Journal of Network Intelligence
Taiwan Ubiquitous Information

Volume **10**, Number **1**, February **2025**

Automobile Entry System Based on Face Recognition

Weidong Fang, Zuoxi Yu*
School of Electronic, Electrical Engineering and Physics
Fujian Provincial Key Laboratory of Automotive Electronics and Electric Drive
Fujian University of Technology, Fuzhou 350118, China
wdfang@126.com, zuoxiyu122@163.com

Jeng-Shyang Pan
College of Computer Science and Engineering
Shandong University of Science and Technology, Qingdao 266590, China

Chaoyang University of Technology, Taichung 413310, Taiwan
jengshyangpan@gmail.com

*Corresponding author: Zuoxi Yu

Received October 20, 2023, revised January 30, 2024, accepted April 7, 2024.

Abstract. *With the development of technology, the automobile has become an indispensable part of people's daily lives. People's needs for automobile entry systems have also changed, in automobile safety and ease of use have become more and more important. In recent years, face recognition technology has made significant progress, and face recognition technology has been widely used in various fields, especially face recognition based on deep learning has great advantages in accuracy, recognition speed, and security, and can provide a more secure and reliable way of identity verification. Traditional automobile entry systems usually use mechanical keys and remote control keys, which do not remove the key and have certain shortcomings in security and user experience. Face recognition-based car entry systems can make up for these shortcomings and provide a more convenient and intuitive user experience. Combining face recognition with automobiles is also a hot topic in current scientific research. In this paper, a set of automobile entry systems with high efficiency and security is designed according to the face recognition method research, using three deep learning models: face detection, live body detection, and face recognition. In face detection, the RetinaFace lightweight model is used the network structure is improved, and the detection speed is increased by 14.9%. For face live detection and face recognition, the MobileFaceNet lightweight network is used as the base network for live detection and face recognition, achieving a 98.9% accuracy rate on the CelebA Spoof live detection dataset. In face recognition, feature extraction is performed on the detected faces after face detection, and the recognition results are output by comparing with the recorded faces. Improvements to its network improved the recognition accuracy by 0.18%, 0.77%, and 0.73% on the LFW, CFP_FP, and AgeDB30 datasets, respectively. The model was deployed on Raspberry Pi and connected to CANoe via CAN bus to realize the unlocking of car face recognition.*
Keywords: Car entry system,Face detection, Face recognition, Live body detection

1. **Introduction.** With the rapid development of society, the automobile has become one of the necessary transportation for most families to travel and become one of the necessities of life. The automobile is not only a means of transportation but also represents the latest achievements in the field of science and technology. Traditional automobile technology has been developed to be very mature. However, current car entry systems rely on physical

keys, such as remote control keys, NFC keys, or UWB keys, which not only bring a lot of inconvenience to car owners but also have certain security risks. The rapid development of face recognition technology in recent years provides a new opportunity for automotive entry systems. Face recognition technology has the security, convenience and people's demand for automobile entry systems, and the combination of artificial intelligence and automobiles is the main direction of current research [1, 2].

Early automobile entry systems, unlocked by a physical key, had problems such as low security, easy imitation, and cumbersome unlocking methods. In response to the problems of the mechanical key, a remote control key was introduced, which unlocks the vehicle through wireless radio frequency. Compared with the traditional mechanical key, it is convenient to operate and also improves the security of the vehicle. However, there are problems such as same-frequency interference, signal being copied and retransmitted, and still need to unlock manually in operation. With the popularization of intelligence in automobiles, many novel technologies such as digital keys based on Ultra Wide Band (UWB) technology, Near Field Communication (NFC) technology, and APP cloud services have appeared in automotive entry systems. UWB digital key utilizes a base station arranged in the vehicle with a handheld UWB key for ranging to unlock the vehicle [3]. UWB utilizes ultra-wide baseband pulses with an extremely wide frequency spectrum for communication. It can effectively avoid the relay station attack and has the characteristics of low power consumption, and high security, but the high cost of UWB, but also the lack of uniform standards of the problem. NFC digital key is used in the vehicle to set nfc reader and handheld nfc digital key to communicate with each other, data exchange, and verification identity information, to unlock the vehicle [4]. NFC communication is realized by electromagnetic induction and usually needs to be the key and the card reader between close contact to unlock the vehicle. NFC communication is realized by electromagnetic induction, which usually requires close contact between the key and the reader to establish a communication connection. both UWB digital key and NFC digital key are not bound to the user's identity, and the physical key needs to be worn on the body when using the key, which will lead to the loss of the key and cause the car can't be opened, and the vehicle is at the risk of being stolen if it is picked up by the lawless elements. Based on the APP cloud service car unlocking through the APP on the phone to complete the binding with the vehicle can be remote control of the vehicle through the APP [5]. but APP cloud service needs to be used in the case of a network, in some underground garages or other places where the signal is not good may not be able to use. The development of face recognition technology provides a new idea for the car entry system.

