Level B is face spoofing using masks made of paper or video of a face (moving and blinking), which requires preparation by the attacker. Level C, custom-made ultra-realistic 3D masks, where it requires preparation using specialized equipment by the attacker.

To defend against face spoofing attacks, face liveness detection could be used. Face liveness detection is a method that determines whether the presented face image is from a live or fake individual. A live face will exhibit natural movements, such as blinking, smiling, or head movements. Live faces will also show changes in skin tone or texture due to blood flow, breathing, and other physiological processes. In contrast, fake faces often lack these natural movements and physiological responses, which can be detected by face liveness detection methods. For example, a still photo or video of a face may lack blinking or other facial movements, indicating that the image is fake. Similarly, a face mask or a digital image of a face may not show changes in skin tone or texture, indicating that it is not a live individual.

In face recognition systems, face liveness detection is used so that fake images are filtered and not passed to the recognition process. Face liveness detection works by analyzing clues or presenting liveness challenges that determine whether the subject in question is live or fake. Presenting liveness challenge has several drawbacks such as it consumes time and requires interaction. There is also a liveness detection method using additional hardware such as a thermal camera to obtain additional life sign data [5].

With these problems in hand, the proposed solution is to develop a liveness detection system to improve the security of face recognition-based attendance system on a portable device using Convolutional Neural Network (CNN) to analyze clues on the presented face. Liveness challenge and additional hardware are not chosen because liveness challenge will dismiss the non-intrusive aspect of face recognition while additional hardware will make it costly and hard to implement because it requires specific hardware.

The contributions in this research are:

1. Lightweight liveness detection CNN model based on pre-trained model trained using transfer learning method
2. Lightweight face recognition-based attendance system with liveness detection

The remainder of this paper is organized as follows: Section 2 contains works on portable attendance system and face liveness detection. Section 3 discusses the proposed face liveness detection model and attendance system prototype development, while Section 4 evaluates the performance of the proposed system. Section 5 concludes the paper with conclusions and future work.

2. Related works

2.1. Portable attendance system

In 2019, Hasban et al., (2019) proposed an attendance system that utilizes Viola-Jones algorithm to detect faces and Local Binary Pattern Histogram (LBPH) method to recognize detected faces. IR camera is used to enable the system to recognize individuals in the dark. Using Raspberry Pi as the backbone for an attendance system is simple to produce, inexpensive, and appealing. In the same year, Nyein and Oo (2019) also proposed an attendance system. The attendance system proposed uses FaceNet with Support Vector Machine (SVM) instead of LBPH which achieved an accuracy of 98.66 % using the proposed method on a private dataset that contains images of the researchers' classmates and images from social media.

In 2020, Lindner et al., (2020) did a study comparing the processing time of Viola-Jones, MTCNN, and FaceNet with softmax layer as the classifier on several single-board computers. In the experiment done in this study, the average processing time on Raspberry Pi 3 B+ for Viola-Jones is 0.92 s, MTCNN is 3.08 s, and FaceNet is 0.88 s.

Table 1
Attendance system related works.

Publication	Face detection	Face recognition	Hardware	Performance
Hasban et al. [6]	Viola-Jones	LBPH	Raspberry Pi 2B	Accuracy: 56 %, Processing Time: 0.12s
Nyein and Oo [7]	Viola-Jones	FaceNet + SVM	Raspberry Pi	Accuracy: 98.66 %
Lindner et al. [8]	Viola-Jones, MTCNN	FaceNet + Softmax	Raspberry Pi 3B+	Accuracy: 98 % Processing Time: 0.92 s (Viola-Jones), 3.08 s (MTCNN), 0.88 (FaceNet)
Naufal et al. [9]	Haar Cascade	Deep learning	–	Accuracy: 95.23 %

In 2021, Naufal et al. (2021) discussed the development of a face recognition system using deep learning based on a camera. The purpose of the research is to minimize fraud and maximize accuracy in the attendance process. The system uses OpenCV with Python and a combination of Haar Cascade and deep learning to train the image database, which is then implemented into the camera for fast and efficient processing. The accuracy rate of the system is reported to be 95.23 %.

The summarized details of these related works are presented in Table 1, which provides insights into the methodologies, hardware, and performance metrics of each proposed attendance system.

2.2. Face liveness detection

In 2017, Atoum et al. proposed a method for distinguishing live and fake faces by extracting the local features and holistic depth maps from the face images. Local features are extracted from random patches of the face area and depth features are extracted from the entire face area. Local features are used to learn spoof patterns independent of spatial face area. Depth features are used to differentiate 3D and 2D objects, where 3D objects are considered live faces and 2D objects are considered fake faces (faces presented on screen or paper). From the experiment done in this study, it is concluded that combining both features provide a promising result.

Alotaibi and Mahmood (2017), proposed a method of distinguishing live and fake faces by using nonlinear diffusion based on additive operator splitting schema to get a diffused image which is then fed to a deep convolutional neural network. Nonlinear diffusion is used to obtain depth information from an image. SoftMax activation function is used as a classifier. Based on the experiments conducted in this study, it is concluded that using nonlinear diffusion to extract features from a face image is better than other hand-designed feature extraction, such as the Difference of Gaussian and Local Binary Pattern.

In 2018, Sengur et al. proposed a method to distinguish between live and fake faces by using extracting features from FC6 and FC7 layers of a pre-trained CNN model, which is then classified using SVM. The pre-trained CNN models used in this study are AlexNet and VGG16. From the experiments conducted in this study, it is concluded that promising results can be achieved using concatenated features from the fc6 layers of AlexNet and VGG16.

Rehman et al. (2018), proposed a strategy for training a CNN model with a face-spoofing dataset that has a limited amount of training samples. The strategy is a data randomization technique that continuously picks random mini batches from the entire training set on each epoch rather than randomly arranging the training set once. With the proposed strategy, training time could be reduced, and overfitting can be mitigated.

In 2019, George and Marcel proposed a method to distinguish between fake and live faces using CNN which was trained with pixel-wise binary supervision. The pixel-wise binary supervision forces the network to learn shared features from different patches of training data. From the experiment done on this study, it is concluded that using this method provides a promising result with just using a single CNN model and single frame analysis instead of multiple frames.

In 2020, Yu et al. proposed a method using central difference convolution and CNN to distinguish between live and fake faces. Central difference is introduced into 2D spatial convolution to enhance its representation and generalization ability. Central difference convolution (CDC) consists of two steps: sampling and aggregation. The sampling step is similar to that in 2D spatial convolution, while the aggregation step is different. CDC prefers to aggregate the central-oriented gradient of sampled values. To implement CDC in a modern deep learning framework, CDC is merged into 2D spatial convolution. In the experiment done in this study, it is shown that this method yields a promising result.

