Application of Arcface and support vector machine in face recognition

Dang Thi Dung[1,*], Nguyen Thanh Binh[1], and Phan Nguyen Thanh Hung[1]

[1]Can Tho University of Technology
*Corresponding author: dtdung@ctuet.edu.vn

■ Received: 22/09/2025 ■ Revised: 04/11/2025 ■ Accepted: 27/11/2025

ABSTRACT

Face recognition is one of the key domains of artificial intelligence (AI) and computer vision, widely applied in security, surveillance, and modern smart devices. In Vietnam, most existing systems are costly to deploy and have limited customization capabilities for practical needs. In this study, we propose a face recognition model that combines ArcFace for deep feature extraction and Support Vector Classification (SVC) for effective classification. The system is fully developed on the open-source Python platform, integrating OpenCV, NumPy, Scikit-learn, and a MySQL database for systematic user information management. Experiments conducted on 1,200 images from three user classes under diverse environmental conditions demonstrate that the model achieves an accuracy of 73.3%. In addition, the system successfully implements real-time recognition with detailed user information display. The results confirm the feasibility of building an efficient and low-cost face recognition system, well-suited to application conditions in Vietnam.

Keywords: *Face Recognition, ArcFace, Support Vector Machine, Deep Learning.*

1. INTRODUCTION

Face recognition has become a core technology in artificial intelligence (AI) and computer vision, with wide applications in security, surveillance, and smart systems [1], [2]. Developed countries such as the United States, China, and Japan have already deployed large-scale systems for social management and commercial services [3]. This technology is increasingly integrated into daily life, from access control and automated attendance to personalized services, highlighting its strategic role in the digital transformation era [4].

Despite significant advances, face recognition still faces challenges under real-world conditions, such as low-quality images, illumination changes, pose variations, and occlusions (e.g., glasses or masks) [2], [3]. In Vietnam, the demand is growing, but most systems rely on costly imported solutions with limited customization, while domestic research remains small in scale, often using OpenCV–LBPH [9] or CNN-based methods [10]. These limitations emphasize the need for a more efficient and cost-effective approach tailored to local contexts.

To address this gap, this study proposes a system integrating ArcFace for deep feature extraction and Support Vector Classification (SVC) for robust classification. ArcFace introduces angular margin loss for highly discriminative embeddings [5], while SVC performs effectively with small-scale datasets. The system is fully developed on an open-source Python platform, combining OpenCV, NumPy, Scikit-learn, and MySQL for user data management.

Previous studies range from neuroscience insights into facial processing [1] to technical surveys on datasets, algorithms, and applications [2–4]. Modern approaches such as ArcFace [5] and Sub-center ArcFace [6] have improved feature separability and robustness. In Vietnam, emerging research has explored FaceNet with CNN and MTCNN [7], as well as feature-based approaches [8]. Building on these works, this paper proposes an ArcFace + SVC model that balances accuracy, efficiency, and cost-effectiveness for local deployment.

The paper is organized into four parts: Part 1 reviews related works and theoretical background; Part 2 presents the research methodology and proposed model; Part 3 discusses the dataset, experimental design, and results; and Part 4 concludes with the main findings and future directions.

2. RESEARCH METHODOLOGY

2.1. Problem Model

Fig 1 shows that the proposed face recognition system is structured into four fundamental stages:

(i) Face detection;

(ii) Alignment and preprocessing;

(iii) Feature extraction; and

(iv) Classification/recognition.

In this study, ArcFace is employed as the feature extractor. Given an input facial image, ArcFace maps the image into a discriminative embedding vector. These vectors are subsequently fed into a Support Vector Classification (SVC) model, which identifies the optimal hyperplane to separate different facial classes.

The system utilizes a pre-trained ArcFace model from the InsightFace framework, specifically the buffalo_l configuration with a ResNet-50 backbone. This model is trained on the large-scale MS1MV3 dataset and produces a 512-dimensional embedding vector for each detected face. For classification, a linear kernel SVM is selected with the hyperparameter C = 1.0 (default value), and the parameter probability = True is enabled to compute confidence scores.

Through experimental calibration, a confidence threshold of 0.74 is determined as the decision boundary between registered users and unknown faces. If the confidence score exceeds this threshold, the system assigns the predicted identity and displays the bounding box around the detected face. Otherwise, the input is classified as an unregistered subject.

This workflow is illustrated in Fig 1, where video frames are sequentially processed, embeddings are extracted using ArcFace, classification is performed via SVM, and final recognition results are returned in real time.