Li et al. [6] proposed a face recognition algorithm based on the combination of weighted wavelet variation and discrete cosine transform applied in car entry system. Pathak et al. [7] used OpenCV method for face recognition in car entry system so as to unlock the vehicle. Aruna et al. [8] used Siamese neural network method for face recognition to unlock the car.

In summary, automobile entry system using face recognition technology is highly secure and convenient. This paper proposes a face recognition based automobile entry system with face detection, face recognition, and live body detection functions, and offline, high precision, and lightweight. The design of this system is of great significance for improving vehicle safety and enhancing the user's automobile experience.

2. Related work.

2.1. Current research status of facial related algorithms.

Face detection is a prerequisite step for face recognition, where the key information of the face and the face coordinates are detected by the face. With the development of face detection research in recent years, face detection based on Convolutional Neural Networks (CNNs) has been rapidly developed and many advanced CNN architectures have been proposed to improve the accuracy and speed of face detection. Face detection can be viewed as a single target detection, and some target detection models can be improved to be used for face detection. Hu and Huang [9] used ResNet-50 to replace the original feature extraction network of SSD and combined it with CamShift algorithm to improve the accuracy and detection speed of face detection. Qi et al. [10] proposed a YOLOv5-based YOLO5Face face detection model with modifications to the backbone, using smaller sized kernels in SPP, and adding P6 outputs in the PAN block to achieve better performance in the WiderFace dataset. Deng et al. [11] proposed the RetinaFace lightweight model, RetinaFace, a face detection and face key-point localization for deep learning model, which can detect faces in images and recognize their keypoints in an efficient manner, featuring small model size and high accuracy.

Face live detection plays an important role in face recognition security [12]. Generally live detection is a classification task, i.e., determining whether an image is live or not. As face recognition technology is widely used in the society, face attack methods have evolved and increased. The common means of face attack are printed paper photo attack, electronic screen attack, video playback attack, silicone mask and so on. Therefore it is necessary to add face live detection in the face recognition process. Yang et al. [13] proposed a spatio-temporal anti-fraud network which can significantly improve the performance on public face fraud prevention datasets. Wang et al. [14] proposed an efficient decomposition representation learning method for cross-domain face PAD.Firstly, the feature decoupling module is utilized to extract the fraudulent information from different domains, and later on the Multi-domain learning module is used to learn the decoupled information from different domains, and then domain-independent intrinsic features can be obtained. Quan et al. [15] proposed a semi-supervised learning framework and an adaptive transfer mechanism to improve the robustness of the vivo detection model. Shao et al. [16] proposed a regularized fine-grained meta-learning framework for the face vivo detection task, which can be used to improve the generalization ability of the model. ability. This will help to improve the robustness and accuracy of the face recognition system.

Face recognition is the process of converting one-dimensional feature vectors and matching them with pre-stored faces in the database by feature extraction from the images after face detection. In recent years, face recognition based on deep learning has been rapidly developed, and in 2014, Deep Face [17] proposed by Face Book reached 97.35% accuracy on LFW dataset respectively; then, VGGFace [18] proposed by Oxford University improved this index to 98.95%; Google team proposed by FaceNet [19], which improved this metric to 99.6%.

2.2. **RetinaFace.** RetinaFace is a deep learning based face detection algorithm, which has the advantages of high efficiency, accuracy and scalability compared to traditional algorithms. In this paper, RetinaFace algorithm is used to realize face detection RetinaFace network structure is shown in Figure 1.

The main structure of RetinaFace contains four parts: Feature Extraction Network, FPN, SSH, and Head. The backbone feature extraction network uses MobileNet V1-0.25 network structure, MobileNet V1-0.25 is MobileNet V1 channel number compression to the original 1/4. RetinaFace's network structure is shown in Figure 1, the input image is feature extracted by the MobileNet V1-0.25 network, and C3, C4, C5 feature maps are taken as output. C3, C4, C5 feature maps as output.

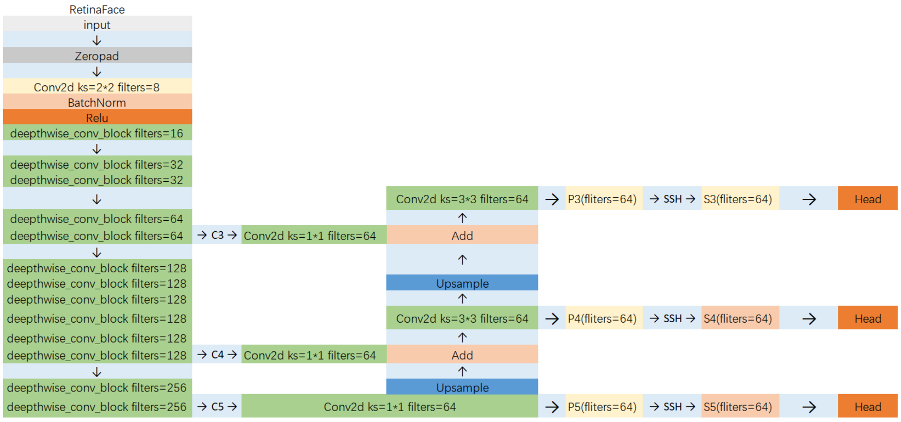

Figure 1. Network structure of RetinaFace

FPN (Feature Pyramid Network) will further enhance the feature extraction capability by adjusting the number of channels and up-sampling the feature maps generated by the backbone feature extraction network after 1×1 convolution, and the extracted feature maps of different scales are fused to obtain the P3, P4, and P5 feature layers.