Singh et al. (2020), proposed a method for face liveness detection that utilizes facial thermography to perform fast and accurate analysis of temperature distribution across individual facial features [14]. Their proposed algorithm can significantly enhance the security of face recognition systems by detecting face liveness. The method was evaluated using facial thermograms of 100 subjects from an In-House dataset, with maximum reliability achieved during face liveness detection with temperature sensation. The proposed method obtained an accuracy of 96.57 %.

Table 2
Liveness detection related works.

Publication	Method	Dataset	Result		
			EER (%)	HTER (%)	Accuracy (%)
Atoum et al. [9]	Patched-based CNN + Depth-based CNN	CASIA, MSU, Replay Attack	2.67, 0.35, 0.79	2.27, 0.21, 0.72	–
Alotaibi et al. [1]	Nonlinear diffusion + CNN	NUAA, Replay Attack	–	0.98, 10	99, 16
Sengur et al. [10]	AlexNet + VGG16 + SVM	NUAA, CASIA	–	–	88.09, 94.01
Rehman et al. [11]	CNN + Data-randomization	CASIA, Replay Attack	4.59, -	4.59, 5.74	–
George et al. [12]	CNN + Pixel-wise binary supervision	Replay Mobile, OULU	0.0, -	-, 0.42	–
Yu et al. [13]	Central Difference Convolution + CNN	OULU	–	0.2	–
Singh et al. [15]	Facial thermography	In-house dataset	0,13	3.64	96.57
Mohamed et al. [16]	Sequential Deep Learning	CelebA-Spoof	–	–	87

In 2021, Mohammed et al. proposed a method to distinguish between fake and live faces using sequential deep learning architecture in which convolution and pooling layers are stacked from input to output. While training the CNN, data augmentation such as shear range, zoom range, and horizontal flip is performed on the training dataset. Dataset used to train the CNN is CelebA-Spoof dataset, which is larger than any existing dataset related to this field in 2021. With the experiment done in this study, the proposed CNN approach achieved a relatively acceptable accuracy for testing which is 87 %.

The summarized details of these related works are presented in Table 2, which provides insights into the methodologies, dataset, and performance metrics of each proposed methods.

Based on the literature review on the previously proposed attendance system, the security of attendance systems could be improved by implementing liveness detection before authenticating and recognizing the faces of the subjects. For liveness detection, it can be concluded that CNN based method could be used to classify live and fake faces with a promising result. Therefore, in this study, a lightweight CNN liveness detection model that can be implemented on an attendance system using Raspberry Pi will be proposed. For the face detection and face recognition step, Viola-Jones and MobileFaceNet could be used as the two methods are proven to be accurate and can be run on a portable device.

3. Theory and methods

The implementation of this research is divided into three main stages: the planning stage, the model development stage, and the evaluation stage, as shown by Fig. 1. In the first stage, background literature review, problem identification, and research's goals and scope of work outlining are performed. The main problem of this research is the need of implementing liveness detection in face recognition-based attendance systems to prevent face spoofing attacks. The goal of this research is to develop a face recognition-based attendance system with liveness detection that can perform well on portable devices, such as Raspberry Pi. To achieve that, a literature review on attendance systems and liveness detection was required.

The researcher then obtains the necessary dataset to train a CNN model during the model development step. CelebA-Spoof and NUAA Imposter datasets were obtained. Image processing is required for these dataset's pictures to be useful. Cropped and aligned photos from these databases are produced to provide images of aligned faces. After the datasets have been processed, a CNN model is trained using a technique known as transfer learning. Transfer learning is the process of re-training an existing pre-trained model for a different use case. Then the trained model is tested and evaluated using the confusion matrix, Equal Error Rate (EER), Half Total Error Rate (HTER), and F1 Score.

In the last part of this research, requirements for an attendance system are obtained, and an attendance system with an additional security layer, face liveness detection, is created utilizing the requirements gathered. The gathered are adequate accuracy, low processing time, and the ability to prevent face spoofing. The overall system performance is then evaluated and tested by computing the processing speed.

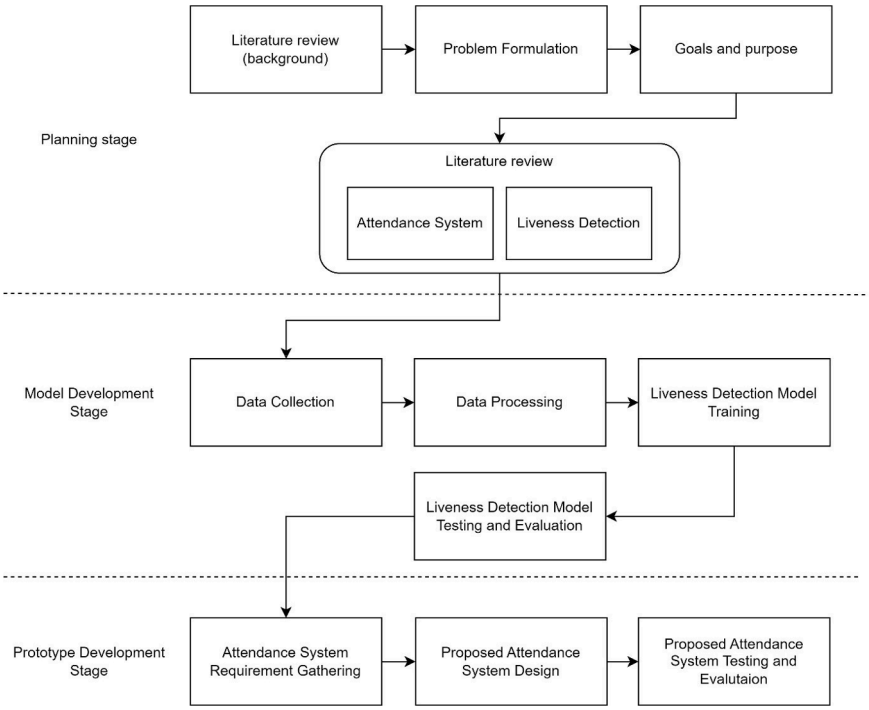


Fig. 1. Research stages.

4. Model development stage

4.1. Data collection

Two publicly available datasets were obtained to be used as training and evaluation data for the proposed liveness detection CNN model. The obtained datasets are CelebA-Spoof and NUAA imposter datasets. CelebA-Spoof dataset was chosen because it is publicly available and it is the largest dataset available in this field according to Mohamed [16], while NUAA imposter dataset was chosen because it is publicly available and has been used in several previous studies.

4.2. Data processing

The obtained datasets must first be processed before they can be utilized to train and evaluate the proposed liveness detection CNN model. Images from these datasets were processed by cropping and aligning so that only images of aligned faces are retained and images with no face detected will be discarded. To detect and align faces on these images, RetinaFace model from a Python library for face recognition named DeepFace [17] was used. Afterward, these images from each dataset were split into train, test, and validation.

