Mathematical Biosciences and Engineering

MBE, 20(6): 11081–11100.
DOI: 10.3934/mbe.2023490
Received: 11 January 2023
Revised: 11 April 2023
Accepted: 13 April 2023
Published: 23 April 2023

http://www.aimspress.com/journal/MBE

Research article

Finger vein recognition method based on ant colony optimization and improved EfficientNetV2

Xiao Ma and Xuemei Luo[*]

School of Electrical Engineering, Guizhou University, Guiyang 550025, China

*** Correspondence:** Email: mx990821@163.com.

Abstract: Deep learning is an important technology in the field of image recognition. Finger vein recognition based on deep learning is one of the research hotspots in the field of image recognition and has attracted a lot of attention. Among them, CNN is the most core part, which can be trained to get a model that can extract finger vein image features. In the existing research, some studies have used methods such as combination of multiple CNN models and joint loss function to improve the accuracy and robustness of finger vein recognition. However, in practical applications, finger vein recognition still faces some challenges, such as how to solve the interference and noise in finger vein images, how to improve the robustness of the model, and how to solve the cross-domain problem. In this paper, we propose a finger vein recognition method based on ant colony optimization and improved EfficientNetV2, using ACO to participate in ROI extraction, fusing dual attention fusion network (DANet) with EfficientNetV2, and conducting experiments on two publicly available databases, and the results show that the recognition rate using the proposed method on the FV-USM dataset reaches The results show that the proposed method achieves a recognition rate of 98.96% on the FV-USM dataset, which is better than other algorithmic models, proving that the method has good recognition rate and application prospects for finger vein recognition.

keywords: finger vein recognition; ROI extraction; ant colony optimization; DANet; EfficientNetV2

1. Introduction

In the field of biometrics, technologies such as fingerprint, face and iris recognition have emerged. These biometric techniques have their advantages and disadvantages. For example, face recognition is simple and convenient, fingerprint recognition is cheap, iris recognition has a high security factor, but face recognition is easily affected by factors such as plastic surgery, fingerprints are easily damaged and stolen, and iris recognition is relatively expensive and difficult to promote. Contrary to the characteristics of this technology, biometric recognition is still vital, not only reliable and not easy to be copied, but also low cost and very easy to collect, which has become a popular direction for the future development of real identification industry.

Finger vein recognition is a new type of biometric identification method, which uses the superficial vein pattern under the skin of the finger area as an identification feature for identification. The vein network has the advantages of in vivo, not easy to forge, and portability, etc. The finger vein is known as the key hidden in the finger, and it makes up to a certain extent for the shortcomings of other biometric features that are easily affected by various external environmental factors such as angle, light, and shadow. Compared with the above biometric authentication technology, the use of finger vein features for identification has many advantages, as follows:

1) In vivo identification and high anti-counterfeiting. The finger vein image must be collected in the “live” state, which is a living human characteristic. The vein pattern and texture features are extracted from the finger vein image for authentication and identification, effectively avoiding the bad situation of identity information loss, theft and forgery.

2) Internal features. The finger vein is located inside the human body, and the finger surface or external environment will not affect the finger vein recognition result.

3) High usability. It is suitable for all people and the hemoglobin in finger veins is sensitive to infrared light and easy to collect.

4) High accuracy. Finger vein features have uniqueness, and each person’s finger vein pattern is different, so using finger vein as a biometric feature for identification has a high accuracy rate of authentication.

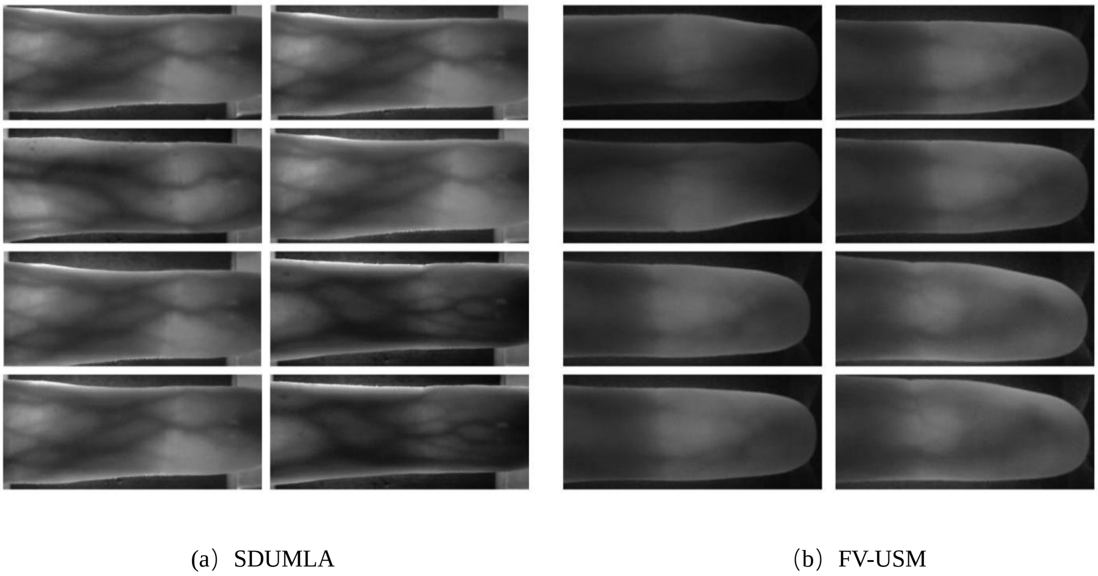
5) Permanence. The existence of finger vein pattern is permanent and does not change with the age or time of the person.The advantages such as these make finger vein identification technology research and development present a rapid booming trend in recent years, but there are still many challenges and room for improvement for the existing finger vein identification technology development status quo if we want to achieve realistic scenario application deployment of finger vein feature identification system and improve identification performance. For example, the sample image quality collected by finger vein collection equipment is unstable and easily blurred, and the position of user's finger is not fixed when the vein sample is collected, which easily leads to the deflection of the collected finger vein image and other problems, causing the finger vein recognition algorithm to reduce the recognition rate, and even making the extracted finger vein features easily fail to match. In response to the above-mentioned problems that exist and need to be solved, experts and scholars have proposed the need to study and adopt various complex image processing algorithms to overcome the noise and uncertainties that are easily generated in the process of these captured vein images, and when faced with the above problems, they are now divided into two major types of solutions: traditional feature extraction methods and deep learning, and existing research proves that with the enrichment of finger vein dataset types and data enhancement, the Deep learning models combined with finger vein recognition can achieve higher efficiency than traditional feature extraction. After 2015, with the rapid development of deep learning methods and