SSH (Single Stage Headless) can further increase the sensory field, and three feature layers S3, S4 and S5 are obtained after SSH. The following prediction results are obtained by S3, S4, and S5: classification, prediction frame, and face keypoint prediction.

2.3. **MobileFacenet.** MobileFacenet is a lightweight convolutional neural network model. Its network structure consists of the following key components.MobileFacenet borrows the design idea of MobileNetV2 and adopts a depth-separable convolutional layer to reduce the number of parameters and computational complexity. Depth separable convolution is divided into two steps, depth convolution and point-by-point convolution, the former is used to deal with the information interaction between channels, and the latter is used to deal with the features in the spatial dimension. The overall architecture of MobileFacenet is designed to be simple and efficient, and through the combination of depth-separable convolution and bottleneck residual module, the number of parameters and computational complexity are greatly reduced while maintaining accuracy. This makes MobileFacenet ideal for mobile devices and embedded systems.

3. Vehicle Entry System Modeling Improvement.

3.1. **Face Detection Model Improvement.** Using RetinaFace model in face detection. Our goal is to improve the speed and accuracy of face detection in automotive face recognition tasks. To achieve this goal, we have adapted and improved the original RetinaFace structure. In the improved RetinaFace, we use FPN (Feature Pyramid Network) to handle faces at different scales. In the original RetinaFace structure, the FPN consists of three feature layers, P3, P4, and P5, which are used to detect faces of different sizes. However, in automobile face recognition tasks, we usually only need to detect larger faces, and there is usually no need to detect smaller faces. Therefore, we remove the feature maps at the bottom layer of the FPN in the improved RetinaFace. Specifically, we realize the ignoring of smaller faces by removing the branch related to P3 in the structure of RetinaFace. In this way, we can reduce unnecessary computation and processing, thus speeding up face detection. Meanwhile, we retain the P4 and P5 branches for detecting larger faces to ensure accuracy. The improved RetinaFace structure is shown in Figure 2.

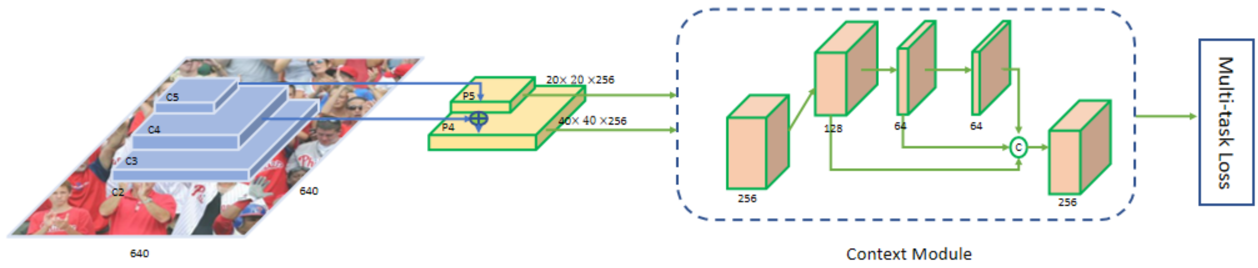

FIGURE 2. Structural diagram of the improved RetinaFace

3.2. **Face Recognition Model Improvement.** In the automobile face recognition task, it is difficult to deploy a large model considering the limited computational power of the embedded side. Therefore, in this paper, MobileFaceNet, a lightweight face recognition model, is chosen as the base model to further optimize the network performance and network expressiveness by adding the Squeeze-and-Excitation (SE) attention mechanism [20]. The SE module is introduced to capture the correlations between each channel and use these correlations to re The SE module consists of two parts, Squeeze and Excitation, firstly, each channel in the input feature map is "Squeezed" by global average pooling to obtain a channel weight vector. Then, this weight vector is "Excited" by two fully connected layers, i.e., the weight vector is mapped to the range of [0,1] by a sigmoid activation function. Finally, this weight vector is multiplied by the original feature map to obtain a new feature map. This new feature map contains the recalibrated channel information to enhance the expressiveness and performance of the model. The SE algorithm flow is shown in Figure 3. It shows the order and relationship between Squeeze and Excitation operations. By introducing the SE module, we can enhance the model's ability to capture important features while maintaining a lightweight network structure. This helps to improve the accuracy and performance of the automobile face recognition task while adapting to the computational resource constraints at the embedded end.