4.3. Liveness detection model training

A technique called transfer learning is utilized to train the liveness detection CNN model. Transfer learning is a training method to reuse a pre-trained model, which is a model that has been trained using another dataset, on a new problem. Transfer learning is used because it has an advantage where it requires fewer data and resources compared to training a new model from scratch. Several pre-trained models were used to carry out the experiment in this paper, including models trained for object recognition and face recognition. When selecting the pre-trained model, the model's speed and weight were considered because it must be able to run on a device with limited computational power. Keras, an open-source software library that provides a Python interface for artificial neural networks, provides several pre-trained models that can be used.

From the available pre-trained models on Keras (Table 3.1), MobileNetV2 was chosen because it is the smallest-sized model with decent accuracy. The MobileNetV2 model was trained to be used for object recognition. Because the proposed liveness detection model will be used to differentiate between fake and live faces, pre-trained models trained for face recognition were also considered. FaceNet and MobileFaceNet were chosen because FaceNet has previously been shown to be capable of running on the Raspberry Pi, and MobileFaceNet is the lightweight version of FaceNet.

There are various steps in the model training process. The pre-trained models were loaded first, and then their classification layer was removed. Then, additional layer(s) were added to these pre-trained models. New layers were added on top of the pre-trained models to enable the model to learn new features that will help differentiate between live and fake face images. The pre-trained models, which were trained using their respective datasets, were used as feature extractors and the new layers were trained to learn how to interpret the extracted features on a new dataset. Afterward, previously acquired datasets were loaded, and the images were resized according to each model's input size. To load the datasets, Keras' ImageDataGenerator class was used because it enables loading data into batches, which solves the problem of huge datasets not being able to fit into memory. Next, the pre-trained models were trained using Keras' fit method. Finally, the models were saved after training to be evaluated.

4.4. Liveness detection model testing and evaluation

After the liveness detection model has been trained, the testing and evaluation process is done. To test and evaluate the performance of the model, several metrics are used, such as Equal Error Rate (EER), Half Total Error Rate (HTER), and F1 Score. After obtaining the score for each metric mentioned before, the scores will then be compared against scores from previous studies to determine whether the performance of the proposed model is adequate or not.

The final stage determines the performance and success rate of the system that has been created. Error Rate (EER), Half Total Error Rate (HTER), and F1 Score will be used to assess the result. EER was calculated using formula 3.3, HTER using formula 3.4, and F1 score using formula 3.5. Previous research used these evaluation metrics; therefore, these metrics were chosen to be the evaluation metrics to compare the results.

$$\mathrm{FAR} = \frac{\mathrm{FP}}{\mathrm{FP} + \mathrm{TN}} \tag{3.1}$$

Table 3
Keras pre-trained models [16].

Model	Size (MB)	Top-1 accuracy	Top-5 accuracy	Parameters (Millions)
MobileNetV2	14	71.3 %	90.1 %	3.5
MobileNet	16	70.4 %	89.5 %	4.3
NASNetMobile	23	74.4 %	91.9 %	5.3
EfficientNetB0	29	77.1 %	93.3 %	5.3
EfficientNetV2B0	29	78.7 %	94.3 %	7.2

$$\text{FRR} = \frac{\text{FN}}{\text{FN} + \text{TP}} \tag{3.2}$$

$$\text{Equal Error Rate} = |\text{FAR} - \text{FRR}| \tag{3.3}$$

$$\text{Half Total Error Rate} = \frac{\text{FRR} + \text{FAR}}{2} \tag{3.4}$$

$$\text{F1 Score} = \frac{\text{TP}}{TP + \frac{1}{2}(FP + FN)} \tag{3.5}$$

Where:
TP = True Positive
TN = True Negative
FP = False Positive
FN = False Negative
FAR = False Acceptance Rate
FRR = False Rejection Rate

5. Prototype development stage

5.1. Attendance system requirement gathering

Before designing an attendance system, requirements must be gathered beforehand to ensure the proper functionality of the system. The requirements are gathered by observing the attendance system used at Bina Nusantara University, which utilizes a Radio Frequency Identification (RFID) card and reader as a way to authenticate students' attendance [18]. Because RFID attendance systems accurately identify individuals based on the data stored in the RFID card, fraud is a risk if the card is in the possession of another person. When a large number of students arrive at class at the same time, sometimes there is a line in front of the RFID reader since the attendance procedure takes some time to process. From the observation, it is concluded that for an attendance system to work properly, it must have adequate accuracy in recognizing individuals, have low processing time, and have a security feature to prevent fraud.

5.2. Proposed attendance system design

After obtaining the requirements for an attendance system, the proposed attendance system is designed. To fulfil the requirements, the previously trained liveness detection model is implemented into a conventional attendance system which is an attendance system with no face spoofing prevention functionality. The general steps of an attendance system are face detection, face recognition, and attendance recording. The proposed attendance system adds a liveness detection step before face recognition to prevent fake faces from being recognized and recorded by the system.

On the proposed attendance system with liveness detection, Viola-Jones Algorithm is used for face detection and MobileFaceNet is used for face recognition. Viola-Jones Algorithm is used for face detection because it has a low processing time, which is important considering the system will be run on a portable device, and the limitation where it is only good at detecting frontal face is not an issue as it is required for the subject to face directly at the camera. MobileFaceNet is used for face recognition because it is a highly efficient

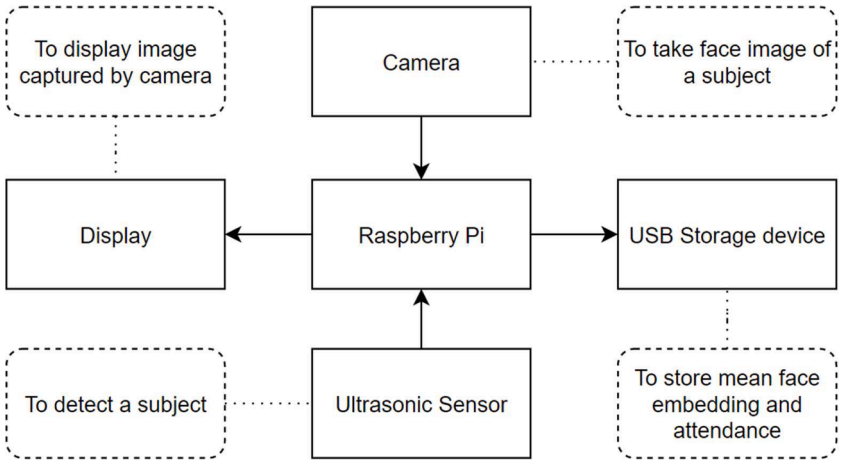


Fig. 2. Proposed system design.