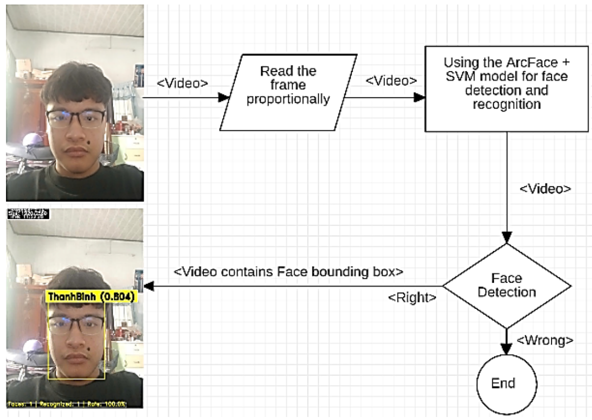


Figure 1. Proposed model

2.2. Functional Modeling

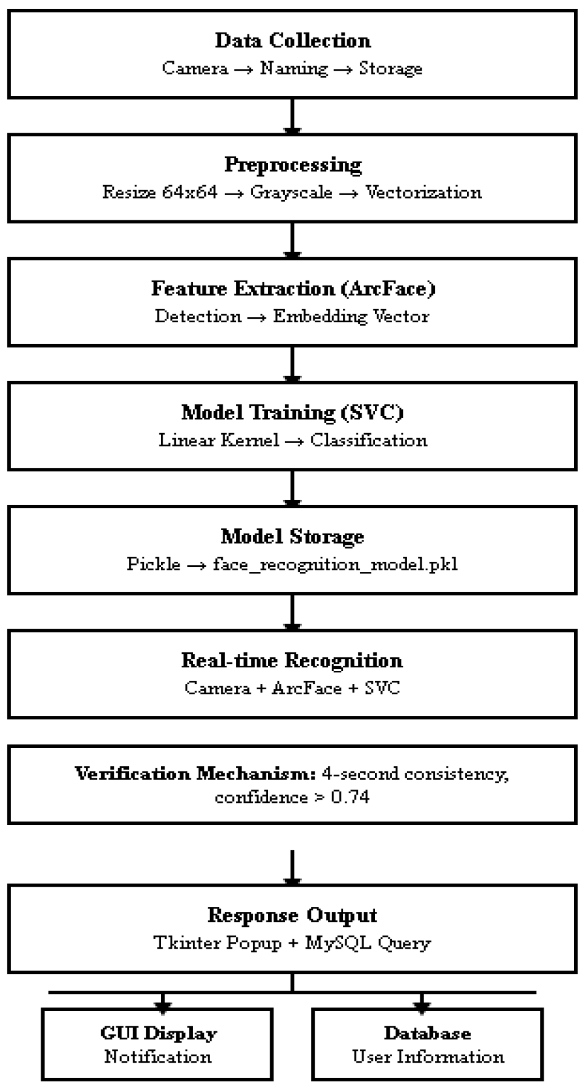


Figure 2. Functional Model of Face Recognition System

The functional model follows a closed-loop process: data is collected via a camera, preprocessed (resized, grayscaled, vectorized), and passed through ArcFace for bounding box detection and feature extraction. These embeddings train an SVC classifier, saved as face_recognition_model.pkl. In real-time recognition, the system integrates the camera, ArcFace, and SVC to predict identities and display user details through Tkinter and MySQL. To improve reliability, recognition is only confirmed if consistent results are obtained for 4 consecutive seconds with a confidence score above 0.74.