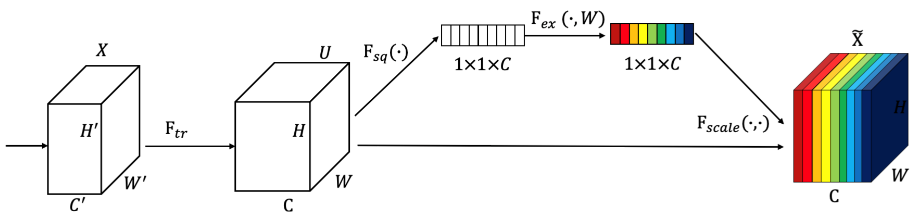

FIGURE 3. Flowchart of SE algorithm

Squeeze performs a global pooling operation through the global average pooling layer to compress the input feature map into a $1 \times 1 \times C$ feature map. The real number Z corresponding to each feature channel denotes the attention weight on the feature channel, which is used to enhance the critical feature channel and suppress the non-critical feature channel. The formula for this is:

$$z_c = F_{sq}(u_c) = \frac{1}{H \times W} \sum_{i=1}^{H} \sum_{j=1}^{W} u_c(i,j) \tag{1}$$

The purpose of the Excitation operation is to enhance the important features in the feature map and weaken the unimportant features, through the two fully connected layers,the vector z obtained in the previous step is processed to get the channel weight value

s that we want, after the two fully connected layers, the different values in s indicate the weight information of different channels, giving the channels different weights. The formula is:

$$s = F_{ex}(z, W) = \sigma(g(z, W)) = \sigma(W_2\delta(W_1 z)) \quad (2)$$

The MobileFaceNet feature extraction network has five Bottleneck layers, and the network structure of Bottleneck is shown in Figure 4. The structure of the Bottleneck layer after adding the SE module is shown in Figure 5.

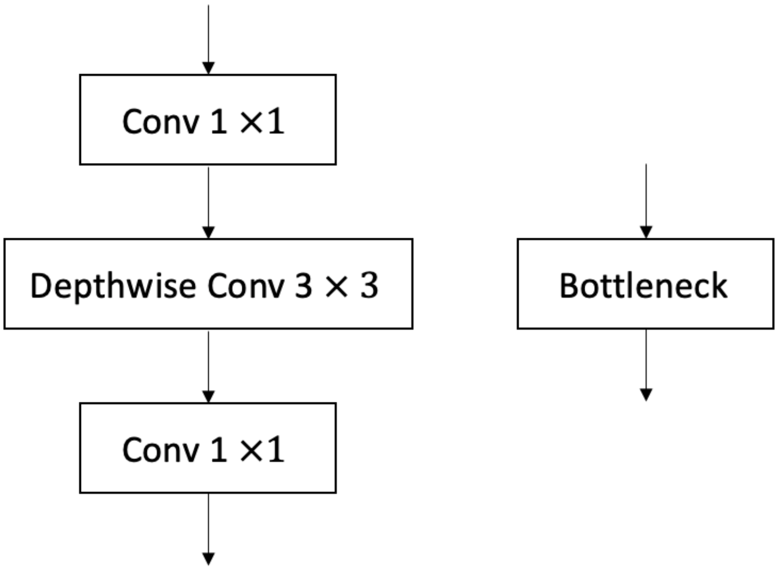

FIGURE 4. Distribution of energy consumption under different driving conditions

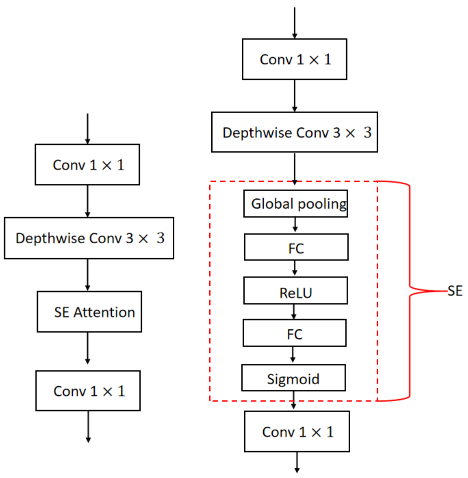

FIGURE 5. Energy consumption prediction model framework

3.3. **Improvement of face vivo detection model.** In this paper, we use a dual NIR+RGB camera for vivisection. The difference between the NIR camera and the common RGB camera is that the NIR camera receives electromagnetic wavelengths between 780nm-2526nm. Infrared camera works by infrared lamps emit infrared light irradiation object, infrared diffuse reflection, is received by the infrared camera, the formation of images. Therefore, in the near infrared electronic screen and smooth surface photos can not be normal imaging, which can be directly judged as a non-living body. In this paper, MobileFaceNet is used as the network basis to design a face living body detection model and infrared camera to realize face living body detection.