CNN model that was created specifically for high-precision real-time face recognition on portable devices. With a model that was only 4 MB in size, MobileFaceNet was able to attain a fairly comparable accuracy to that of other heavier models such as FaceNet [19].

The Euclidian distance between the presented subject's face embedding will be compared against a prepared dataset of registered subjects to recognize the presented subject. To handle unknown subjects, a distance threshold was utilized. If the closest Euclidean distance between the subject's face embedding and stored face embeddings is greater than the specified threshold, the subject will be categorized as unknown. The Euclidean distance threshold used in this study is obtained from DeepFace [17] Python library.

The design of the proposed system is shown in Fig. 2, where several components are used. The Raspberry Pi 4B 8GB is the key component utilized to analyze data from the sensors. When a person approaches the system, an ultrasonic sensor determines when the system should begin the facial recognition process. The camera is used to capture photographs of the subject's face. A USB storage device is used to store data of individuals whose attendance will be recorded.

Before being able to recognize subjects, face images of each subject must be stored to be compared against. The process of registering a new subject is shown in Figure 3.5. USB camera captures images and if a face is detected, the system will store several face

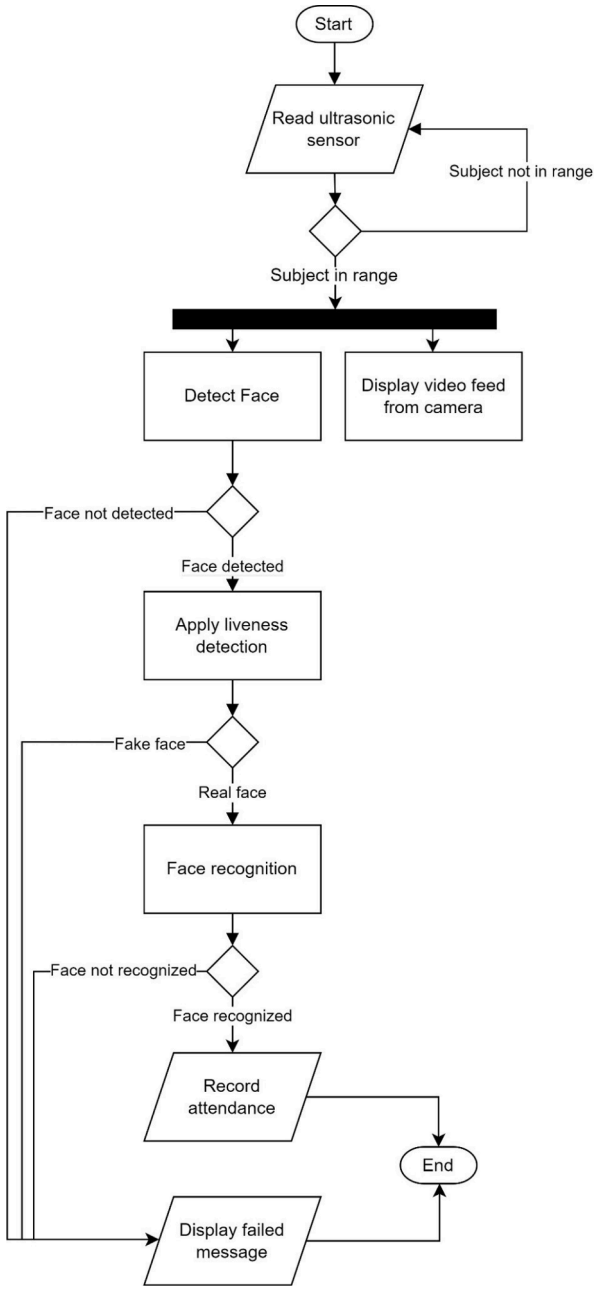


Fig. 3. Attendance system flowchart.

images of the subject and stores them in a USB storage device. The stored face images will then be processed by calculating the mean of every face embedding obtained from every face image using MobileFaceNet. The result of calculating mean face embeddings will be stored on the USB storage device as a pickle file.

The way the attendance system works as a whole can be seen in Fig. 3. The initial stage in this system's function is to determine whether or not there is a subject in front of it using an ultrasonic sensor. Only if a subject is detected by the ultrasonic sensor will the face recognition process begin. When the system determines that a subject is in range, the display will show live video from the USB camera and the face detection process will start. If a face is detected, the liveness detection method is used to determine whether the observed face is fake or live. Fake faces will be rejected, and the system will restart from the beginning. Live faces will be accepted, and the system will recognize and record the individual's attendance.

5.3. Proposed attendance system testing and evaluation

To test and evaluate the proposed attendance system, several images and videos were collected personally by the researcher. These images and videos were then used to test the performance of the proposed attendance system in real life environment. To test the proposed attendance system, level A and level B spoof samples were fabricated from the collected images. The images for level A spoof were transferred to a phone and printed on paper, whereas the images for level B spoof were printed and turned into paper mask cut-outs, and videos were transferred to a phone. Afterward, the proposed attendance system would be presented with live, level A spoof, and level B spoof samples. The proposed attendance system performance will then be evaluated based on the accuracy acquired.

6. Results and discussion

6.1. Liveness detection model training

Table 4 shows the accuracy of MobileNetV2, MobileFaceNet, and FaceNet models with several variations of additional layers added on top that has been trained using 1 × 10–5 learning rate for a maximum of 100 epochs and early stopping if there were no validation accuracy improvement for 5 epochs. The variations are:

1. Variation A, with no additional layers, added to the pre-trained model
2. Variation B, with one stack of 128-unit dense layer and 20 % dropout rate Dropout layer added to the pre-trained model
3. Variation C, with one stack of 256-unit dense layer and 50 % dropout rate Dropout layer added to the pre-trained model
4. Variation D, with two stacks of 256-unit dense layer and 50 % dropout rate Dropout layer added to the pre-trained model
5. Variation E, with three stacks of 256-unit dense layer and 50 % drop-out rate Dropout layer added to the pre-trained model

For MobileNetV2, the configuration that obtains the highest accuracy for CelebA-Spoof dataset is variation C while for NUAA dataset, the best configuration is variation E. For MobileFaceNet, the configuration that obtains the highest accuracy for CelebA-Spoof dataset is variation B while for NUAA dataset, the best configuration is variation C. For FaceNet, the configuration that obtains the highest accuracy for CelebA-Spoof dataset is variation C while for NUAA dataset, the best configuration is both variation C and variation E. From the results above, it can be concluded that adding one stack of Dense layer with 256 units and a Dropout layer with a 50 % drop-out rate yields the best result.

Figs. 4–6 show the accuracy and loss curve while training the models on CelebA-Spoof dataset using variation C. Validation accuracy tends to be higher and validation loss tends to be lower than the training counterparts because dropout layer was used, and this behavior is as expected. From these three graphs, it can be seen that all three models have reached their best fit as the accuracy did not

Table 4
Additional layer variations accuracy comparison.