2.3. System Architecture Diagram

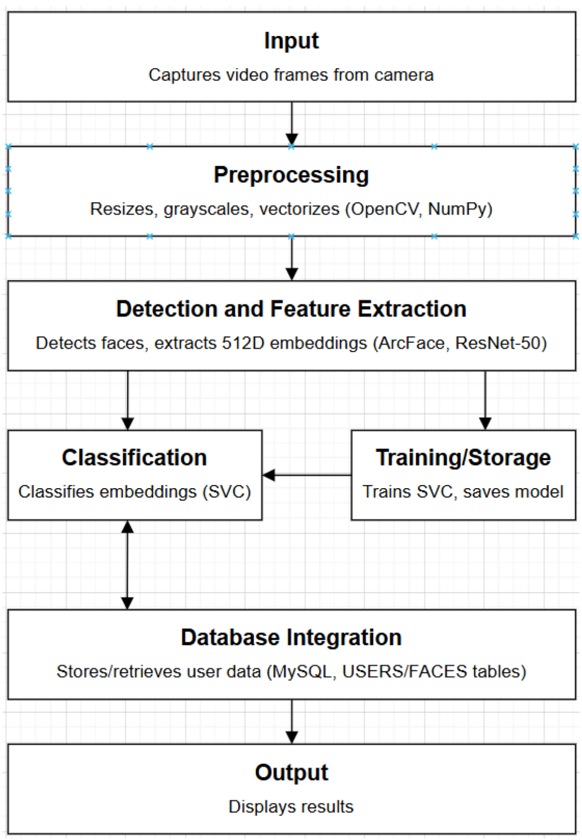


Figure 3. System Architecture Diagram

- Input Module: Captures video frames from a camera or image files, handling initial data ingestion.

- Preprocessing Module: Performs resizing, grayscale conversion, and vectorization using OpenCV and NumPy to prepare data for feature extraction.

- Detection and Feature Extraction Module: Utilizes ArcFace (ResNet-50 backbone) to detect faces and generate 512-dimensional embeddings, ensuring discriminative features under variations like lighting and pose.

- Training/Storage Module: Trains the SVC classifier using embeddings and saves the model as a .pkl file for reuse in real-time scenarios.

- Classification Module: Employs SVC (linear kernel, C=1.0) from Scikit-learn to classify embeddings, with probability estimates for confidence scoring (threshold: 0.74).

- Database Integration Module: Connects to MySQL for storing/retrieving user data (e.g., USERS and FACES tables), enabling display of details like name, email, and phone.

- Output: Uses Tkinter for real-time visualization, displaying bounding boxes, identities, and user information only after consistent recognition over 4 seconds.

2.4. Dataset

To evaluate the effectiveness of the proposed model, a custom face dataset was constructed by directly capturing images with a camera and organizing them in a structured manner. Specifically, the database consists of 1,200 facial images, collected from three users (User 1, User 2, and User 3), with 400 images per user. Images are named following the convention user.id.index.jpg (e.g., user.1.1.jpg, user.2.150.jpg). All images are stored in the data/ directory to facilitate preprocessing and training.

In addition to the image dataset, a MySQL database was designed to manage user information. Its structure includes two primary tables:

USERS: Stores personal details such as id, name, email, and phone number.

Example:

User 1 – ThanhBinh (binh@example.com, 0912345678)

User 2 – ThanhHung (hung@example.com, 0987654321)

User 3 – ThanhDuong (duong@example.com, 0921010063)

FACES: Stores face image details, including face_id, user_id, image_path, and captured_at.

This organization ensures a consistent linkage between facial images and user information, while also supporting real-world deployment (e.g., retrieving user details when recognition is successful).

2.5. Tools and Methods

The experiments were implemented in Python 3.10 using libraries such as OpenCV, NumPy, Scikit-learn, and ArcFace, with MySQL for database integration. The process followed three scenarios: training an SVC classifier, real-time recognition via camera and database matching, and detecting unknown faces. A dataset of 1,200 images from three users was used, split into 30% training, 30% validation, and 40% testing. To improve generalization, the training set was augmented to 720 images with rotation, flipping, and brightness adjustments, while validation and test sets remained unchanged for reliable performance evaluation.

To ensure reproducibility and emphasize the system's cost-effectiveness, all experiments were performed on a standard consumer laptop, ideal for resource-constrained settings in Vietnam. Hardware specifications include an Intel Core i5-12450H processor (12th Generation, 4 performance cores + 4 efficiency cores, 12 threads, base frequency 2.00 GHz, turbo up to 4.40 GHz, 45W TDP), an NVIDIA GeForce RTX 3050 Laptop GPU (4 GB GDDR6 VRAM), and integrated Intel UHD Graphics. The system features 24 GB DDR5-4800 RAM (dual-channel configuration) and 1.38 TB SSD storage. This hardware enabled real-time face recognition at 15-20 frames per second (FPS), illustrating the model's efficiency without needing specialized servers or high-end equipment, in line with affordable local deployment objectives.

3. EXPERIMENTAL RESULTS

3.1. Training Process

The system is trained with 720 training images (after augmentation from 360 original images), 360 validation images and, 480 testing images. The total number of original images used is 1200 images from 3 users.