Many excellent algorithms have emerged in the field of vivo detection, however, successfully deploying these algorithms to the embedded side faces several challenges, and this paper aims to overcome these challenges to realize a vivo detection model that runs efficiently on embedded systems. To accommodate the limitations of the embedded side, we have chosen MobileFaceNet as the live body detection model. In the live body detection task, we set 2 categorization numbers, i.e., live and non-live bodies. Considering that this task requires more detailed face features, we adjusted the configuration of MobileFaceNet and set the size of the input image to $80 \times 80 \times 3$. This design choice helps to achieve efficient feature extraction with limited resources. Since the final output of the MobileFaceNet model is a 1×128 dimensional feature map and the number of classifications in the vivo detection task is 2, we added a 1×2 output fully connected layer at the end of the network structure. The improved network structure for live body detection is shown in Table 1:

Table 1. Improved network structure

Input	Operator	t	c	n	s
80×80×3	Conv3×3		64	1	2
40×40×64	Depthwise Conv3×3		64	1	1
40×40×64	Bottneck	2	64	5	2
20×20×64	Bottneck	4	128	1	2
10×10×128	Bottneck	2	128	6	1
10×10×128	Bottneck	4	128	1	2
5×5×128	Bottneck	2	128	2	1
5×5×128	Conv1×1		512	1	1
5×5×512	Linear GDConv7×7		512	1	1
1×1×512	Linear Conv1×1		128	1	1
1×1×128	Linear Conv1×1		2	1	1

Where t is the number of times the Bottleneck is internally upscaled, c is the number of channels, n is the number of repetitions, and s is the step size.

4. **Experimental results and analysis.**

4.1. **Introduction to the experimental dataset.** In this paper, the following datasets are used in training RetinaFace and MobileFaceNet and face live detection models:

WIDER FACE [21] dataset contains $32,203$ images and $393,703$ face frames. Each subset contains 3 levels of detection difficulty: Easy, Medium, and Hard.This dataset is widely used for training and evaluating face detection algorithms and is one of the largest publicly available face detection datasets.

CASIA-WebFace [22] dataset contains $494,414$ images of 10575 different individuals, LFW [23] dataset contains $13,233$ images of 5749 individuals; CFP-FP [24] dataset has

7,000 images of 500 individuals.AgeDB30 [25]dataset contains 12,240 images of 440 individuals.

CelebA Spoof [26] dataset contains 625537 images of 10177 people. It contains 43 people's face attribute information, attack power type information, lighting information, and environment information. It is the largest dataset in the current open source live detection dataset.

4.2. **Model training.** This experiment mainly runs the software environment and hardware environment, the algorithms in this paper are programmed and implemented in WIN10 operating system, the programming language used is Python 3.8, the deep learning framework is PyTorch 1.8, and CUDA 11.1. Hardware configurations mainly include the CPU is Inter Core i5-12400F, the GPU is NVDIA GeForce RTX 3070 8GB, and 48GB of RAM.

RetinFace selects mobilenet V1-0.25 network and uses WIDER FACE dataset for training,using stochastic gradient descent to update the gradient, the initial learning rate is 0.001, batch_size is 32, epoch=250. in WIDER FACE dataset its experimental results are shown in Table 2, and the results of the initial model and the test results of the improved model are shown in Figure 6, and the results of the improved model running on Raspberry Pi 4B are shown in Table 3.

TABLE 2. RetinaFace experiment results

Model Network Structure	Accurate(/%)			Model size(/MB)
	Easy	Medium	Hard	
MTCNN	85.1	82.0	60.7	1.92
RetinaFace(RestNet50)	95.3	94.1	84.3	104.7
RetinaFace(MobileNet0.25)	90.4	87.7	73.6	1.72
Improved RetinaFace	91.9	80.4	36.4	1.71
YOLO5Face	95.9	94.4	84.5	350

FIGURE 6. Test results for the RetinaFace model

According to the experimental results, the improved RetinaFace has more accuracy degradation under the Hard difficulty of the WIDER FACE dataset, but it does not affect the performance of face detection under Easy and Medium difficulty. And it outperforms other models in terms of test accuracy and model size.

TABLE 3. model testing

Before streamlining (processing 2000 images at 720p)	Streamlined (processing 2000 images at 720p)	enhancement effect
148.1s	126.1s	14.9%

Running the deployment of the improved RetinaFace model on a Raspberry Pi 4B processed 2000 images at 720P with a 14.9% speedup in face detection. As shown in Table 3.

In the live body detection task, the CelebA Spoof dataset is used for training and the gradient is updated using stochastic gradient descent. The initial learning rate is 0.1, batch_size is 256, epoch=50. the Precision change during training is shown in Figure 7.