their successful application in the field of computer vision, the "deep learning + finger vein recognition" method has gradually become a hot topic in this field. 2016, Radzi et al. of Malaysia [1] used a simple CNN with 4 layers to test on small sample datasets of 50 and 80 objects, respectively. were tested and achieved high recognition rates; in 2018, Das et al. in Italy [2] proposed a 10-layer CNN, trained on four publicly available datasets, respectively, with recognition rates above 95%; in 2020, Noh et al. in Korea [3] used DenseNet-161 to train on finger vein texture images and finger vein shape images, respectively, and fused the two output scores of CNNs were fused for recognition. Zhao et al. [4] used a lightweight CNN for classification and focused on the loss function by using a central loss function and dynamic regularization. Hao et al. [5] proposed a multi-task neural network that performs ROI and feature extraction sequentially through two branches. On the other hand, Lu et al. [6] proposed the CNN competitive Order (CNN-CO) local descriptor, which is generated by using CNN pre-trained on ImageNet. After the effective CNN filter is selected from the first layer of the network, CNN-CO calculates the image filtered by CNN, constructs the competitive order image, and finally generates the CNN-CO pyramid histogram. Finally, Kuzu et al. [7] studied the application of transfer learning by using the pre-training CNN model trained on the ImageNet data set, and achieved satisfactory results. The literature [8] used CNN models to extract feature vectors from finger vein images and perform classification. CNN methods are robust to image quality in feature extraction and classification. However, the drawback of these methods is that the determination of CNN structure needs to undergo a lot of experiments. The designed network also requires a large number of images for training to achieve the expected results, which poses training difficulties. In addition, the large number of trainable parameters in CNNs and the long training time are not suitable for real-time recognition of finger veins. Zhang Y. et al [9] combined CNN models with Gabor filters in the field of biometric recognition to propose a method in which the employed method is able to learn Gabor filter parameters adaptively. The current study pointed out that the CNN method for finger vein recognition based on deep learning consists of the following steps: image acquisition, preprocessing, feature extraction, classification, and validation [10,11]. Among them, the CNN is the most central part that can be trained to obtain a model capable of extracting finger vein image features. In closed-set experimental schemes, dataset partitioning is usually used to divide the dataset into training, validation, and test sets, so that model training, tuning, and evaluation can be performed [12,13]. Liu [14] et al. used a seven-layer CNN, which includes five convolutional layers and two fully connected layers. The biggest challenge is how to protect the biometric data while maintaining the practical performance of the identity verification system. To solve this problem, Liu [15] et al. proposed a novel finger vein recognition algorithm based on deep learning and random projection for secure biometric template scheme called FVR-DLRP. To address this problem Yang et al. [16] used binary decision diagrams (BDD) to develop a new biometric template protection algorithm for deep learning based finger vein biometric systems. Zhang et al. [17] proposed a lightweight fully convolutional generative adversarial network (GAN) architecture called FCGAN that uses preliminary batch normalization and tightly constrained loss functions to achieve finger vein image enhancement. Zhang et al. [18,19] studied finger vein recognition based on sub-convolutional neural networks. The convolutional layer uses the LeakyReLU activation function and the pooling layer uses the maximum downsampling method. Sidiropoulos et al. [20] aimed to briefly review the finger vein recognition applied to the feature extraction methods. Other influential works include Daas et al. [21]. Zhang J et al. [22,23] proposed a novel low-cost U-Net (LCU-Net) for environmental microorganism (EM) image segmentation task to help microbiologists detect and identify EM more efficiently. LCU-Net is an improved convolutional neural network (CNN) based on U-Net, initial and connection operations. It solves the limitation of single U-Net receptive field setting and the problem of high storage cost. After experiments, the experimental results show that the proposed LCU-Net has some effectiveness and potential in the field of practical EM image segmentation and introduces an automatic image analysis method based on artificial neural networks to optimize it. However, the automatic analysis of biological images faces many challenges, such as the diversity of applications leading to the requirement of robustness of the algorithm, the insignificance and easy under-segmentation of image features, and the diversity of analysis tasks. Therefore, the characteristics of neural network-based biological image analysis are reviewed. The introduction of deep learning model in finger vein detection also has an obvious disadvantage that it lacks a large number of data sets, but the corresponding solution is data enhancement technology. More and more studies have proved that deep learning can achieve very good results. In this paper, we propose a new finger vein recognition method based on a dual attention fusion network jointly EfficientNetV2 (DAF-EfficientNetV2) and ACO, ACO is used because of its excellent edge detection performance using the Mini-ROI method to determine the left and right boundaries and the upper and lower boundaries, and then we use DAF-EfficientNetV2 to train the pre-processed images, and experiments using the FV-USM database show that the recognition rate of the method is up to 98.96%, which is better than the current mainstream advanced algorithms.

2. Finger vein recognition method

Biometric recognition needs to be based on a certain recognition system, in finger vein recognition, a typical biometric system includes 3 stages: ROI extraction, model design and training, and experimental comparison, the following subsections describe in detail all the stages of the finger vein recognition system proposed in this paper for personal identification and verification.

2.1. Introduction to ROI extraction



(a) SDUMLA (b) FV-USM

Figure 1. Example of finger vein image in NIR light, image taken from SDUMLA and FV-USM databases.

The main region of the extracted biometric features is called ROI, Figure 1 shows the ROI images in 320 × 240 and 640 × 480 size in near infrared light. These images are taken by a charge coupled element (CCD) consisting of a high definition camera, A/D converter, near infrared light source and lens. The main purpose of preprocessing is to focus the original image on a smaller region such as ROI, and to represent the features by more complex and differentiated way to represent the features, used to facilitate the comparison and recognition phase, many techniques have been proposed in the above mentioned studies, we propose ACO in our research method because it brings clear edge information, in the next section we will present this method.

2.2. Edge detection based on Ant Colony algorithm

Ant Clony Optimization (ACO) is a swarm intelligence algorithm, which is a group of unintelligent or slightly intelligent individuals (Agents) that exhibit intelligent behavior by collaborating with each other, thus providing a new possibility for solving complex problems. The ant colony algorithm was first proposed by Italian scholars Colorni A., Dorigo M., etc. in 1991. After more than 20 years of development, the ant colony algorithm has made great progress in theoretical as well as applied research. The ant colony algorithm is a bionic algorithm inspired by the foraging behavior of ants in nature. In nature, during ant foraging, the colony is always able to follow to find an optimal path from the nest to the food source. Figure 2 shows such a foraging process. There is a colony of ants, if A is the nest and B is the food source (and vice versa). The colony will travel along a straight line path between the nest and the food source. If there are two long and short routes between A and B, then the ants at point A will have to make a decision whether to travel left or right. Since there is no pheromone left on the road at the beginning, the ants have an equal probability of traveling in both directions. However, when an ant passes by, it will release pheromones in its path and the pheromones will be emitted at a certain rate. Pheromone is one of the tools for communication among ants. The ants behind it make a decision to go left or right by the concentration of pheromones on the road. It is obvious that the pheromone will be more and more concentrated along the short side of the path, thus attracting more and more ants to travel along this path.

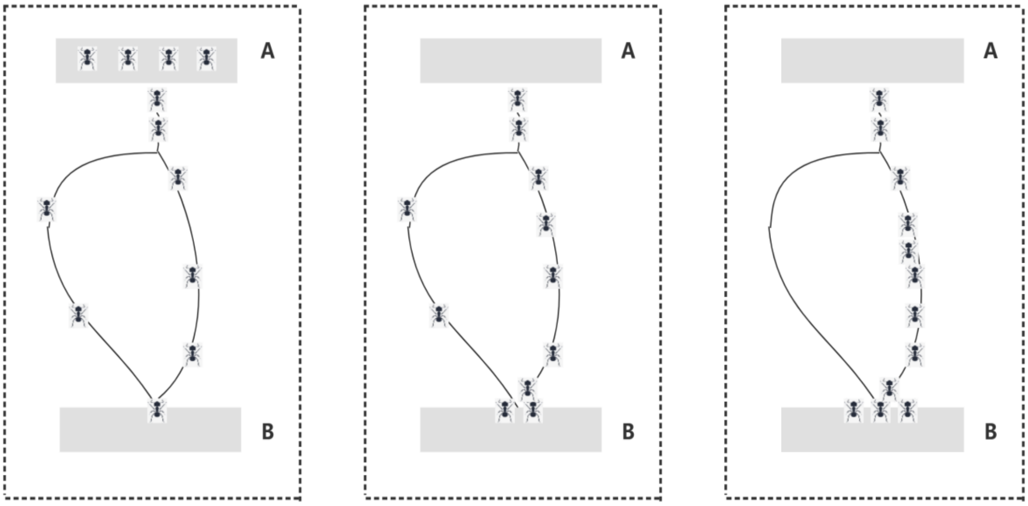


Figure 2. Schematic diagram of the ant colony principle.