Variation	New layer	Model	Test accuracy (%)	
			CelebA-Spoof	NUAA
A	None	MobileNetV2	88.85	75.74
		MobileFaceNet	76.54	94.62
		FaceNet	79.01	97.92
B	1 x (Dense(128) +Dropout(0.2))	MobileNetV2	89.55	69.51
		MobileFaceNet	81.55	98.28
		FaceNet	81.58	99.51
C	1 x (Dense(256) +Dropout(0.5))	MobileNetV2	92.02	76.05
		MobileFaceNet	81.68	99.26
		FaceNet	81.76	99.81
D	2 x (Dense(256) +Dropout(0.5))	MobileNetV2	92.00	70.80
		MobileFaceNet	80.02	98.90
		FaceNet	80.42	99.75
E	3 x (Dense(256) +Dropout(0.5))	MobileNetV2	88.45	76.42
		MobileFaceNet	80.37	98.65
		FaceNet	81.14	99.81

start to decrease and the loss did not start to increase at the end of the training.

From the results of the experiment, as seen in Table 5 and Fig. 7, the accuracy of MobileNetV2 is high on CelebA-Spoof dataset and low on NUAA dataset while Mobile FaceNet and FaceNet have high accuracy on NUAA dataset and low on CelebA-Spoof dataset. Because MobileFaceNet is the lighter version of FaceNet, it is assumed to be similar to FaceNet in performance. To figure out why MobileNetV2 performs better on CelebA-Spoof dataset while MobileFaceNet and FaceNet perform better on NUAA dataset, a new dataset, called CelebA-Spoof mini, was created by limiting the number of images on CelebA-Spoof dataset to be equal to the number of images on NUAA dataset. By creating this new dataset, the difference in the number of images on each dataset is eliminated, and the only difference between these two datasets is the range of variation of the images.

From these results, it can be concluded that both MobileFaceNet and FaceNet perform well on test images with a smaller range of variation while MobileNetV2 can handle a wider range of test image variation. Based on these results, MobileNetV2 is chosen for this research because it is the most suitable model for real-life environment usage as it can handle a wider range of variation.

6.2. Liveness detection model evaluation

To have a fair performance comparison, both CelebA-Spoof and NUAA datasets used for testing have been modified to be as similar as possible to test datasets used in previous studies. For the CelebA-Spoof dataset, 200 images were used, equally divided between live and spoof, while the NUAA dataset used 597 live images and 1040 spoof images.

Table 6 illustrates that the proposed method achieved superior performance with an accuracy and F1 Score of 92.02 % and 94.5 %, respectively. In comparison, the Sequential Deep Learning method by Mohamed et al. [16] demonstrated an accuracy of 87 % and an identical F1 score of 94.5 %. While in Table 7, it can be seen that the proposed method had a lower accuracy of 76.05 % compared to an accuracy of 88.09 % from AlexNet + VGG16 + SVM method by Sengur et al. [11] and 99 % from Nonlinear diffusion + CNN method by Alotaibi and Mahmood [1]. Despite having a lower accuracy on NUAA dataset, the proposed method had an HTER of 0.13 % which is lower than the HTER of Nonlinear diffusion + CNN method by Alotaibi and Mahmood [1], which is 0.98 %.

The comparison made between the proposed method and methods from previous studies on CelebA-Spoof and NUAA datasets shows that on CelebA-Spoof dataset, the proposed method has a higher accuracy, and on NUAA dataset, even though the proposed method's accuracy is lower, it has a better HTER score. On CelebA-Spoof dataset, the number of images used for testing is the same as on the previous study by Mohamed et al. [16], which is 200 images equally divided between live and spoof. While, on NUAA dataset, the number of images used for testing is not the same as previous studies by Sengur et al. [11]. and Alotaibi and Mahmood [1], which were 5761 and 9123, respectively.

Difference in the number of images used for testing reduces the validity of the comparison and because there are only a limited number of images available for training, the proposed model's performance might be affected on the NUAA dataset. The CelebA-Spoof dataset contains images with a broader range of conditions than the NUAA dataset, and it more accurately represents the state of the real-world environment. Therefore, with the increase of performance on CelebA-Spoof dataset, the proposed model is expected to perform better on real-world environment.

6.3. Proposed attendance system prototype

The assembled proposed attendance system prototype could be seen in Fig. 8. The ultrasonic sensor was connected via General-purpose input/output (GPIO) pins on the Raspberry Pi using a connection circuit as seen in Fig. 9. The assembled connection circuit works as a voltage divider, which reduces the 5 Volts output of the ultrasonic sensor to a value that is safe for the Raspberry Pi's GPIO pins which work on 3.3 Volts.

This prototype used a USB camera with a maximum resolution of 1920 × 1080 pixels, a frame rate of 30 frames per second (FPS), and a 110-degree angle of view. The USB storage device used was a 16GB USB flash drive, despite the model's total size being only 14

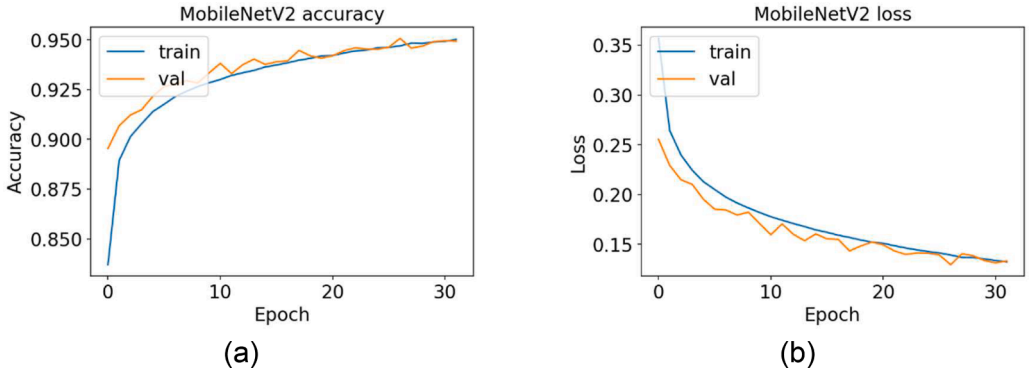


Fig. 4. (a) Training and validation accuracy graph (b) Training and validation loss graph of celeba-spoof using variation C MobileNetV2. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