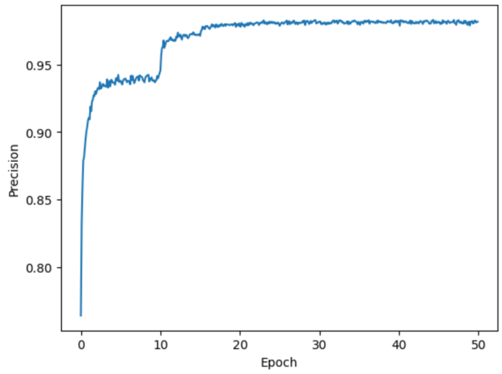

FIGURE 7. Changes in model training Precision

The face recognition model was trained using CASIA-WebFace for model training, and the test dataset was chosen from LFW, CFPFP, and AgeDB30 for model evaluation. Stochastic gradient descent was used to update the gradient with an initial learning rate of 0.01, batch_size of 128, and epoch=90. The model training results are shown in Figure 8, and the accuracies of the two models are shown in Table 4.

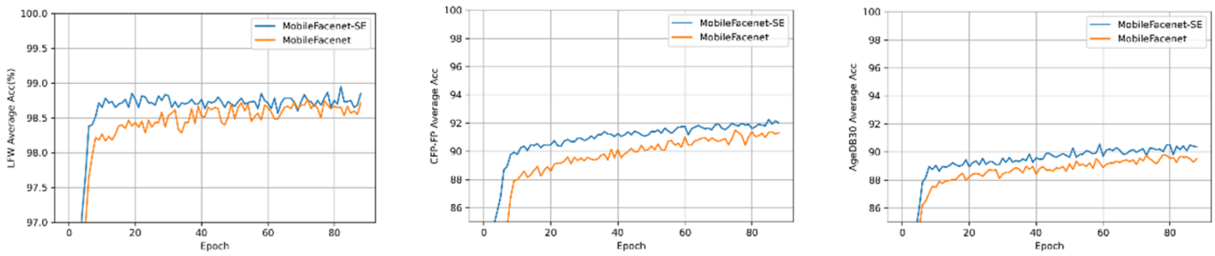

FIGURE 8. Comparison of the accuracy of the two models when trained

As evidenced by the data in Figure 8 and in Table 4, the model accuracy is improved by 0.18%, 0.77%, and 0.73% on the three test sets after adding the SE module to MobileFaceNet. As evidenced by the data in Figure 8 and Table 4, the model accuracy after adding the SE module improves by 0.18%, 0.77%, and 0.73% on the three test sets, respectively, and the addition of the SE module in MobileFaceNet can effectively improve the model's accuracy compared to the larger models such as Facenet and DeepFR which are smaller in size, and there is not a big difference in the recognition accuracy. The model is

Table 4. Comparison of the accuracy of the two models

Model Network Structure	Accurate(/%)			Model size(/MB)
	LFW	CFP_FP	AgeDB30	
MobileFacenet	98.77	91.49	89.82	4.889
MobileFacenet-SE	98.95	92.26	90.55	5.014
Facenet	99.63	-	-	30
DeepFR	98.95	-	-	500

smaller in size and has little difference in recognition accuracy compared to larger models such as Facenet and DeepFR.

4.3. **Model deployment.** The models are deployed using NCNN, a mobile platform deployment tool developed by Tencent, which is an efficient neural network computational library that allows fast neural network inference on embedded devices. The trained face detection, live body detection and face recognition model files are converted to ONNX files and then simplified, and converted to bin and param files so that they can be read and run by the C++ side, and finally the C++ code is written in the Raspberry Pi to call the trained model to get the output of the model.

4.4. **Experimental result.** When the face information is not entered into the system, the face detection model detects the face and marks the face and five key points. The result shows 'Recognition error' which means that matching with the face in the database fails, the user is not found and a message with ID 1AB and data A4B1 is sent through CAN1 and the matching fails.

Experimental Platform The Raspberry Pi 4B is chosen as the main controller for this design. Its advantage lies in its small size, low price, with fast running speed, which is very suitable for application in this system. (CAN open environment, CANoe), a bus development software developed by Vector for automotive buses, supports the automotive distributed control system development process and can realize bus simulation and emulation of the whole vehicle, which is mainly applied to system design and modeling simulation, ECU development and bus simulation.

After clicking start in CANoe, the face is placed in the camera and face recognition starts. After successful face recognition, the face will be framed out and the five key points of the face will be marked out and the personal information of the person in the database will be displayed, and the device will send a message with ID 1AB and data A4B2 via CAN1. When the ECU detects a message with ID 1AB on the CAN bus, it will receive it and make a judgment based on the data of the message. If the data of the message is A4B2, it means that it is a message after successful face recognition as shown in Figure 9.

When the face information is not entered into the system, the face detection model detects the face and marks the face and five key points. The result shows 'Recognition error' which means that the matching with the face in the database fails, the user is not found and sends the message with ID 1AB and data A4B1 through CAN1, the matching fails as shown in Figure 10.