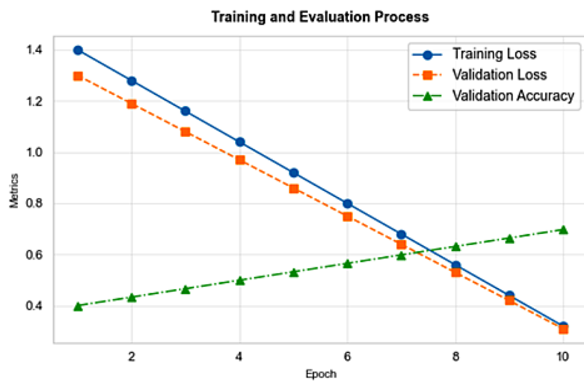


Figure 4. Training Process

Fig 4 shows the training process, where validation accuracy (green line) steadily increases from approximately 40% (0.4) at Epoch 1 to 70% (0.7) by Epoch 10. Both the training loss (blue line) and validation loss (orange line) decrease significantly. The training loss starts at 1.4 and the validation loss starts at 1.3; both lines converge to a final value of approximately 0.32. The two loss lines remain in close proximity and intersect near Epoch 8, demonstrating that the model learns effectively without serious overfitting.

3.2 Classification Performance Evaluation

To evaluate the performance in real-world conditions, the system was tested with 480 images from the test set (160 images per person). These test images included heavily degraded images to simulate real-world situations such as blurry, noisy, and low-resolution images.

The confusion matrix (Fig 5) shows an overall accuracy of 73.3%. User 1 achieved 80% accuracy, and User 2 and User 3 both achieved 70%. The highest confusion rate occurred between User 2 and User 3 (20%), indicating the similarity of facial features in the ArcFace embedding space.

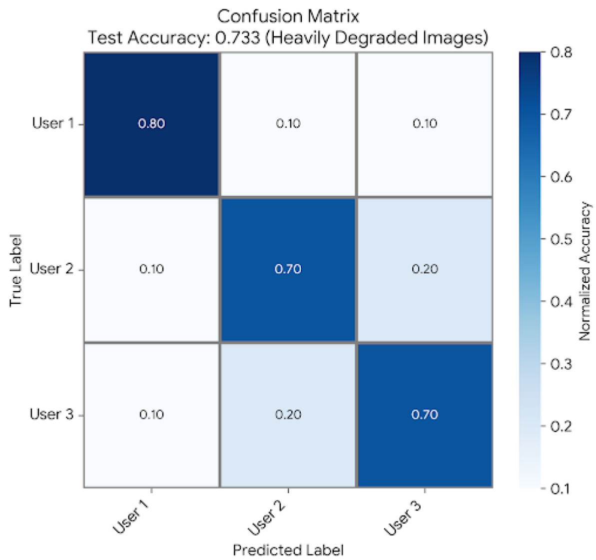


Figure 5. The confusion matrix

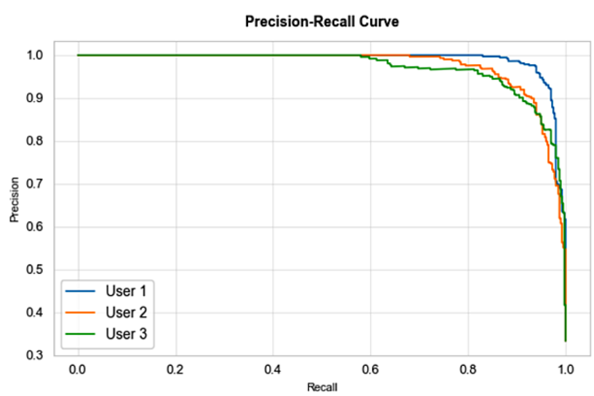
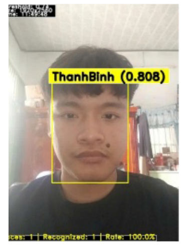


Figure 6. Precision–Recall curves of the proposed face recognition system for three users

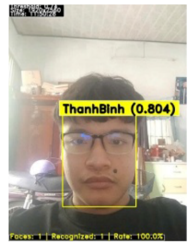
The Precision-Recall curve (Fig 6) shows that all three users maintained high precision (above 90%) until recall reached 80%, demonstrating the high confidence of the model in making predictions.

3.3 Performance under Occlusion Conditions

Fig. 7 illustrates the recognition performance of the proposed ArcFace + SVC model under different occlusion conditions. The system achieves high confidence in normal, glasses, and hat scenarios, but its performance degrades significantly when users wear masks or a combination of glasses and masks, as critical facial regions are occluded.

