A method combining multiple perspectives to enhance accuracy in facial recognition problems

Nguyen Duc Hanh[1], Nguyen Trong The[2*]

[1]Military Technical Academy, 236 Hoang Quoc Viet, Bac Tu Liem, Hanoi, Vietnam;
[2]Institute of Information Technology, 2 Hong Ha, Tan Binh, Ho Chi Minh City, Vietnam.
[*]Corresponding author: nguyentrongthevcntt@gmail.com
Received 2 Feb. 2024; Revised 20 Mar. 2024; Accepted 8 Apr. 2024; Published 20 May 2024.
DOI: https://doi.org/10.54939/1859-1043.j.mst.95.2024.76-84

ABSTRACT

This article introduces an advanced method in the field of facial recognition, using a unique technique that combines Convolutional Neural Networks (CNN) and Multilayer Perceptron (MLP) to integrate different perspectives. The highlight of this method is the application of CNN to analyze image features from multiple angles, along with MLP, to optimize the information synthesis process, thereby enhancing the accuracy of facial recognition under varying lighting conditions and angles. The main goal is to address the challenge of performance degradation in facial recognition in real-world situations, especially when there is a significant change in the viewpoint. This study details the model-building process from data collection and processing, training complex neural networks, and evaluating effectiveness through standard and experimental datasets.

Keywords: Facial recognition; Deep learning; Convolutional neural networks; Integrating multiple perspectives; Image processing; Viewpoint optimization; Multi-perspective analysis; Recognition performance improvement; Security applications.

1. INTRODUCTION

Among various human recognition problems using biometric technology, facial recognition is currently the most focused issue. Facial recognition serves powerfully in many areas of life, especially in high-tech fields requiring security and confidentiality. Therefore, the problem of facial recognition remains a hot topic, and people are constantly seeking ways to perfect it for the best recognition results. Facial recognition on computers has been researched and implemented by many authors, and over time, the effectiveness of the proposed methods has been increasingly improved.

In 1998, Wenyi Zhao and colleagues [1] used the PCA (Principal Component Analysis) method combined with LDA (Linear Discriminant Analysis). Initially, the facial image was projected from the raw image space to the facial space using PCA. Then, the LDA method was used to create a linear classifier capable of classifying facial classes. In 2001, Guodong Guo and colleagues [2] proposed using the SVM (Support Vector Machine) method for facial recognition. They used a strategy of combining multiple binary classifiers to build a multi-class SVM classifier. In 2013, in the article "Recognizing Part of the Face Without Alignment" [3] by S. Liao, A. K. Jain, and S. Z. Li, an algorithm was proposed for recognizing parts of the face without needing the coordinates of the two eyes for alignment. This algorithm is effective with facial images obstructed by objects, non-frontal images, images with limited angles of view, and overexposed images. In 2014, Y. Taigman, M. Yang, Ranzato, M. A., and L. Wolf [4] introduced "Facial Recognition Using the DeepFace Algorithm." In the paper, the research team from the Facebook Research Center and Tel Aviv University, Israel, proposed an algorithm named DeepFace, using images uploaded by users

to Facebook as the dataset. In 2015, F. Schroff and colleagues from the Google research team proposed an algorithm named FaceNet, learning to map facial images into a compact Euclidean space with measurable distances corresponding to facial similarity. Apart from Google, major technology companies also have their facial recognition models, such as Facebook's DeepFace [5], Microsoft's Azure Face [6], and Amazon's Rekognition [7]. These models allow facial recognition based on image and video data with high efficiency.

In 2023, by proposing the Progressive Learning Loss (PLFace) method [8], Baojin Huang and colleagues developed a progressive training strategy to learn how to balance performance between masked and unmasked facial recognition. PLFace stands out with its ability to adjust the relative importance of facial samples at different training stages, significantly improving the accuracy of masked facial recognition without decreasing the performance of normal facial recognition. Meanwhile, Xiaopeng Li and colleagues introduced MobileViTFace [9], a lightweight and efficient facial recognition model combining the structure of convolutional neural networks (CNN) and transformers (ViT). Compared to the standard ViT model, MobileViTFace requires less training data and has lower computational complexity, making it easy to deploy on edge devices. This model achieves superior recognition accuracy over lightweight models based on CNN and reduces parameter count and FLOPs by 5 times compared to ResNet-50 while maintaining similar accuracy.