5. **Conclusions.** Aiming at the problems of low accuracy, poor stability, and slow running speed of hardware platforms in traditional face recognition, we have adopted a series

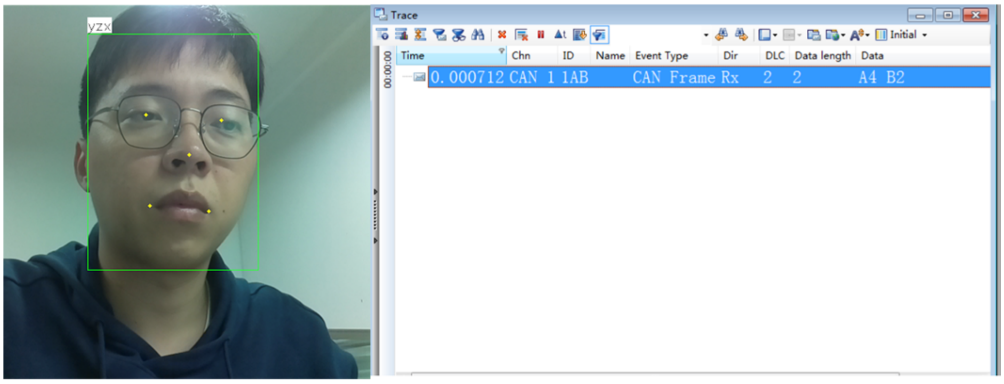

FIGURE 9. Entering face test results

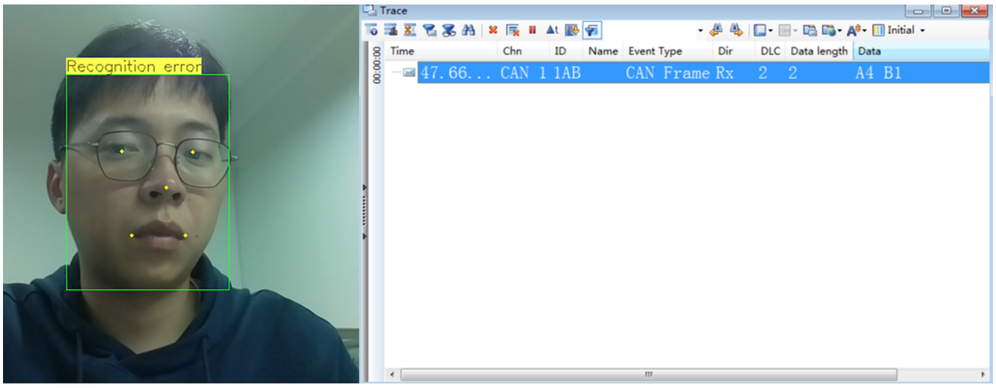

FIGURE 10. Unrecorded face test results

of improvement measures. First, in the face detection module, the RetinaFace deep learning model is optimized, which successfully improves the detection speed by 14.9% by removing the lowest-level feature maps in the FPN. Secondly, to improve the security of the system, a live body detection method is introduced, which effectively prevents spoofing attacks. In addition, the SE attention mechanism is introduced in the face recognition module to further improve the model performance. The improved MobileFacenet deep learning model improves by 0.18%, 0.77%, and 0.73% on the dataset. The car entry system can run on Raspberry Pi in real-time and is connected to the car via CAN bus. Our research results are important in practical applications. With the optimized face detection model, the system can detect faces quickly and accurately in real-time operation, which improves the response speed and user experience of the whole system. The introduction of the live body detection method enhances the security of the system and effectively prevents fake face attacks and spoofing behaviors. The improved face recognition model improves the accuracy of face recognition. However, there are some limitations in our research. In this paper, we mainly focused on the face recognition part of the vehicle entry system but did not involve the interaction with other systems, such as the vehicle control system and cell phone APP. Therefore, future research can further explore cross-platform compatibility, so that the face recognition-based vehicle entry system can be better integrated with other systems to achieve more comprehensive vehicle intelligence.

Acknowledgment. This research was funded by Fuzhou Science and Technology Plan Project: 2022-Q-009 and by Fujian Provincial Health Commission: 2020CXA053.

REFERENCES

[1] A. A. Elngar and M. Kayed, "Vehicle security systems using face recognition based on internet of things," *Open Computer Science*, vol. 10, pp. 17–29, 2020. [Online]. Available: 10.1515/comp-2020-0003

[2] E. E. Vigneswaran, V. G. Raj, and M. Selvaganesh, "Authentication system for automobiles using face recognition methodology," *2022 4th International Conference on Smart Systems and Inventive Technology (ICSSIT)*, 2022.

[3] N. Palmen, "Hella launches digital car key with uwb technology," *Automotive Industries*, 2021.