The main steps of image edge detection using ant colony algorithm are as follows:

1) Mathematical model of image

Assume that the image to be detected is a grayscale image I(i, j) with a size of M × N. Put the data structure on. The image is regarded as an undirected graph, as shown in Figure 3 below:

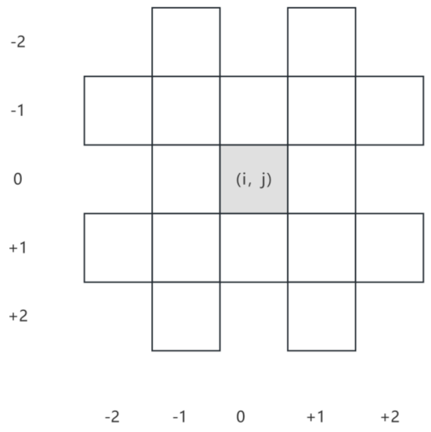


Figure 3. Connected (i,j) pixel structure diagram.

During construction, a group of artificial ants builds solutions ranging from limited solution components to full. The connected graph represents the problem to be solved. There are a number of build steps involved in the build process. Ants in Move back and forth on the image until each ant has reached the target number in the build step. The solutions to the construction process starts with a partial solution and add solutions to get results at each step. The solution is to select a set of adjacent nodes from the node at the current position in the image. The selection of ingredients is done according to a certain probability.

2) Initialization process

During initialization, m ants are randomly placed in the image, and each ant is assigned to the M × N image. Any location. The greater the number of ants m, the faster the algorithm's search will become, but it will increase the consumption. So you need to choose the right m value. The value of the ant number m depends on the actual problem and the image. Size for different settings, the more complicated the problem, the bigger the image, the more ants need to be used. Usually desirable ant number $\sqrt{\mathrm{M} \times \mathrm{N}}$, this can be in a relatively short time guaranteed number.Target preferred edge detected. The image consists of a composition of background, target and edge, and their characteristics are reflected. On the pixel gray gradient value, you can set a gradient threshold to control them, will be as much as possible. Ants flock near or on the edge to speed up their search. In order to successfully activate the ants, The original value of each element in the pheromone matrix is set to the constant τ_{init}, with a small but non-zero value. In addition, Heuristic information matrix is strength value based on local variation. Heuristic information is determined during initialization, because it depends only on the value of that pixel image, it is continuous. $\mathrm{V_c(I_{i,j})}$ is a function that acts on a local set of pixels at the pixel at (i, j), given by Figure 3.

$$V_c(I_{i,j}) = |I_{i-2,j-1} - I_{i+2,j+1}| + |I_{i-2,j+1} - I_{i+2,j-1}| + |I_{i-1,j+2} + I_{i+1,j+2}| + |I_{i-1,j+2} - I_{i+1,j-2}| + |I_{i-1,j-1} - I_{i+1,j+1}| + |I_{i-1,j+1} - I_{i-1,j-1}| + |I_{i-1,j} - I_{i+1,j}| + |I_{i,j-1} - I_{i,j+1}| \quad (1)$$

The heuristic information in pixels (i, j) is determined by local statistics, whose position is

$$\eta_{i,j} = \frac{V_c(I_{i,j})}{V_{max}} \tag{2}$$

Where 1 is the intensity value on the pixel (i, j).V (i, j) C is a function around the pixel (i, j), in the image. Run on the prime local group (Figure 3).

3) Ant path selection

Each ant selects the position to be moved by calculating the probability in the 3 × 3 neighborhood pixel according to Formula 3. Suppose the KTH ant is in the position of i, and j is a pixel adjacent to point i, then the probability of this ant choosing vertex j is:

$$P_{ij}^{k}(t) = \begin{cases} \dfrac{\left[\tau_{ij}(t)^{\alpha}\cdot[\eta_{ij}]^{\beta}\right]}{\Sigma[\tau_{ij}(t)^{\alpha}]\cdot[\eta_{ij}]^{\beta}} \\ l \notin l_i^k \cap D \\ 0, j \notin J_i^k \end{cases} \tag{3}$$

τij (t) represents the pheromone intensity, t represents the number of iterations, 7 is the heuristic guide function, the value of pixels in the gradient of $\eta_{\mathrm{ij}} = \nabla \mathrm{I}_{\mathrm{j}}$.

α is a parameter used to control the ant’s degree of exploration, and β is a parameter that represents the influence of pheromone cues. If α = 0, the maximum pixel gray gradient is selected, and the algorithm degenerates into a random greedy algorithm. The larger the value of α, the ants will be more likely to choose the path taken by other ants. If β = 0, only the re-action of pheromone enhancement mechanism produces the algorithm result. The non-optimal solution obtained in the shortest time is obtained. The higher the value of β, the more likely the ant will choose the neighborhood point with high gradient value. Therefore, a compromise relationship should be found between the degree of control the ants explore the parameters and the influence of the control parameters on tracking pheromones.

4) pheromone renewal

Local pheromone updates are applied at various times during the build process, after the ant crosses the edge. After each build step, the ant will update the pheromone value associated with it crossing the last edge, according to Formula 4:

$$\tau_{ij}(t) = (1-\rho)\cdot\tau_{ij}(t) + \Delta\tau_{ij}(t) \tag{4}$$

Where, τij (t) is the sum of the excitins released by the ant moving to j, and its value $\Delta\gamma_{\mathrm{ij}}(\mathrm{t}) = \sum \Delta\gamma(\mathrm{t})$. When the first k ant select vertices $\Delta\gamma_{\mathrm{ij}}^{\mathrm{k}}(\mathrm{t}) = 0$,when the first k ants want to vertex j, in order to make the ant search convergence speed was improved, its value is set to vertex j of gradient function, namely, which is suitable for constant C. In the absence of pheromone evaporation on the path, if the search is likely to lead to a dramatic change in the initial value, which leads to the final test results are not good, in order to guarantee the complete solution space, the pheromone volatilization mechanism must be introduced into the algorithm design, introducing the pheromone volatilization coefficient. P (0 < P > 1) If the ant does not select this vertex, the pheromone at this vertex will gradually disappear with the increase of time.

5) Termination conditions

When the ant completes the specified number of steps, the final pheromone matrix is used to classify each pixel, any edge or non-edge, at the end of the algorithm. Then the number of pixels per ant passes through the post-image, and a threshold is set for comparison to determine the target edge, the threshold calculation method is called Otsu threshold technique. The algorithm iterates ant walking steps, which are affected by the number of ants, the size of the image and the complexity of the image. Examples of the original image and the image after using ACS and eliminating isolated points are shown in Figure 4, with the chosen parameters of k = 816, (340 × 240 ROI image) and τ_{init} = 0.1.