However, these models are often applied to recognition with just one viewpoint. Sometimes, incomplete facial information affects recognition results when encountering issues with viewing angles, environmental influences, and the expressions of the recognized individuals. There is a need for a method that allows models to obtain and use more facial information, thereby enhancing recognition effectiveness. On this basis, our research team proposes a method combining multiple perspectives to increase accuracy in facial recognition problems.

The content of the article includes four parts: part 2 presents the theoretical basis and describes the method we propose; part 3 presents the simulation, calculation, and experimental results of this method; and part 4 discusses the conclusions and future development directions of the research.

2. TWO-VIEW FACIAL RECOGNITION MODEL

The method we propose involves using pairs of facial images of a person, extracted from two cameras arranged to capture the subject's images from the right and left sides. With images obtained from the cameras, the authors use the HOG (Histogram of Oriented Gradient) algorithm [10] to extract the subject's face.

The method of combining multiple perspectives in facial recognition is illustrated in the following diagram:

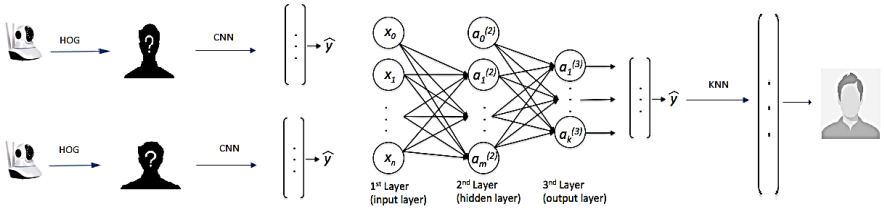

Figure 1. *System diagram of the method combining two perspectives in facial recognition.*

Face detection and vectorization

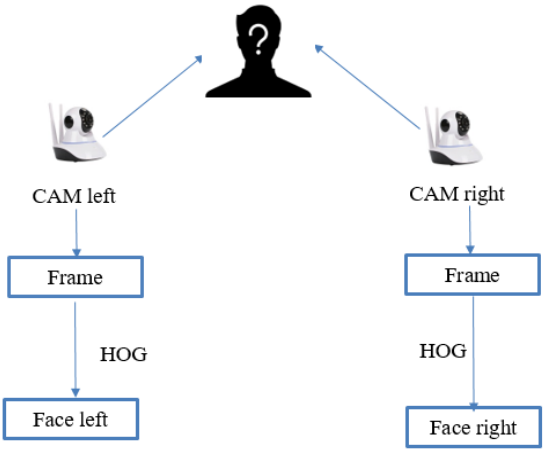

Figure 2. *Detect facial images from Camera Frame using HOG.*

The result of the extraction system will obtain pairs of left-right images of the face, forming a dataset consisting of a set of left-right image pairs. The system also stores a dataset of frontal facial images of people registered in the system, vectorized into 128-dimensional vectors and labeled.

The process of face vectorization uses a CNN network - a collection of stacked Convolution layers using nonlinear activation functions like ReLU and tanh to activate the weights in the nodes. Each layer, after passing through the activation functions, creates more abstract information for the subsequent layers.

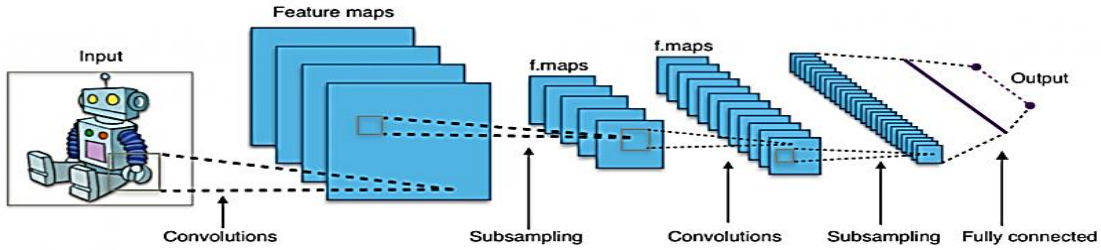

Figure 3. *CNN structure.*

The idea of this two-view combination method is to use the obtained image pair to vectorize into a pair of 128-dimensional vectors. This pair of vectors will be used for training and recognizing whether a person is in the stored database or not.

A CNN network is used to convert facial images into 128-dimensional vectors to vectorize the image pairs.