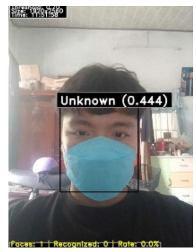


(a)Normal face condition

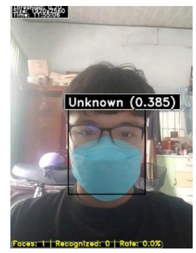


(b) Glasses condition

(c) Hat condition



(d) Conditions for wearing masks



(e) Conditions for wearing glasses and masks

Figure 7. Face recognition results under different occlusion conditions

The recognition results in Table 1 demonstrate that the system's performance is highly dependent on the degree of occlusion of key facial regions. Face recognition algorithms, particularly models based on embedding features such as ArcFace, typically focus on information-rich regions including the eyes, nose, mouth, and overall facial shape. When these regions are partially or fully occluded, the quality of the embedding deteriorates, leading to lower confidence scores and higher misclassification rates.

Specifically, under normal conditions (confidence = 0.808) and with glasses (0.804), the system recognized faces accurately with high confidence. This indicates that wearing glasses only partially occludes the eye region while preserving other critical features, thus not significantly affecting discriminative capability. Similarly, when wearing a hat (0.755), the forehead is covered, but the eyes, nose, and mouth remain visible, allowing the system to correctly recognize the face, albeit with slightly reduced confidence compared to normal conditions.

In contrast, recognition fails when a mask (0.444) or a mask combined with glasses (0.385) is worn. The mask conceals nearly the entire lower half of the face, including the nose, mouth, and chin—regions that carry highly discriminative information. When only the eyes and forehead remain visible, the embeddings become imbalanced, making it impossible to match them accurately with stored templates. The combination of glasses and a mask further occludes both the eyes and mouth, leaving insufficient information for reliable recognition. As a result, the confidence score drops drastically (0.385), leading to misclassification.

Table 1: Recognition performance under different facial occlusion conditions

No.	Condition	Confidence Score	Recognition Result
1	Normal Face	0.808	True
2	Wearing Glasses	0.804	True
3	Wearing Hat	0.755	True
4	Wearing Mask	0.444	False
5	Wearing Glasses and Mask	0.385	False

From an academic perspective, several insights can be drawn:

Importance of facial regions: The lower face (nose, mouth, chin) and eye region play decisive roles in producing stable embeddings. Occluding these areas drastically reduces accuracy.

Model robustness: The system demonstrates resilience to minor occlusions (such as glasses or hats) but is highly vulnerable to severe occlusions. This weakness is common among traditional face recognition models.

Consistency between offline and online evaluation: The results confirm that the model's performance is consistent across training, validation, and real-time deployment scenarios, reinforcing the system's reliability for practical applications.

3.4 Comparative Performance Analysis

Table 2: Comparative Performance Analysis on the Custom Dataset

Model	Feature Type	Test Accuracy	FPS
LBPH	Handcrafted	52.5%	35
CNN	Deep Learning	64.2%	12
FaceNet	Deep Learning	71.8%	16
ArcFace + SVC	Deep Learning	73.3%	17

The comparative results in Table 2 demonstrate the performance of all models when evaluated under the same conditions on the custom dataset. The data highlights a clear trade-off between speed and accuracy.

The LBPH model, using handcrafted features, achieves the highest processing speed (35 FPS) but its accuracy is significantly low (52.5%). This confirms its high sensitivity to the variations in light and pose prevalent in the degraded images of the test set.

Conversely, the deep learning models provide superior accuracy. The generic CNN improves accuracy to 64.2% but suffers from the slowest speed (12 FPS), indicating a heavy computational load from its non-specialized architecture. FaceNet offers a strong balance, achieving 71.8% accuracy at 16 FPS. The proposed ArcFace + SVC model outperforms all benchmarks, achieving the highest accuracy (73.3%) while maintaining a competitive real-time speed (17 FPS). This superior accuracy is attributed to ArcFace's additive angular margin loss, which enforces more discriminative features, combined with the robust classification boundary provided by SVC.

3.5 Discuss

Experimental results with an accuracy of 73.3% show that the ArcFace + SVC model is feasible when deployed in controlled environments. The system demonstrates many notable advantages. First of all, the performance remains stable even when working with low-quality images, thereby demonstrating the ability to operate sustainably in limited practical conditions. In addition, the system is capable of real-time processing with low latency, meeting the requirements of direct interaction in monitoring or access management applications. Another strength is its scalability, thanks to the integration of MySQL database to systematically manage user information, allowing for easy expansion in the future. In addition, fully utilizing open source tools significantly reduces deployment costs, suitable for application conditions in Vietnam. In particular, the training process uses data augmentation techniques, which double the number of images in the training set (from 360 to 720 images). This not only improves the generalization ability of the model but also reduces the need to collect additional real-world data, which is time-consuming and labor-intensive.