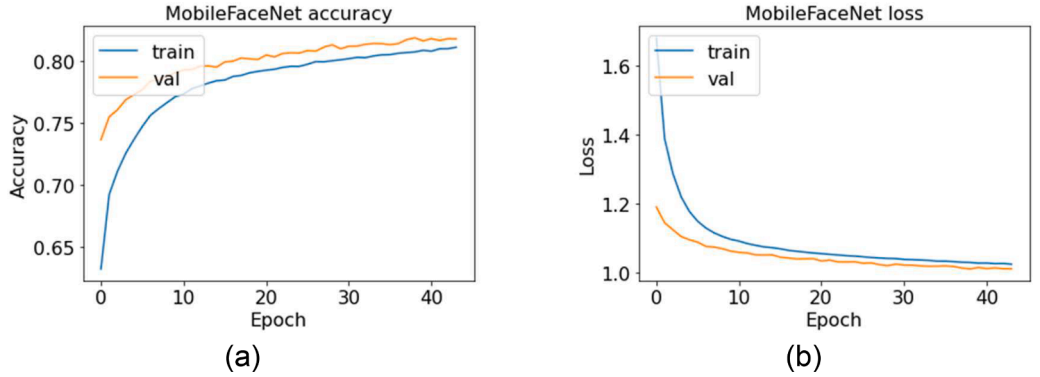


Fig. 5. (a) training and validation accuracy graph (b) Training and validation loss graph of CelebA-Spoof using variation C MobileFaceNet. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

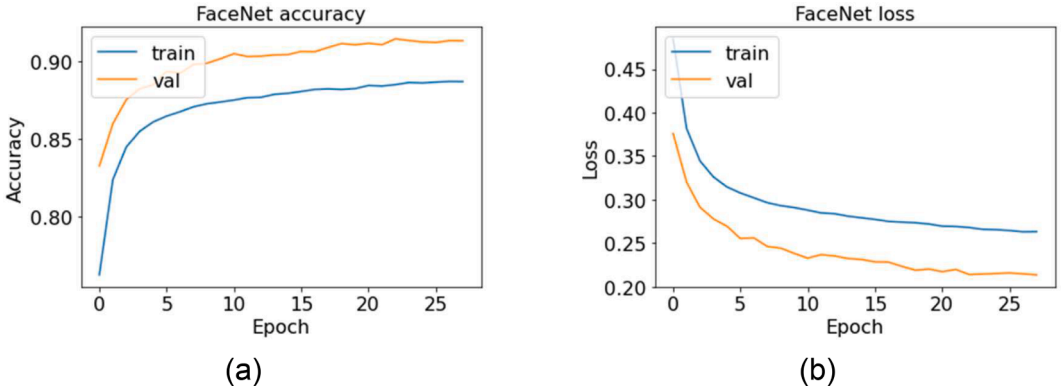


Fig. 6. (a) Training and validation accuracy graph (b) training and validation loss graph of CelebA-Spoof using variation C FaceNet. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

Table 5
Models accuracy and average processing time comparison.

Model	Average processing time	Test Accuracy (%) CelebA-Spoof	NUAA	CelebA-Spoof mini
MobileNetV2	0.30s	92.02	76.05	85.33
MobileFaceNet	0.27s	81.68	99.26	73.30
FaceNet	0.43s	81.76	99.81	78.55

Megabytes. As for the display, a five-inch, 400 × 800-pixel display was used. This display is connected to the Raspberry Pi using High-Definition Multimedia Interface (HDMI) and is powered by a USB connection to the Raspberry Pi.

An experiment was then conducted to determine the ideal distance threshold for the prototype. The experiment's findings, which are presented in Table 8, indicate that a distance of 50–75 cm is ideal for the ultrasonic sensor to initiate the attendance process. In the experiment, various presentation styles were tested, including live faces, Level A spoofs, Level B spoofs, and Level C spoofs. For the evaluation of live faces, five different individuals were included. For Level A spoofs, five paper printouts of a face image and five face photos displayed on a mobile device were tested. For Level B spoofs, five paper masks and five videos of moving and blinking faces were evaluated. Finally, a realistic latex face mask was used to evaluate Level C spoofs.

An accuracy and F1 Score of 0 % was obtained for distances under 25 cm because the camera's captured face image was cropped due to the face being too close to the camera. For distances over 25 cm and under 50 cm, an accuracy of 80 % and F1 Score of 85 % was attained with five false level A spoof attack predictions. An accuracy of 92 % and F1 Score of 94 % was attained for distances greater than 50 cm and less than 75 cm with two incorrect predictions of a level A spoof attack. For distances over 75 cm and under 100 cm, accuracy was 88 % and F1 Score was 92 % with three incorrect predictions of a live subject. For distances greater than 100 cm, an accuracy of 72 % and F1 Score of 83 % was obtained with five incorrect predictions of live subject and two incorrect predictions of level A spoof attack.

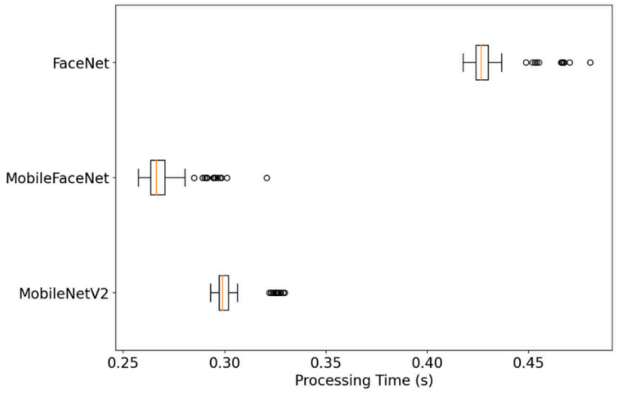


Fig. 7. Models processing time box plot. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

Table 6
Results comparison with CelebA-Spoof Dataset [20].

Method	ACC (%)	EER (%)	HTER (%)	F1 Score (%)
Sequential deep learning [16]	87	–	–	86.8
Transfer learning on mobileNetV2 (**Proposed**)	**92.02**	**0.69**	**0.41**	**94.5**

Table 7
Results comparison with NUAA Dataset [21].

Method	ACC (%)	EER (%)	HTER (%)	F1 Score (%)
AlexNet + VGG16 + SVM [11]	88.09	–	–	–
Nonlinear diffusion + CNN [1]	**99**	–	0.98	–
Transfer learning on mobileNetV2 (**Proposed**)	76.05	**0.27**	**0.13**	**84.1**

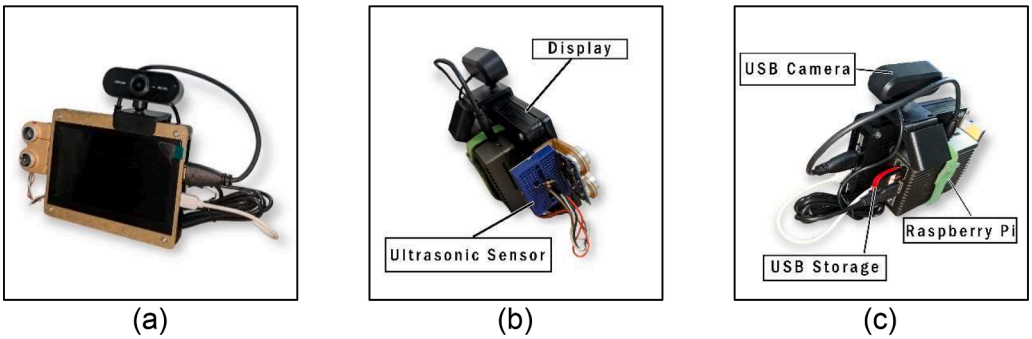


Fig. 8. Assembled prototype (a) Front-side view (b) Right-side view (c) Left-side view. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