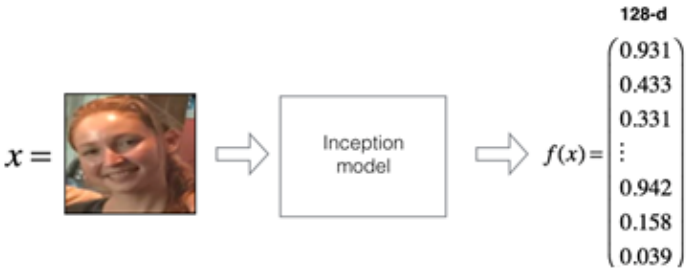

Figure 4. *Using the CNN network to vectorize facial images.*

Using a Multi-Layer Perceptron (MLP) to combine the pair of 128-dimensional vectors: For the specific problem the authors are researching, the MLP network used is a three-layer network including the input layer is the pair of 128-dimensional vectors obtained from the left-right image pair, the hidden layer has n neurons determined through experimentation, and the output layer is a 128-dimensional vector. This composite vector will be used to compare with the set of 128-dimensional vectors of labeled individuals in the database to decide whether this person is already in the system or not, and if so, who they are.

The training process is carried out with a dataset consisting of pairs of vectors: the 128-dimensional vector pair of the left-right facial images as input data, and the 128-dimensional vector of the frontal facial image as the desired output of the network.

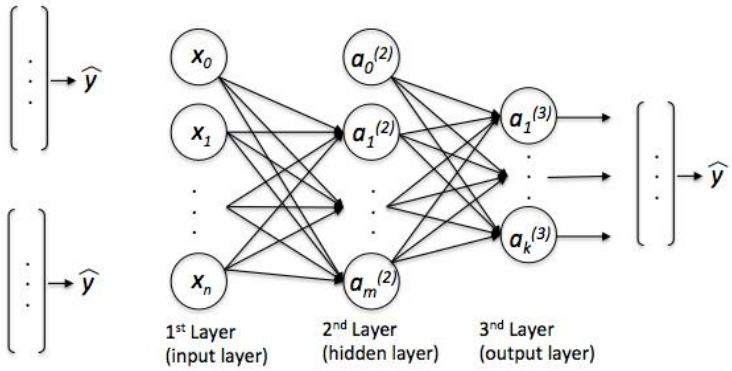

Figure 5. *Synthesizing the pair of 128-dimensional vectors into a composite vector.*

K-nearest neighbors (KNN)

After synthesizing the new 128-dimensional vector, the system uses the KNN algorithm, calculating the distance to the vectors in the database to identify the face. For the facial recognition problem being studied, the authors choose the number of neighbors as 1. The output of the MLP network will be used to compare with the labeled dataset to determine whether the face to be recognized is in the dataset, and if so, to identify the person.

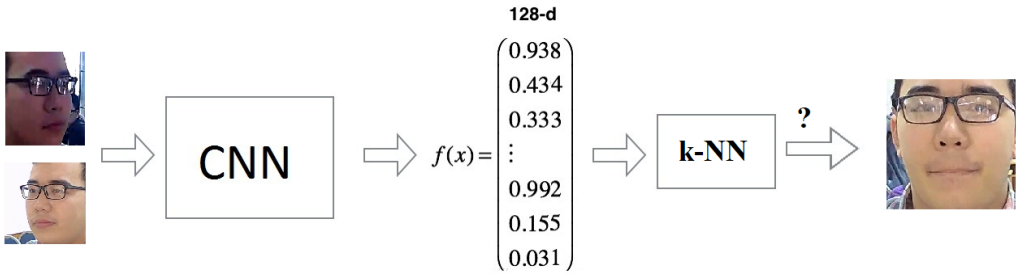

Figure 6. *Diagram describing the recognition result using k-NN.*

3. SIMULATION, CALCULATION, DISCUSSION

3.1. Input data

To conduct the simulation, the author group created a training and testing dataset by using cameras to collect data in classrooms at the University of Technology. In each classroom where data was collected, two cameras were used.

The cameras, IP types connected to computers via WiFi, recorded at a speed of 25 frames per second, and both cameras captured images simultaneously on the computer.

The two cameras were used to capture facial images from two different angles at the same time, and the subject passed between the two cameras. The cameras were placed at the classroom entrance, spaced 1.2 m – 1.4 m apart. Students walked between the two cameras, stopping in front of each camera from 0.7 m - 1 m away, ensuring the stopping position formed an angle of no more than 45 degrees with both the left and right cameras, pausing for about one to two seconds. To ensure successful data collection, each student passed the cameras twice. The authors used the cameras to collect frontal facial data of the subjects.