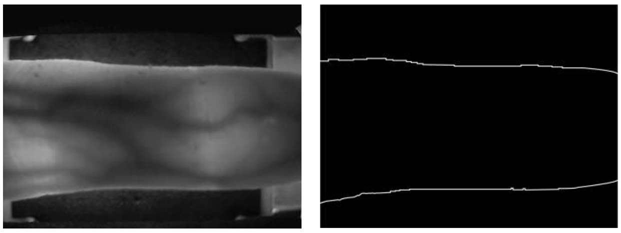


Figure 4. Original image and image after eliminating isolated points.

2.3. Mini-ROI extraction

The task discussed here is how to determine the left and right boundaries and the upper and lower boundaries of the ROI, an ideal method for detecting the region of interest of finger vein features should finally select the ROI that contains as much feature information as possible, but the area is as small as possible and the core vein features. Therefore, this paper proposes a Mini-ROI extraction method, which we will introduce in detail next.

1) Determining the left and right boundaries

Relevant biomedical reports show that when the finger is irradiated with NIR light, the joint area in the image is brighter because the tissue fluid in the joint space of the finger is more penetrating compared to the bone. Thus, the finger joints can be accurately located by the difference in image brightness, i.e., the brightest area in the image should be the finger joint area, and the left and right boundaries can be determined based on the finger joint position.

All pixels in each column of the image I(x,y) of size I(x,y) are summed up, and the summation operation is defined as follows:

$$\delta_i = \sum_{i=1}^{a} I(i,j), j = 1,2,\ldots,b \tag{5}$$

The sum of each column is counted and the maximum sum is obtained, i.e., the position of the peak point, which determines the corresponding position of the corresponding finger joint.

$$\text{Col}_{\max} = \arg\max(\delta_i), i \in [1, a] \tag{6}$$

Based on this method, the positions of the finger joints are obtained. Figure 5 shows the accumulation of the pixel values of each column of pixels of a finger vein image. It is clear that there are two peak points of the sum of the left and right pixel values, i.e. the two brightest regions, which correspond to the two knuckles of the finger. By recording their columns, it is easy to locate the two knuckles.

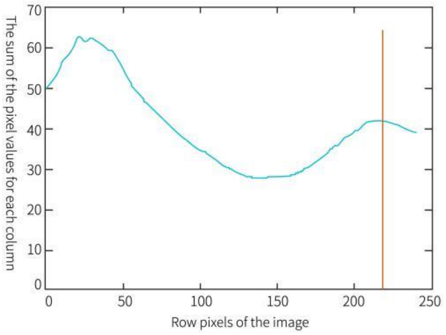


Figure 5. Cumulative sum of pixel values for each column of the image.

In this case, when selecting the left and right boundaries, it is possible to intercept the area between two knuckles on the one hand, and the area on both sides of a particular knuckle on the other. However, according to the database SDUMLA, it is easy to find that most of the finger joints near the right edge (i.e., near the fingertip) are detected as peak points, while the left finger joints far from the fingertip are not clearly detected in some images due to illumination and other reasons. The peak points are not detected clearly in some of the images due to illumination and other reasons, so that the peak point detection fails. Therefore, in order to ensure the stability of the same database processing, the knuckle positions near the fingertip can be detected uniformly when processing this database, and the accumulation range of the columns can be reduced to [0.4a,0.8a] according to the characteristics of the database, so as to reduce the calculation of invalid areas. Finally, using the knuckle position as the reference line, a certain length is intercepted from both sides proportionally, and the rotationally corrected image and the intercepted image after determining the left and right boundaries are shown in Figure 6.

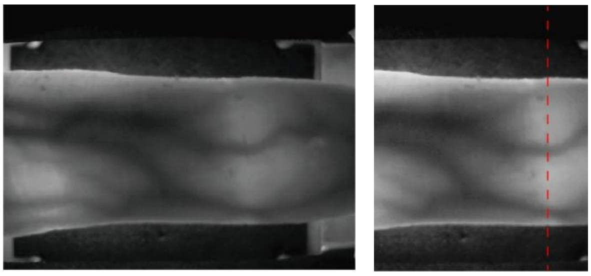


Figure 6. Determining the left and right boundaries of the Mini-ROI.

2) Determining the upper and lower boundaries

In the region binary image after rotation correction we can get the coordinates of the upper and lower edge points, by which we can determine the determination of the upper and lower edges of Mini-ROI. Firstly, in the binary image of the finger edge after rotation correction, we randomly select the vertical coordinates of some of the upper edge points to find the average value y_{up} by using a fixed reasonable interval, considering that the finger edge points are discrete and have high and low values, in order to reduce the error, we choose the value of $y_{up} - 2$ as the upper edge vertical coordinate value, which is written as y1: Similarly, then we select the vertical coordinates of some of the lower edge points to find the average value y_{low} and choose the value of y_{low}–2 as the After obtaining the upper and lower edges of the target region, the final finger vein Mini-ROI region of interest image is obtained, and the final ROI is obtained after image enhancement, and the example of the Mini-ROl region intercepted from the externalized finger vein sample and the whole preprocessing process are shown in Figures 7 and 8.

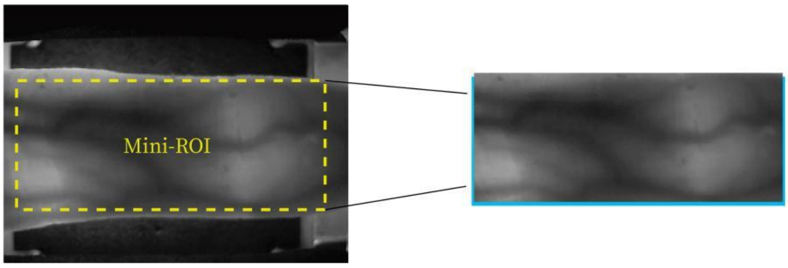


Figure 7. Mini-ROI area.

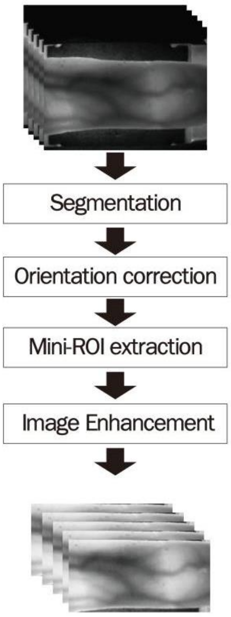


Figure 8. Complete pre-processing flow.

3. Finger vein detection based on the EfficientNetV2 network

Finally, this paper proposes a finger vein detection model based on EfficientNetV2, which was proposed by the Google team in 2021 and became the optimal model of that year. The feature extraction part was tested using the Fused-MBCon module and the deep MBConv module and tested on the public dataset on ImageNet. Its structure is shown in Figure 9. This document uses an EfficientNet V2-s configuration for testing, networking, the architecture is shown in Table 1, with a training input size of 300 × 100.

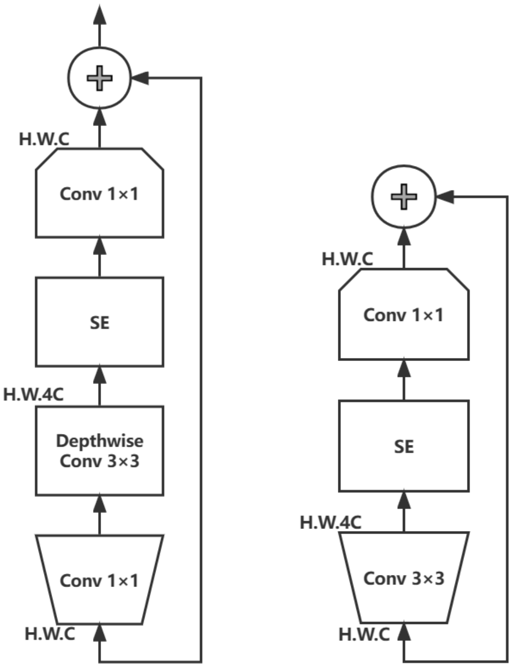


Figure 9. Structure of MBConv and Fused-MBConv.