6.4. Proposed attendance system evaluation

On the prototype for the proposed attendance system, several presentation types were tested, including live subjects, pictures on display, prints on regular paper, prints on glossy paper, paper masks, videos on display, and realistic latex face mask. For each presentation type, 20 attempts were made. According to the findings presented in Table 9 of the experiment, the accuracy and F1 Score for live subjects were 96 % and 97 %, respectively. For level A spoof, which included a picture on display and a printed picture, the accuracy and F1 Score were 79 % and 84 %. For level B spoof, involving a paper mask and a video on display, the corresponding values were 83.7 % and 87.5 %. Additionally, the experiment demonstrated that Level C spoof, which utilized a realistic latex face mask, achieved an accuracy of 70 % and an F1 Score of 82 %.

The prototype had difficulty distinguishing between live and fake faces that were printed on regular paper, resulting in worse performance on Level A spoof than on Level B spoof. This is due to the fact that the texture of the printed image on regular paper is

(a) Front-side view (b) Right-side view (c) Left-side view

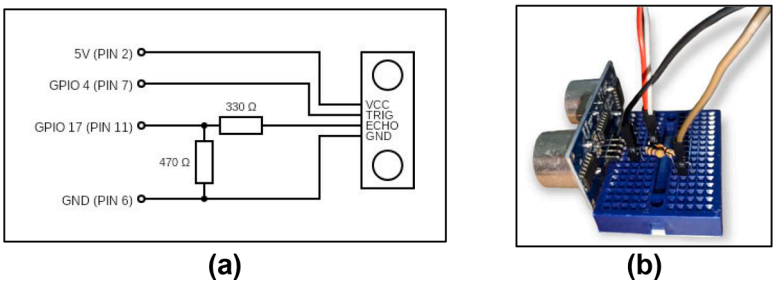


Fig. 9. (a) Circuit diagram (b) Ultrasonic sensor connection. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

Table 8
Prototype accuracy with different ultrasonic sensor distance threshold.

	$x \leq 25$ cm	$25 < x \leq 50$ cm	$50 < x \leq 75$ cm	$75 < x \leq 100$ cm	$x > 100$ cm
Accuracy	0 %	80 %	92 %	88 %	72 %
F1 Score	0 %	85 %	94 %	92 %	83 %

Table 9
Prototype accuracy.

Presentation type	Accuracy (%)	F1 Score (%)	Average processing time (seconds)
Live	96	97	0.56
Picture on display	82	86	0.48
Printed picture (normal paper)	70	78	0.51
Printed picture (glossy paper)	85	88	0.57
Paper mask	87	90	0.49
Video on display	80	85	0.51
Realistic latex mask	70	82	0.49

similar to that of a live face and lacks reflections, making it harder to differentiate between them. It should be noted that Level C spoof attacks are expected to perform even worse than Level A and Level B because the realistic mask used in Level C spoof attacks closely resembles a real human face, making it more difficult for the system to detect the difference between a live face and a face covered by a realistic mask. Nonetheless, the entire process of liveness detection, face detection, face recognition, and attendance taking took less than 0.6 s.

From The prototype had difficulty distinguishing between live and fake faces that were printed on regular paper, resulting in worse performance on Level A spoof than on Level B spoof. This is due to the fact that the texture of the printed image on regular paper is similar to that of a live face and lacks reflections, making it harder to differentiate between them. It should be noted that Level C spoof attacks are expected to perform even worse than Level A and Level B because the realistic mask used in Level C spoof attacks closely resembles a real human face, making it more difficult for the system to detect the difference between a live face and a face covered by a realistic mask. Nonetheless, the entire process of liveness detection, face detection, face recognition, and attendance taking took less than 0.6 s.

Table 10, it can be seen that even after adding an additional liveness detection step, the proposed system has a faster processing time compared to previous studies except for Viola-Jones and LBPH combination on Raspberry Pi 2B. Even though a better version of the Raspberry Pi was used in this study, the processing time achieved was significantly faster, more than twice as fast as in several previous studies. Therefore, adding a liveness detection step to a face recognition-based attendance system running on a Raspberry Pi has a negligible impact on processing time. Adding liveness detection to a face recognition-based attendance system should always be

Table 10
Comparison of processing time.

Device	Face detection	Face recognition	Liveness detection	Processing time (seconds)
Raspberry Pi 4B [22]	Histogram of Oriented Gradients	Dlib face recognition	No	1.96
Raspberry Pi 2B [6]	Viola-Jones	LBPH	No	0.12
Raspberry Pi 3B+ [8]	Viola-Jones, MTCNN	FaceNet + Softmax	No	1.80
Raspberry Pi 4B **(proposed)**	Viola-Jones Algorithm	MobileFaceNet	Yes	0.52

considered for security reasons.

7. Conclusions

Face recognition systems, especially those that do not implement liveness detection, are vulnerable to face spoofing attacks where attackers would try to gain access by presenting a fake face of another individual. On face recognition-based attendance systems, liveness detection is used to prevent individual attendance that is not really present to be taken. Attendance systems typically run on a portable device as it is more practical and efficient because it could be relocated to any appropriate location as needed. Having a limitation of computing power on portable devices, implementing liveness detection will be a challenge as the liveness detection method that will be implemented must be lightweight so the attendance system could still be run on portable devices. By adding a liveness detection step, the processing time of each face presented to the system will be impacted.

In this study, several pre-trained CNN models were used to perform experiments. The pre-trained models used were pre-trained models trained for face recognition and object recognition. Transfer learning with several variations of new layers was performed on MobileNetV2, FaceNet, and MobileFaceNet. From the results of the experiment, variation C obtained the best result. While MobileFaceNet has a lower average processing time, MobileNetV2 was chosen because the average processing time difference is small, has better accuracy on CelebA-Spoof dataset which has a larger range of variations and has decent accuracy on NUAA dataset.

On the proposed attendance system prototype, several threshold distances for the ultrasonic sensors were evaluated. The prototype obtained the best accuracy on a distance of 50 cm to 75 cm. A distance below 50 cm has lower accuracy because spoof attacks were detected as live faces and a distance above 75 cm has lower accuracy because live faces were detected as fake faces. With the obtained optimal distance, the prototype accuracy was evaluated. The proposed attendance system prototype obtained an accuracy of 96 % and F1 Score of 97 % for live subjects, 79 % and 84 % for level A spoof attacks, 83.7 % and 87.5 % for level B spoof attacks, and 70 % and 89 % for level C spoof attacks. Further work includes research on liveness detection with occluded faces and/or low illumination conditions.