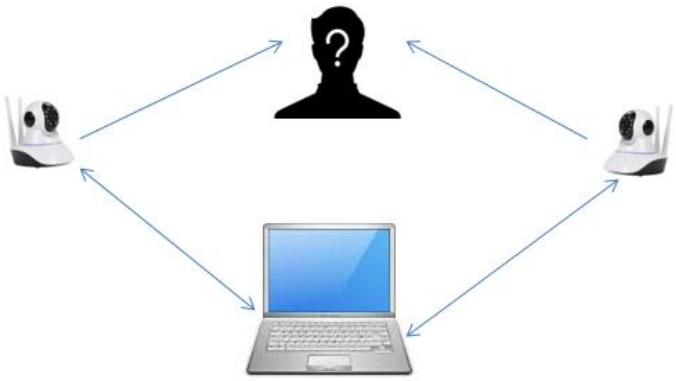

Figure 7. *Facial data collection model.*

The data collection resulted in nearly 600 images of 30 people, each person in the list having 5 to 10 pairs of facial images from the left and right cameras.

In addition to the self-collected dataset, the authors used a second dataset, the WebFace dataset from CASIA, a large-scale dataset containing about 10,000 subjects and 500,000 images, collected from the internet, with each subject's facial images captured from various angles.

3.2. Tools, simulation method

During the research, the author group conducted simulations based on Python 2.7 and used the Anaconda 3.0 library set.

Input data was taken from two computer cameras. To extract faces from videos, the author group used the HOG algorithm to detect the subject's face in the frames of the captured video. Images from the same time from the left-right cameras were paired to ensure the detection and encoding of faces, with cameras arranged to capture complete facial features: eyes, nose, mouth, etc., even at an angle.

```
So lop: 3
So Neural cua lop an cua mlp:124
MLPRegressor(activation='relu', alpha=1e-05, batch_size='auto', beta_1=0.9,
       beta_2=0.999, early_stopping=False, epsilon=1e-08,
       hidden_layer_sizes=124, learning_rate='constant',
       learning_rate_init=0.001, max_iter=200, momentum=0.9,
       nesterovs_momentum=True, power_t=0.5, random_state=1, shuffle=True,
       solver='lbfgs', tol=0.0001, validation_fraction=0.1, verbose=False,
       warm_start=False)
```

Figure 8. *Trained MLP network.*

Using CNN to vectorize the left-right image pairs into two 128-dimensional vectors, and vectorizing the frontal facial image. A neural network is used to combine the pair of two 128-dimensional vectors into one 128-dimensional vector.

Before combining the pair of vectors, an MLP network was trained with 180 pairs of facial images: two angled images as input and one frontal image as the desired output. After running measurements with the train and test datasets, the threshold coefficient η for k-NN was obtained.

Figure 9. *Example of paired faces after detection.*

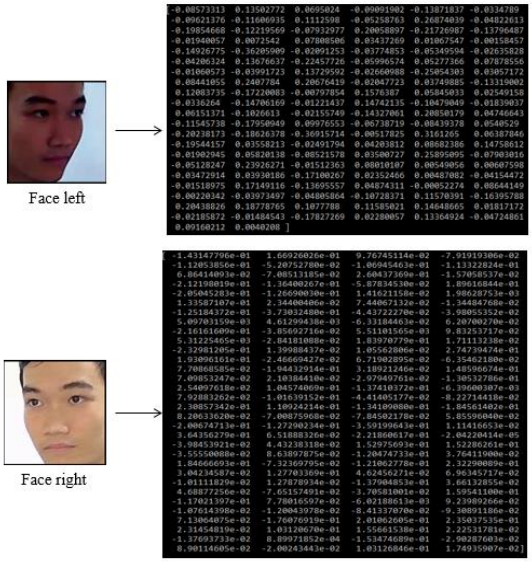

Figure 10. *Description of results using CNN to vectorize facial images.*

Using k-NN for classification to determine which person the combined vector belongs to. The experiment used 60 known faces as the training dataset for the KNN classifier and 20 faces from the collected data**,** along with the CASIA WebFace dataset as the test dataset**,** to assess accuracy.

3.3. Simulation results and comments

Parameters set for the MLP network: Number of layers: 3 layers with one input layer, one output layer, and one hidden layer; Number of neurons in the hidden layer: 124 neurons; Learning_rate parameter: 0.001; Number of iterations: 200. Threshold parameters for MLP and k-NN to identify who the tested face is in the system: the coefficient η in the experiments was: η = 0.3; η = 0.5.