Table 1. Network structure of EfficientNetV2-s.

Stage	Operator	Stride	#Channels	#Layers
0	Conv3 ×3	2	24	1
1	Fused-MBCnov1，k3 ×3	1	24	2
2	Fused-MBCnov4，k3 ×3	2	48	4
3	Fused-MBCnov4，k3 ×3	2	64	4
4	MBCnov4，k3 ×3，SE0.25	2	128	6
5	MBCnov6，k3 ×3，SE0.25	1	160	9
6	MBCnov6，k3 ×3，SE0.25	2	256	15
7	Cnov1 ×1 & Pooling & FC	-	1280	1

4. Finger vein recognition model based on improved EfficientNetV2

In this study, the purpose of building a high-precision and low-consumption finger vein recognition model was carried out with the lightweight network EfficientV2 as the backbone network. Considering the wealth of information at the shallow level of the network, the network is fused using dual attention and the shallow modules are updated.

4.1. Dual attention network

DANet applies the attention mechanism to semantic segmentation, mainly based on Position-based attention and Channel-based attention to obtain rich context information and improve the segmentation accuracy of the model. Postion-based attention module makes features anywhere on the feature map related to all other locations, and similar pixels should still maintain their close relationships (even if the distance is far away, the advantage of attention). The channel-based attention module can capture the channel dependencies between any two channel graphs and use the weighting of all channel graphs and update each channel graph. By adding these two attention modules, the feature representation of the model can be further enhanced and the model segmentation performance can be improved.

The images are fed into a feature extraction network with ResNet as the backbone, similar to Deeplab, removing the downsampling operation of the fourth and fifth modules and replacing pooling with hole convolution instead. Therefore, the feature map output of the feature extraction network has changed from 1/32 to 1/8. The feature map generated by the feature extraction network is fed into two parallel attention modules, and the dependency between the feature map location and the channel is obtained and the information of the two is fused into the classifier by element addition operation.Details of each module are described below.

The position attention module is shown in the Figure 10, A (C × H × W) is converted into B (C × H × W), C (C × H × W), D (C × H × W) by different convolutional layers. B/C/D reshape is (C × N) size, denoted B'/C'/D', where N = H × W represents the number of pixels per channel. Multiply the transpose of B' with the dot multiplication of the C' matrix (the geometric meaning of the point multiplication of vectors is to calculate the similarity of two vectors, the more similar the two vectors, the larger their point multiplication) and the Softmax normalization operation to obtain the feature map S (N × N contains the similarity information of the pixel position).

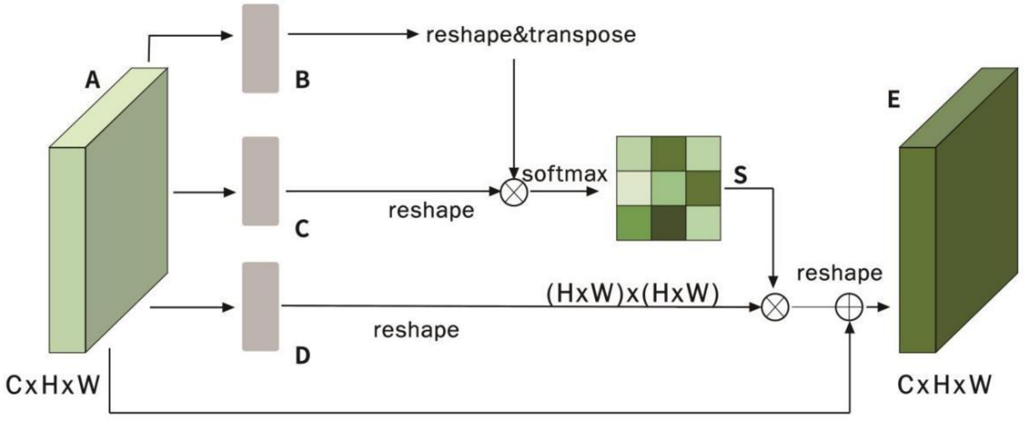


Figure 10. Position attention module.

$$s_{ji} = \frac{\exp(B_i \cdot C_j)}{\sum_{i=1}^{N} \exp(B_i \cdot C_i)} \tag{7}$$

Sji represents the effect of the i-th position pixel on the jth position, that is, the degree of association/correlation between the ith position and the jth position, the larger the more similar. S and D' matrix points are multiplied to get S', and the correlation factor is multiplied by the feature map to obtain the correlation. S' and the original figure A are added by elements to obtain E, and the relevant information is fused into the original figure.

$$E_j = \alpha \sum_{i=1}^{N} (s_{ji} D_I) + A_i \tag{8}$$

Alpha is the proportion of adjusted relevance, which is continuously updated through training iterations. This module can obtain a global perspective, selectively aggregating spatial context aggregations. Similar feature pixels have the same uplift and growth, and will not be dominated by neighboring pixels (different classes), solving the problem of inconsistency within classes caused by convolution operations.

The channel attention mechanism is shown in the Figure 11, reshape (C × N) and reshape and transpose (N × C) for A, respectively. Multiply the two feature maps obtained and get the channel attention map X (C × C) through softmax. Then the transpose (C × C) of X and A (C × N) of reshape are multiplied by matrix, multiplied by the scale coefficient β, and then reshape to the original shape, and finally added with A to obtain the final output E, the β initializes to 0 and gradually learns to get a greater weight.

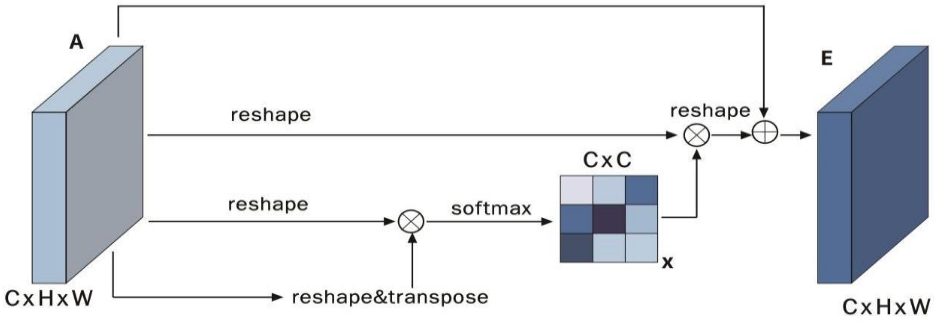


Figure 11. Channel attention mechanism.

$$x_{ji} = \frac{exp(A_i \cdot A_j)}{\sum_{i=1}^{C} exp(A_i \cdot A_j)} \tag{9}$$

$$E_j = \beta \sum_{i=1}^{C}(x_{ji}A_i) + \ A_j \tag{10}$$

4.2. D-MBConv module

The original EfficientNetV2 consists of shallow Fused-MBCnov and deep MBConv, and since the finger vein map contains a lot of spatial information, the shallow network in this paper uses the DANet dual attention fusion mechanism. Embed the DANet network in Fused-MBCnov and name the new module D-MBCnov, the detailed structure is shown in Figure 12.