[4] C. Busold, A. Taha, C. Wachsmann, A. Dmitrienko, H. Seudié, M. Sobhani, and A.-R. Sadeghi, "Smart keys for cyber-cars: Secure smartphone-based nfc-enabled car immobilizer," in *Proceedings of the third ACM Conference on Data and Application Security and Privacy*, 2013, pp. 233–242.

[5] V. G. M. J. Joseph, P. Sehdev, M. R. Khosravi, and F. Al-Turjman, "An iot-enabled intelligent automobile system for smart cities," *Internet of Things*, vol. 18, p. 100213, 2022.

[6] F. Li, X. Su, and H. Zhao, "Key technologies of face sensor recognition entry system for new energy vehicles based on particle swarm neural network," *Journal of Sensors*, vol. 2023, pp. 1–6, 2023.

[7] M. Pathak, K. N. Mishra, S. P. Singh, and A. Mishra, "An automated smart centralised vehicle security system for controlling the vehicle thefts/hacking using iot and facial recognition," in *2023 International Conference on Computational Intelligence and Knowledge Economy (ICCIKE)*. IEEE, 2023, pp. 516–521.

[8] S. Aruna, M. Maheswari, and A. Saranya, "Car security system with face recognition using convolution neural network," *Materials Today: Proceedings*, vol. 68, pp. 152–155, 2022.

[9] X. Hu and B. Huang, "Face detection based on ssd and camshift," in *2020 IEEE 9th Joint International Information Technology and Artificial Intelligence Conference (ITAIC)*, vol. 9. IEEE, 2020, pp. 2324–2328.

[10] D. Qi, W. Tan, Q. Yao, and J. Liu, "Yolo5face: why reinventing a face detector," in *European Conference on Computer Vision*. Springer, 2022, pp. 228–244.

[11] J. Deng, J. Guo, Y. Zhou, J. Yu, I. Kotsia, and S. Zafeiriou, "Retinaface: Single-stage dense face localisation in the wild," *arXiv preprint arXiv:1905.00641*, 2019.

[12] Z. Yu, Y. Qin, X. Li, C. Zhao, Z. Lei, and G. Zhao, "Deep learning for face anti-spoofing: A survey," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 45, no. 5, pp. 5609–5631, 2022.

[13] X. Yang, W. Luo, L. Bao, Y. Gao, D. Gong, S. Zheng, Z. Li, and W. Liu, "Face anti-spoofing: Model matters, so does data," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 3507–3516.

[14] G. Wang, H. Han, S. Shan, and X. Chen, "Cross-domain face presentation attack detection via multi-domain disentangled representation learning," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 6678–6687.

[15] R. Quan, Y. Wu, X. Yu, and Y. Yang, "Progressive transfer learning for face anti-spoofing," *IEEE Transactions on Image Processing*, vol. 30, pp. 3946–3955, 2021.

[16] R. Shao, X. Lan, and P. C. Yuen, "Regularized fine-grained meta face anti-spoofing," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 34, no. 07, 2020, pp. 11 974–11 981.

[17] Y. Taigman, M. Yang, M. Ranzato, and L. Wolf, "Deepface: Closing the gap to human-level performance in face verification," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2014, pp. 1701–1708.

[18] O. Parkhi, A. Vedaldi, and A. Zisserman, "Deep face recognition," in *BMVC 2015-Proceedings of the British Machine Vision Conference 2015*. British Machine Vision Association, 2015.

[19] F. Schroff, D. Kalenichenko, and J. Philbin, "Facenet: A unified embedding for face recognition and clustering," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 815–823.

[20] J. Hu, L. Shen, and G. Sun, "Squeeze-and-excitation networks," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 7132–7141.

[21] S. Yang, P. Luo, C.-C. Loy, and X. Tang, "Wider face: A face detection benchmark," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 5525–5533.

[22] D. Yi, Z. Lei, S. Liao, and S. Z. Li, "Learning face representation from scratch," *arXiv preprint arXiv:1411.7923*, 2014.

[23] E. Learned-Miller, G. B. Huang, A. RoyChowdhury, H. Li, and G. Hua, "Labeled faces in the wild: A survey," *Advances in Face Detection and Facial Image Analysis*, pp. 189–248, 2016.

[24] S. Sengupta, J.-C. Chen, C. Castillo, V. M. Patel, R. Chellappa, and D. W. Jacobs, "Frontal to profile face verification in the wild," in *2016 IEEE Winter Conference on Applications of Computer Vision (WACV).* IEEE, 2016, pp. 1–9.

[25] S. Moschoglou, A. Papaioannou, C. Sagonas, J. Deng, I. Kotsia, and S. Zafeiriou, "Agedb: the first manually collected, in-the-wild age database," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 2017, pp. 51–59.

[26] Y. Zhang, Z. Yin, Y. Li, G. Yin, J. Yan, J. Shao, and Z. Liu, "Celeba-spoof: Large-scale face anti-spoofing dataset with rich annotations," in *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XII 16.* Springer, 2020, pp. 70–85.