3.3.1. Experimental results with self-collected data

The experimental data consisted of 200 image pairs (left, middle, right) of 20 faces.

Experiment with MLP: Training set: 145 pairs; Test set: 45 pairs. The images in the test set did not appear in the training set. After completing the training for the MLP network, the testing step was conducted for all 200 image pairs, checking the distance between the left-right image vectors and the post-combination vector against the frontal image of each pair for comparison.

Experiment with k-NN: Using k-NN for classification to determine which person the combined vector belongs to. Training set: 20 sets of frontal facial images of 20 people from self-collected data. Test set: 25 left-right image pairs of people in the system. 26 left-right image pairs of people not in the system. These pairs were combined through the MLP network obtained in the previous step to serve as input for k-NN.

Table 1. *Results with MLP based on self-collected data.*

	Left Image	**Right Image**	**Combined**
η	0.35	0.35	0.3
Incorrect results	51	44	3
Correct results	149	156	197
Rate (%)	74.50	78.00	98.50
	Left Image	**Right Image**	**Combined**
η	0.5	0.5	0.35
Incorrect results	9	6	1
Correct results	191	194	199
Rate (%)	95.50	97.00	99.50

Table 2. *Results with KNN based on self-collected data.*

	Left Image	**Right Image**	**Combined**
η	0.35	0.35	0.3
Incorrect results	11	8	1
Correct results	40	43	50
Rate (%)	78.43	84.31	98.04
	Left Image	**Right Image**	**Combined**
η	0.5	0.5	0.35
Incorrect results	2	1	1
Correct results	49	50	50
Rate (%)	95.50	98.04	98.04

3.3.2. Experimental results with CASIA data

The experimental data consisted of 117 image pairs (left, middle, right) of 25 faces.

Experiment with MLP: Training set: 83 pairs; Test set: 34 pairs. After completing the training for the MLP network, the testing step was conducted for all 117 image pairs, checking the distance between the left-right image vectors and the post-combination vector against the frontal image of each pair for comparison.

Table 3. *Results with MLP based on CASIA data.*

	Left Image	**Right Image**	**Combined**
η	0.45	0.45	0.45
Incorrect results	57	48	13
Correct results	60	69	104
Rate (%)	51.28	58.97	88.89
	Left Image	**Right Image**	**Combined**
η	0.5	0.5	0.5
Incorrect results	31	21	7
Correct results	86	96	110
Rate (%)	73.50	82.05	94.02

Experiment with k-NN: Training set: 25 sets of frontal facial images of 25 people from self-collected data. Test set: 25 left-right image pairs of people in the system. 26 left-right image pairs of people not in the system. These pairs were combined through the MLP network obtained in the previous step to serve as input for k-NN.

Table 4. *Results with KNN based on CASIA data.*

	Left Image	**Right Image**	**Combined**
η	0.45	0.45	0.45
Incorrect results	6	4	3
Correct results	45	47	48
Rate (%)	88.23	92.16	94.12
	Left Image	**Right Image**	**Combined**
η	0.5	0.5	0.5
Incorrect results	3	2	2
Correct results	48	49	49
Rate (%)	94.12	96.07	96.07

With the experimental steps on the self-collected datasets and the CASIA test dataset, the results show that combining images of the face from different angles for facial recognition yields good results. The experimental results show that the method of combining two angled images is relatively good and better than the results when using only one left or right image. However, since the self-collected dataset is not yet large enough, and the CASIA dataset only includes faces from different angles but not at the same time of data collection, the results still need more experiments to accurately assess the effectiveness of the method.

4. CONCLUSIONS

The proposed method has utilized and combined machine learning techniques and recognition algorithms for facial recognition problems:

1) Detection and extraction of facial images appearing in videos and photos.

2) Using deep learning and Convolutional Neural Networks (CNN) for Encoding and vectorizing facial features.

3) Artificial neural networks are employed to synthesize vectors of faces obtained from two different angles into a composite vector for recognition.

The research results show that using the method of combining two angled images is better than using a single image taken from the left or right.

The current study is limited to understanding and experimenting with images collected from cameras, which is more accurate than processing static images, and has not yet dealt with real-time processing for direct recognition from operating camera videos. This is also the direction for future development of the topic that the authors wish to continue researching and applying.