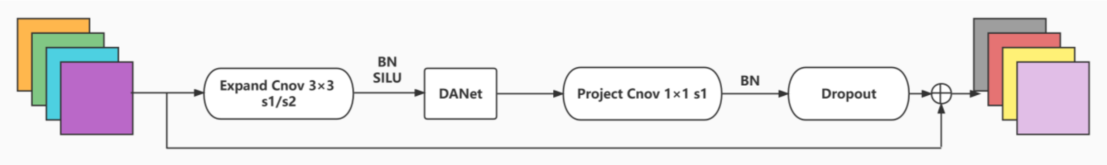


Figure 12. Structure of D-MBCnov.

4.3. EfficientNetV2 based on hybrid attention mechanisms

The D-MBConv module is used to replace the original shallow Fused-MBConv module of EfficientNetV2, and EfficientNetV2 based on the mixed attention mechanism is built, namely Double Attention Fusion-EfficientNetV2 (DAF-EfficientNetV2), and the overall structure of the model is shown in Figure 13. The new network not only considers the important features on the channel in the shallow layer, but also effectively captures a large amount of spatial feature information. Experiments show that the use of dual attention fusion network can effectively improve the classification accuracy and be more competitive with the performance of other mainstream networks.

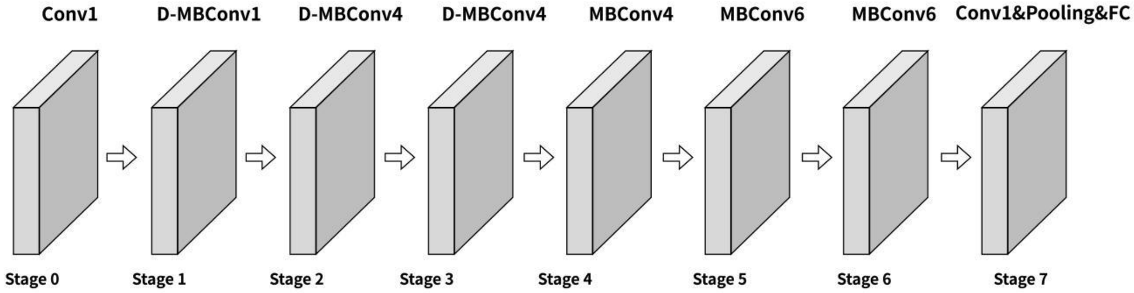


Figure 13. Structure of DAF-EfficientNetV2.

5. Experiment and results analysis

The experimental part of this paper is divided into three main parts. The first part aims to evaluate the performance of DAF-EfficientNetV2 on classification tasks by comparing the experiments with other popular CNN models on four publicly available datasets and presenting the experimental results through four comparison plots to determine its utility and practical application value. We compared popular CNN models such as ResNet-50, VGG-16, DenseNet-121, and MobileNet-V2, but due to the limitation of experimental computing resources, only the finger vein dataset (FV-USM) from Shandong University (SDUMLA) and Universiti Teknologi Malaysia was used for the experiments. We compared the classification performance of each model on these datasets and presented the experimental results in a table. The experimental conditions include Windows 10 operating system, NVIDIA GeForce GTX 1060 graphics card, Intel Core i7 7700HQ processor, Python version 3.7, and PyTorch as the framework. Regarding the two publicly available datasets used, the finger vein dataset from Shandong University divides it into 636 classes. The dataset is 106 volunteers' left and right hands with 3 fingers in each hand, and we divide each finger into one class. FV-USM were collected from 123 volunteers comprising of 83 males and 40 females, who were staff and students of Universiti Sains Malaysia. The age of the subject ranged from 20 to 52 years old. Every subject provided four fingers: left index, left middle, right index and right middle fingers resulting in a total of 492 finger classes obtained. The captured finger images provided two important features: the geometry and the vein pattern. Each finger was captured six times in one session and each individual participated in two sessions, separated by more than two weeks' time. In the first session。The ratio of training set and test set is divided according to 5:1.

The second part conducts ablation experiments for DAF-EfficientNetV2, comparing the performance of DAF-EfficientNetV2 under different data enhancement methods and different hyperparameters, and shows the experimental results with two comparison plots. This part of the experiment aims to evaluate the impact of the D-MBConv module on the performance of DAF-EfficientNetV2, and provide guidance and reference for the use of DAF-EfficientNetV2.

In deep learning, we usually need to evaluate the performance of a model to determine whether the model is effective or not. The most commonly used evaluation metrics are accuracy, precision, recall and f1score. Their meanings are described below.

Accuracy is one of the most basic classification metrics, which indicates the ratio of the number of samples correctly predicted by the model to the total number of samples. The specific calculation formula is:

$$\text{Accuracy} = \frac{TP+TN}{TP+TN+FP+FN} \tag{11}$$

where TP denotes True Positive, the number of samples correctly predicted as positive cases; TN denotes True Negative, the number of samples correctly predicted as negative cases; FP denotes False Positive, the number of samples incorrectly predicted as positive cases; and FN denotes False Negative, the number of samples that are incorrectly predicted as negative cases.

The higher the accuracy rate, the more accurate the model predicts the results. However, when the number of samples in different categories is not balanced, the accuracy rate may not reflect the performance of the model well.

Precision rate is the ratio of the number of samples correctly predicted by the model as positive cases to the number of all samples predicted as positive cases. The specific calculation formula is:

$$\text{Precision} = \frac{TP}{TP+FP} \tag{12}$$

A higher accuracy rate indicates that the model minimizes negative effects. Also, if we need to ensure that we do not incorrectly predict true counterexamples as positive (false positives), then we should focus on increasing the precision rate.

Recall is the ratio of the number of samples correctly predicted by the model as positive cases to the number of samples of all true positive cases. It is calculated by the following formula:

$$\text{Recall} = \frac{TP}{TP+FN} \tag{13}$$

The higher the recall, the better the model is at identifying positive class samples. When we need to ensure that all the true positive examples are found correctly, we can focus on increasing the recall rate.

F1 Score is a metric that combines Precision and Recall, and it is the summed average of the two. The specific formula is:

$$\text{F1 Score} = \frac{2PR}{P+R} \tag{14}$$

A higher F1 Score indicates that the model has achieved a good balance between Precision and Recall. When Precision and Recall are equally important, F1 Score can be a good evaluation metric.

To sum up, accuracy, precision, recall and f1score are important metrics to evaluate the performance of deep learning models. Different metrics are applicable to different scenarios, and the appropriate metrics need to be selected for evaluation according to the task requirements. In this experiment, we choose all the above metrics to evaluate the model.

The experimental part of this paper provides detailed data support and analysis, which provides strong support and guidance for the practical application of DAF-EfficientNetV2.

5.1. Algorithm comparison experiment

We compare DAF-EfficientNetV2 with popular CNN models such as ResNet-50, VGG-16, DenseNet-121 and MobileNet-V2, and conduct experiments using two publicly available datasets, Shandong University (SDUMLA) and the Finger Vein Dataset of Universiti Teknologi Malaysia (FV-USM), respectively. We compare their performance on the classification task and show the experimental results in four comparison plots.

The classification accuracy of each model on the two datasets:DAF-EfficientNetV2 was tested with 98.96% accuracy on FV-USM, while MobileNet-V2 and ResNet50 had 95.62% and 92.28% accuracy, respectively. We can find that DAF- EfficientNetV2 performs significantly better than the other models. The accuracy of all five models on the SDUMAL dataset decreases because this dataset contains a small portion of low-quality images. The above experimental results can provide guidance and reference for further optimization of the models, and also provide strong support and validation for the practical application of the models. Table 2 shows the test accuracies of the five models on two publicly available datasets, where the test accuracies of the models on FV-USM and the test accuracies of the models on SDUMLA are shown.