CRediT authorship contribution statement

Nico Surantha: Conceptualization, Funding acquisition, Supervision, Writing – original draft, Writing – review & editing. **Boy Sugijakko:** Data curation, Formal analysis, Investigation, Project administration, Validation, Writing – review & editing.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

Data will be made available on request.

References

[1] A. Alotaibi, A. Mahmood, Deep face liveness detection based on nonlinear diffusion using convolution neural network, Signal. Image Video Process.. 11 (2017) 713–720. https://doi.org/10.1007/s11760-016-1014-2.
[2] L. Li, X. Feng, Z. Boulkenafet, Z. Xia, M. Li, A. Hadid, An original face anti-spoofing approach using partial convolutional neural network, in: 2016 Sixth International Conference on Image Processing Theory, Tools and Applications (IPTA), IEEE, 2016, pp. 1–6, https://doi.org/10.1109/IPTA.2016.7821013.
[3] P.P.K. Chan, W. Liu, D. Chen, D.S. Yeung, F. Zhang, X. Wang, C.-C. Hsu, Face liveness detection using a flash against 2D Spoofing Attack, IEEE Trans. Inf. Forens. Sec. 13 (2018) 521–534, https://doi.org/10.1109/TIFS.2017.2758748.
[4] S. Schuckers, Presentations and attacks, and spoofs, oh my, Image Vis. Comput. 55 (2016) 26–30, https://doi.org/10.1016/j.imavis.2016.03.016.
[5] L. Song, C. Liu, Face liveness detection based on joint analysis of RGB and near-infrared image of faces, Electronic Imaging (2018) 373, https://doi.org/10.2352/ISSN.2470-1173.2018.10.IMAWM-373, 2018.
[6] A.S. Hasban, N.A. Hasif, Z.I. Khan, M.F. Husin, N.E.A. Rashid, K.K.M. Sharif, N.A. Zakaria, Face recognition for Student Attendance using Raspberry Pi, in: 2019 IEEE Asia-Pacific Conference on Applied Electromagnetics (APACE), IEEE, 2019, pp. 1–5, https://doi.org/10.1109/APACE47377.2019.9020758.
[7] T. Nyein, A.N. Oo, University classroom attendance system using facenet and support vector machine, in: 2019 International Conference on Advanced Information Technologies (ICAIT), IEEE, 2019, pp. 171–176, https://doi.org/10.1109/AITC.2019.8921316.
[8] T. Lindner, D. Wyrwal, M. Bialek, P. Nowak, Face recognition system based on a single-board computer, in: 2020 International Conference Mechatronic Systems and Materials (MSM), IEEE, 2020, pp. 1–6, https://doi.org/10.1109/MSM49833.2020.9201668.
[9] G.R. Naufal, R. Kumala, R. Martin, I.T.A. Amani, W. Budiharto, Deep learning-based face recognition system for attendance system, ICIC Exp. Lett. Part B: Appl. 12 (2021) 193–199.
[10] Y. Atoum, Y. Liu, A. Jourabloo, X. Liu, Face anti-spoofing using patch and depth-based CNNs, in: 2017 IEEE International Joint Conference on Biometrics (IJCB), IEEE, 2017, pp. 319–328, https://doi.org/10.1109/BTAS.2017.8272713.
[11] A. Sengur, Z. Akhtar, Y. Akbulut, S. Ekici, U. Budak, Deep feature extraction for face liveness detection, in: 2018 International Conference on Artificial Intelligence and Data Processing (IDAP), IEEE, 2018, pp. 1–4, https://doi.org/10.1109/IDAP.2018.8620804.
[12] Y.A.U. Rehman, L.M. Po, M. Liu, LiveNet: improving features generalization for face liveness detection using convolution neural networks, Expert. Syst. Appl. 108 (2018) 159–169, https://doi.org/10.1016/j.eswa.2018.05.004.
[13] A. George, S. Marcel, Deep pixel-wise binary supervision for face presentation attack detection, in: 2019 International Conference on Biometrics (ICB), IEEE, 2019, pp. 1–8, https://doi.org/10.1109/ICB45273.2019.8987370.

[14] M. Singh, A.S. Arora, Computer aided face liveness detection with facial thermography, Wirel. Pers. Commun. 111 (2020) 2465–2476, https://doi.org/10.1007/s11277-019-06996-6.
[15] Z. Yu, C. Zhao, Z. Wang, Y. Qin, Z. Su, X. Li, F. Zhou, G. Zhao, Searching central difference convolutional networks for face anti-spoofing, in: 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), IEEE, 2020, pp. 5294–5304, https://doi.org/10.1109/CVPR42600.2020.00534.
[16] A.A. Mohamed, M.M. Nagah, M.G. Abdelmonem, M.Y. Ahmed, M. El-Sahhar, F.H. Ismail, Face liveness detection using a sequential CNN technique, in: 2021 IEEE 11th Annual Computing and Communication Workshop and Conference (CCWC), IEEE, 2021, pp. 1483–1488, https://doi.org/10.1109/CCWC51732.2021.9376030.
[17] S.I. Serengil, A. Ozpinar, LightFace: a hybrid deep face recognition framework, in: 2020 Innovations in Intelligent Systems and Applications Conference (ASYU), IEEE, 2020, pp. 1–5, https://doi.org/10.1109/ASYU50717.2020.9259802.
[18] S. Kurniali Mayliana, The development of a web-based attendance system with RFID for higher education institution in Binus university, EPJ. Web. Conf. 68 (2014) 00038, https://doi.org/10.1051/epjconf/20146800038.
[19] S. Chen, Y. Liu, X. Gao, Z. Han, MobileFaceNets: efficient CNNs for accurate real-time face verification on mobile devices, in: 2018: pp. 428–438. https://doi.org/10.1007/978-3-319-97909-0_46.
[20] Y. Zhang, Z. Yin, Y. Li, G. Yin, J. Yan, J. Shao, Z. Liu, CelebA-Spoof: large-scale face anti-spoofing dataset with rich annotations, in: 2020: pp. 70–85. https://doi.org/10.1007/978-3-030-58610-2_5.
[21] X. Tan, Y. Li, J. Liu, L. Jiang, Face liveness detection from a single image with sparse low rank bilinear discriminative model, in: 2010: pp. 504–517. https://doi.org/10.1007/978-3-642-15567-3_37.
[22] R.H.A. Jr, J.G. Bernadas, A.F.A. Pangandaman, L.C. Borja, Automated attendance system using RFID, face recognition, and SMS, (n.d.). https://doi.org/10.4108/eai.7-12-2021.2314721.