With this solution, recognition systems need to be equipped with more cameras than other systems, and the cameras also need to have standard specifications in terms of resolution, frame rate, and the synchronized equipment of cameras will bring higher efficiency.

In terms of application, this solution can be applied to security control, entry and exit control at buildings, locations requiring security and confidentiality, and identity verification at control gates.

REFERENCES

[1]. Zhao W., Krishnaswamy A., Chellappa R., Swets D.L., Weng J. *"Discriminant Analysis of Principal Components for Face Recognition"*. In: Wechsler H., Phillips P.J., Bruce V., Soulié F.F., Huang T.S. (eds) Face Recognition. NATO ASI Series, vol 163. Springer, (1998).

[2]. Guodong Guo, Stan Z. Li, Kap Luk Chan., *"Support Vector machines for face recognition"*, School of Electrical and Electronic, Nanyang Technology University, (2001).

[3]. Liao, S., Jain, A. K., Li, S. Z., *"Partial face recognition: Alignment-free approach"*, IEEE Trans. PAMI, 35(5):1193–1205, (2013).

[4]. Y. Taigman, M. Yang, M. Ranzato, L. Wolf, *"DeepFace: Closing the Gap to Human-Level Performance in Face Verification"*, Proc. of Computer Vision and Pattern Recognition Conference (CVPR 2014), Columbus, (2014).

[5]. Y. Taigman, M. Yang, M. Ranzato and L. Wolf, *"DeepFace: Closing the Gap to Human-Level Performance in Face Verification,"* IEEE Conference on Computer Vision and Pattern Recognition, Columbus, OH, USA, (2014).

[6]. A. Verma, D. Malla, A. K. Choudhary and V. Arora, *"A Detailed Study of Azure Platform & Its Cognitive Services"* International Conference on Machine Learning, Big Data, Cloud and Parallel Computing (COMITCon), Faridabad, India, (2019).

[7]. V. Sharma, *"Object Detection and Recognition using Amazon Rekognition with Boto3"*, 6th International Conference on Trends in Electronics and Informatics (ICOEI), Tirunelveli, India, (2022).

[8]. Baojin Huang, Zhongyuan Wang, Guangcheng Wang, Kui Jiang, Zhen Han, Tao Lu, Chao Liang, *" PLFace: Progressive Learning for Face Recognition with Mask Bias"* College of Information Engineering, Northwest A&F University, (2023).

[9]. Xiaopeng Li, Yuyun Xiang, Shuqin Li, *"Combining convolutional and vision transformer structures for sheep face recognition "* Computers and Electronics in Agriculture, vol. 205, p. 107651, (2023).

[10].N. Dalal and B. Triggs, *"Histograms of oriented gradients for human detection"* IEEE Computer Society Conference on Computer Vision and Pattern Recognition, (2005).

TÓM TẮT

Phương pháp kết hợp nhiều góc nhìn để tăng độ chính xác cho bài toán nhận dạng khuôn mặt

Bài báo này giới thiệu một phương pháp tiên tiến trong lĩnh vực nhận dạng khuôn mặt, sử dụng một kỹ thuật độc đáo kết hợp mạng Convolutional Neural Networks (CNN) và Multilayer Perceptron (MLP) để tích hợp nhiều góc nhìn khác nhau. Điểm nhấn của phương pháp này là việc áp dụng CNN để phân tích đặc điểm hình ảnh từ nhiều góc độ, cùng với MLP nhằm tối ưu hóa quá trình tổng hợp thông tin, qua đó nâng cao độ chính xác trong nhận dạng khuôn mặt dưới các điều kiện ánh sáng và góc độ biến đổi. Mục tiêu chính là giải quyết thách thức về sự suy giảm hiệu suất nhận dạng khuôn mặt trong các tình huống thực tế, đặc biệt khi góc nhìn có sự thay đổi lớn. Nghiên cứu này chi tiết cách xây dựng mô hình từ thu thập và xử lý dữ liệu, huấn luyện mạng lưới nơ-ron phức tạp, đến việc đánh giá hiệu quả thông qua các bộ dữ liệu tiêu chuẩn và thực nghiệm.

Từ khóa: Nhận dạng khuôn mặt; Học sâu; Mạng lưới nơ-ron tích chập; Kết hợp nhiều góc nhìn; Xử lý ảnh; Tối ưu hóa góc nhìn; Phân tích đa góc nhìn; Cải thiện hiệu suất nhận dạng; Ứng dụng an ninh.