Table 2. Test accuracy of each model on two public datasets.

Models	A_F	A_S
ResNet-50	92.28%	90.65%
VGG-16	95.74%	92.31%
DenseNet-121	96.28%	94.78%
MobileNet-V2	95.62%	93.53%
DAF-EfficientNetV2	**98.96**%	**97.29**%

In the table, each row represents a model, each column represents a dataset, the number indicates the accuracy of the classification, and the bold numbers indicate the optimal performance of DAF-EfficientNetV2 on that dataset. We can use this table to visually compare the performance of each model on different datasets. Figure 14 shows the loss and accuracy curves of the five models tested on the two publicly available datasets.

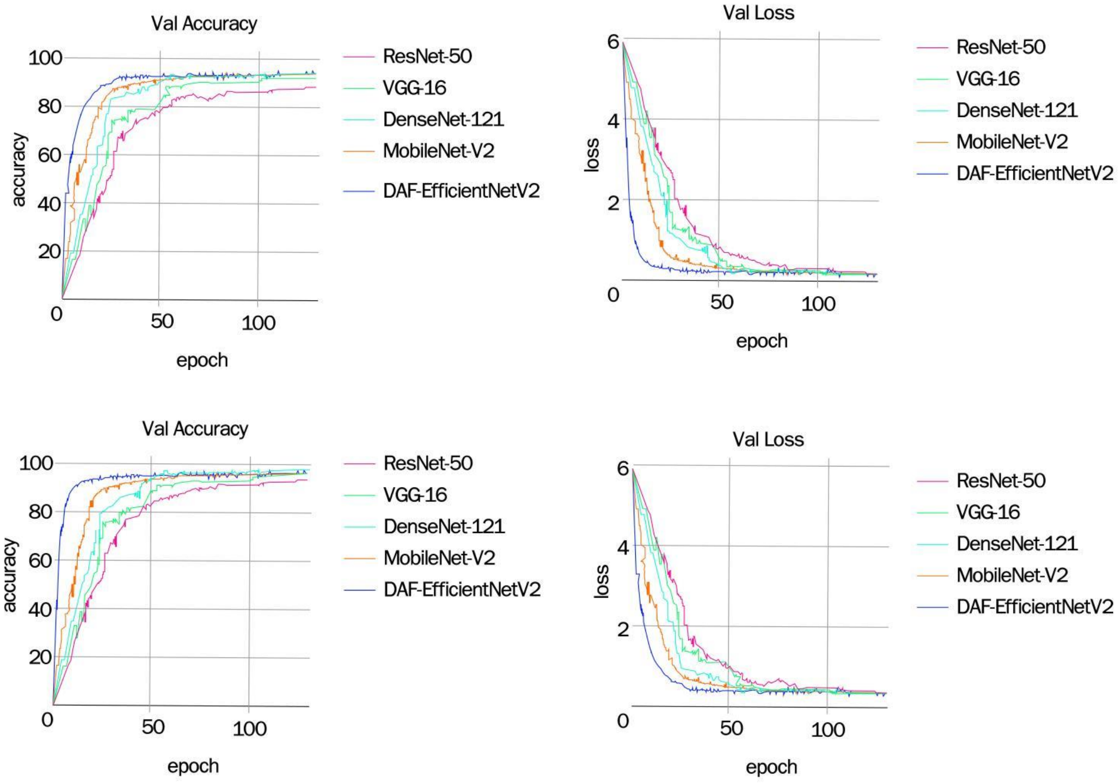


Figure 14. Loss and accuracy curves for each model tested on two publicly available datasets.

5.2. Ablation experiments

In the field of deep learning, data augmentation and hyperparameter optimization are very important research directions, and they can effectively improve the performance and generalization of the model. In order to explore more deeply the effects of these factors and the attention mechanism module in the shallow module on the DAF-EfficientNetV2 model on the model performance, we conducted ablation experiments and also adjusted the hyperparameters such as learning rate and batch size. The experimental results show that the D-MBConv module can effectively improve the performance of the model and achieve better results for all data sets. Also, we found that the optimal hyperparameter settings differed on different datasets, which suggested that we needed to adjust and optimize the hyperparameters in real situations.

In the experiments on learning rate and batch size, we found that increasing the batch size within a certain range can improve the accuracy of the model, but the accuracy of the model decreases instead as the batch size increases further. In addition, when the learning rate is small, increasing the batch size can improve the accuracy of the model, but when the learning rate gradually increases, the effect of increasing the batch size also gradually decreases. Figure 15 shows the loss and accuracy curves of EfficientNetV2 and DAF-EfficientNetV2 tested on two public datasets.

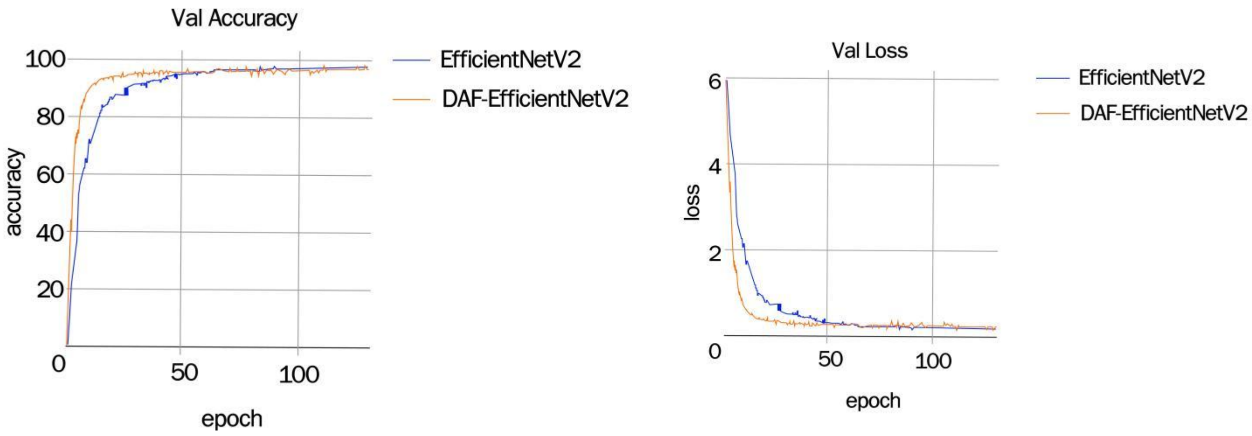


Figure 15. Loss and accuracy curves of EfficientNetV2 and DAF-EfficientNetV2 tested on two publicly available datasets.

5.3. Comparison of other metrics of the model

The proposed algorithm was compared with ResNet-50, VGG-16, DenseNet-121, and MobileNet-V2, and EfficientNetV2 to verify the superiority of the proposed algorithm. The performance on the dataset is shown in Table 3. The improved algorithm is higher than the other five algorithms in terms of accuracy, recall, and F1 value, which fully indicates that the algorithm is more suitable for finger vein recognition.

Table 3. Comparison of recognition effects of different models.