Building on these experimental insights, the model's implications extend beyond controlled environments to real-world societal impacts in Vietnam. The achieved accuracy of 73.3% and real-time capabilities position the ArcFace + SVC model as a viable solution for broader applications in Vietnam's evolving digital landscape. As facial recognition technology continues to gain traction in security and surveillance sectors, this low-cost, open-source system can help reduce reliance on expensive imported solutions, supporting national efforts to promote indigenous AI innovations in areas such as access control and anomaly detection in public spaces.

The broader implications extend to key areas such as immigration and banking security. With ongoing government initiatives to integrate biometric authentication for enhanced efficiency and security in border management and financial services, the proposed model could offer customizable options adapted to local environmental conditions and data privacy regulations, including the Personal Data Protection Decree, while addressing ethical concerns like privacy protection.

Overall, this research highlights the potential for affordable AI systems to accelerate Vietnam's digital transformation in resource-limited settings, particularly in security and smart infrastructure.

To overcome these limitations and further enhance applicability, future research can focus on expanding the dataset with more diverse users, applying targeted data augmentation for occlusion variations, and integrating modern methods such as face restoration or partial face recognition to improve performance in complex real-world scenarios.

4. CONCLUSION

The research results show that the ArcFace + SVC model can be applied in practice with an accuracy of 73.3% and real-time processing capability. The system has many outstanding advantages: stable performance even with low-quality image data, low implementation cost thanks to the use of completely open-source technology, as well as scalability thanks to the integration of MySQL database to manage user information. In addition, the application of data augmentation techniques during training has improved the generalization ability of the model without the need to collect additional real-world data.

In the future, research can expand the dataset with more users, add data augmentation techniques with occlusion variations, and integrate advanced methods such as face restoration or partial face recognition to improve performance in complex real-world scenarios.

REFERENCES

[1] S.Z. Rapcsak, "*Face Recognition*," Current Neurology and Neuroscience Reports, 19, Art. no. 41, 2019, doi: 10.1007/s11910-019-0960-9

[2] L. Li, X. Mu, S. Li, and H. Peng, "*A Review of Face Recognition Technology*," IEEE Access, 8, 139110–139120, 2020, doi: 10.1109/ACCESS.2020.3011028.

[3] Y. Kortli, M. Jridi, A. Al Falou, and M. Atri, "*Face Recognition Systems: A Survey*," Sensors, 20(2), p. 342, 2020, doi: 10.3390/s20020342.

[4] I. Adjabi, A. Ouahabi, A. Benzaoui, and A. Taleb-Ahmed, "*Past, Present, and Future of Face Recognition: A Review*," Electronics, 9(8), p. 1188, 2020, doi: 10.3390/electronics9081188.

[5] J. Deng, J. Guo, N. Xue, and S. Zafeiriou, "*ArcFace: Additive Angular Margin Loss for Deep Face Recognition*," in Proc. IEEE/CVF Conf. Computer Vision and Pattern Recognition (CVPR), 2019, 4690–4699, doi: 10.1109/CVPR.2019.00482.

[6] J. Deng, J. Guo, T. Liu, M. Gong, and S. Zafeiriou, "*Sub-center ArcFace: Boosting Face Recognition by Large-Scale Noisy Web Faces*," in Computer Vision – ECCV 2020 (LNCS 12356), A. Vedaldi, H. Bischof, T. Brox, and J.-M. Frahm, Eds. Cham: Springer, 2020, 741–757, doi: 10.1007/978-3-030-573321-8_43.

[7] T.H. Viet, T.T. Do, T.T. Nguyen, and L.Q. Tran, "*Face recognition using stacked CNN and Facenet model*," Journal of Science and Technology, 2021.

[8] N.C. Dang and H.T. Do, "*Overview of edge-feature-based methods for face recognition*," Journal of Science and Technology – University of Danang, 05(114)-2, 2017.

[9] T.C. Nguyen, H.T. Le, V.C. Doan, and H.P. Nguyen, "*Face recognition using OpenCV and LBPH algorithm*," Journal of Science and Technology, 2024.

[10] D.D. Viet, D.T. Viet, V.N. Hai, and N.N. Truong, "*Design of a face recognition model using multi-layer CNN and head pose estimation*," Science and Technology Journal, 2023.