Classification model	Precision	Recall	F1-score
ResNet-50	86.34	81.92	83.65
VGG-16	91.25	87.41	89.54
DenseNet-121	94.32	90.83	92.58
DAF-EfficientNetV2	97.47	96.96	97.12
EfficientNetV2	96.95	95.46	95.82
MobileNet-V2	92.03	87.58	90.61

6. Conclusions

In this paper, a new finger vein recognition method is proposed, and the study achieves efficient recognition of finger veins by applying ACO to the original images of finger veins, using Mini-ROI for boundary determination, and improving the EfficientNetV2 model by combining a dual attention fusion network with it, compared to previous popular deep learning models, the improved algorithm in this paper achieves the highest test accuracy of 98.96% on the Malaysia The improved algorithm in this paper achieves the highest test accuracy of 98.96% on the finger vein dataset of University of Malaysia (FV-USM) compared to the previous popular deep learning models, but in real life, the CNN used has more trainable parameters and longer training time, which is not very suitable for real-time finger vein recognition, in addition, the current finger vein lacks a large number of high-quality datasets, so for different application scenarios, we still need to further optimize the CNN model and choose the most Therefore, for different application scenarios, we need to further optimize the CNN model and select the most suitable method in order to achieve the best recognition effect while ensuring the efficiency and establishing a better quality and more comprehensive finger vein benchmark database is the future research direction.

Conflict of interest

The authors declare there is no conflict of interest.

References

1. F. Radzi, M. Khalil-Hani, R. Bakhteri, Finger-vein biometric identification using convolutional neural network, *Turkish J. Electr. Eng. Comput. Sci*., **24** (2016), 1863–1878. https://doi.org/10.3906/elk-1311-43
2. R. Das; E. Piciucco, E. Maiorana, P. Campisi, Convolutional neural network for Finger-Vein-Based biometric identification, *IEEE Trans. Inform. Forensics Secur.*, **14** (2018), 360–373. https://doi.org/10.1109/TIFS.2018.2850320
3. K. J. Noh, J. Choi, J. S. Hong, K. R. Park, Finger-Vein recognition based on densely connected convolutional network using score-level fusion with shape and texture images, *IEEE Access*, **8** (2020), 96748–96766.
4. D. Zhao, H. Ma, Z. Yang, J. Li, W. Tian, Finger vein recognition based on lightweight CNN combining center loss and dynamic regularization, *Infrared Phys. Technol.*, **105** (2020), 103221. https://doi.org/10.1016/j.infrared.2020.103221
5. Hao, Z.; Fang, P.; Yang, H. Finger vein recognition based on multi-task Learning, in *Proceedings of the 2020 5th International Conference on Mathematics and Artificial Intelligence*, (2020), 133–140. https://doi.org/10.1145/3395260.3395277
6. Y. Lu, S. Xie, S. Wu, Exploring competitive features using deep convolutional neural network for finger vein recognition, *IEEE Access*, **7** (2019), 35113–35123. https://doi.org/10.1109/ACCESS.2019.2902429
7. R. S. Kuzu, E. Maiorana, P. Campisi, Vein-Based biometric verification using transfer learning, in *Proceedings of the 2020 43rd International Conference on Telecommunications and Signal Processing (TSP)*, (2020), 403–409. https://doi.org/10.1109/TSP49548.2020.9163491
8. Z. Xu, M. M. Kamruzzaman, J. Shi, Method of generating face image based on text description of generating adversarial network, *J. Electr. Imaging*, **31** (2022), 051411. https://doi.org/10.1117/1.JEI.31.5.051411
9. Y. Zhang, W. Li, L. Zhang, X. Ning, L. Sun, Y. Lu, Adaptive learning gabor filter for finger-vein recognition, *IEEE Access*, **7** (2019), 159821–159830. https://doi.org/10.1109/ACCESS.2019.2950698
10. B. Hou, R. Yan, Convolutional autoencoder model for finger-vein verification, *IEEE Trans. Instrum. Meas.*, **69** (2020), 2067–2074. https://doi.org/10.1109/TIM.2019.2921135
11. B. Bharadwaj, J. S. Banu, M. Madiajagan, M. R. Ghalib, O. Castillo, A. Shankar, GPU-Accelerated implementation of a genetically optimized image encryption algorithm, *Soft Comput.*, **25** (2021), 14413–14428. https://doi.org/10.1007/s00500-021-06225-y

12. G. Wang, C. Sun, A. Sowmya, Multi-weighted co-occurrence descriptor encoding for vein recognition, *IEEE Trans. Inform. Forensics Secur.*, **15** (2019), 375–390. https://doi.org/10.1109/TIFS.2019.2922331
13. P. Jayapriya, K. Umamaheswari, Finger knuckle biometric feature selection based on the FIS_DE optimization algorithm, *Neural Comput. Appl.*, **34** (2022), 5535–5547. https://doi.org/10.1007/s00521-021-06705-0
14. W. Liu, W. Li, L. Sun, L. Zhang, P. Chen, Finger vein recognition based on deep learning, in *2017 12th IEEE conference on industrial electronics and applications (ICIEA)*, (2017), 205–210. https://doi.org/10.1109/ICIEA.2017.8282842
15. Y. Liu, J. Ling, Z. Liu, J. Shen, C. Gao, Finger vein secure biometric template generation based on deep learning, *Soft Comput.*, **22** (2018), 2257–2265. https://doi.org/10.1007/s00500-017-2487-9
16. W. Yang, S. Wang, J. Hu, G. Zheng, J. Yang; C. Valli, Securing deep learning based edge finger vein biometrics with binary decision diagram, *IEEE Trans. Ind. Inform.*, **15** (2019), 4244–4253. https://doi.org/10.1109/TII.2019.2900665
17. J. Zhang, Z. Lu, M. Li, H. Wu, GAN-based image augmentation for finger-vein biometric recognition, *IEEE Access*, **7** (2019), 183118–183132. https://doi.org/10.1109/ACCESS.2019.2960411
18. Y. Zhang, Z. Liu, Research on finger vein recognition based on sub-convolutional neural network, in *2020 International Conference on Computer Network, Electronic and Automation (ICCNEA)*, (2020), 211–216. https://doi.org/10.1109/ICCNEA50255.2020.00051
19. S. Liu, E. Huang, Y. Xu, K. Wang, D. K. Jain, Computation of facial attractiveness from 3D geometry, *Soft Comput.*, **26** (2022), 10401–10407. https://doi.org/10.1007/s00500-022-07324-0
20. G. K. Sidiropoulos, P. Kiratsa, P. Chatzipetrou, G. A. Papakostas, Feature extraction for finger-vein-based identity recognition, *J. Imaging*, **7** (2021), 89. https://doi.org/10.3390/jimaging7050089
21. S. Daas, A. Yahi, T. Bakir, M. Sedhane, M. Boughazi, E. Bourennane, Multimodal biometric recognition systems using deep learning based on the finger vein and finger knuckle print fusion, *IET Image Proc.*, **14** (2020), 3859–3868. https://doi.org/10.1049/iet-ipr.2020.0491
22. J. Zhang, C. Li, S. Kosov, M. Grzegorzek, K. Shirahama, T. Jiang, et al., LCU-Net: A novel low-cost U-Net for environmental microorganism image segmentation, *Pattern Recognit.*, **115** (2021), 107885. https://doi.org/10.1016/j.patcog.2021.107885
23. J. Zhang, C. Li, Y. Yin, J. Zhang, M. Grzegorzek, Applications of artificial neural networks in microorganism image analysis: a comprehensive review from conventional multilayer perceptron to popular convolutional neural network and potential visual transformer, *Artif. Intel. Rev.*, **56** (2023), 1013–1070. https://doi.org/10.1007/s10462-022-10192-7