Finger Vein Recognition system using CNN

Geeta Shivani, Dr. Ashok Kumar A. R.
Department of Computer Science and Engineering
R V College of Engineering, Bangalore, India

Abstract: As a new biometric technology, finger vein recognition has received much attention in recent years. Since deep learning is an end-to-end system and has achieved very good results in such fields as face recognition and target detection, we try to apply it to finger vein recognition. The other forms of biometrics few demerits like they can be duplicated or forged as they leave behind some trace. But this doesn't happen in case of finger vein because they are present inside the finger. So this is gaining popularity among the biometric authentication. Medical research shows: (1) Finger veins for different people are not the same (2) For the same person, his/her finger veins vary among different fingers (3) For adults, patterns of finger veins will not change over time. Hence, the finger vein offers a safe and feasible approach for biometric recognition. In this project, five layers of CNN are adopted which include 3 convolution layers and 2 fully connected layers. This network obtains a recognition rate of 99.53%, which proves to be better performing than the traditional algorithm.With the increase in the number of layers, the accuracy increases.The fv-usm model is ued for this. Along with the CNN sequential model, the VGG16 model is used. The initial layers of tfhe model are trained.

Keywords:CNN, VGG16.

I. INTRODUCTION

Biometrics are common these days and are very useful in securing private information, personal identification, and verification. There have been a lot of various methods of identification and verification with respect to biometrics including fingerprint recognition, face recognition, palm print, password, access cards, iris recognition, and so on. These are commonly used in the items we use daily like mobile phones. Finger vein recognition is an addition to this list and is gaining popularity for being very reliable.

Password, Access cards, and PIN (Personal Identification Number) are the basic biometrics used for information protection are easy to implement but they have their own demerits. They can be forgotten or can be easily exposed. The fingerprint is not permanent over time and can be cheated or a dummy can be created becaause it leaves behind trace. Palm print can be easily frayed. Voice, signatures can be easily copied or dub bed. Face recognition becomes difficult in the case of its occurrences such as wearing makeup, glares, face-lifts, wearing hats or cap.

What we require now is a cost-efficient, accurate and reliable biometrics system. All single person's veins are having unique physical and behavioral features. It provides a greater degree of security that protects information and access control in a much better manner. The biometric technique which identifies a person by using the vein patterns present inside the fingers is the finger vein system. Every person has unique vein patterns. A comparative evaluation of the major biometric modalities is described in Table 1. Each modality has its advantages and disadvantages. For example, the fingerprint is the least expensive method and is commonly used for identifying suspected criminals. However, the age and occupation of a person may cause some difficulties in capturing a complete and accurate fingerprint image. Fingerprints are also relatively easy to be copied, so that users may be restricted when used for security purposes.

Biometrics	Accuracy	Data Size	Cost	Security Level	Long-term Stability
Finger Vein	High	Medium	High	High	High
Fingerprint	Medium	Small	Low	Low	Low
Face	Low	Large	High	Low	Low
Iris	High	Large	High	Medium	Medium
Voice	Low	Small	Medium	Low	Low
Hand Geometry	Low	Large	High	Low	Low

II. RELATED WORK

The matching is the same as classification in [1]. The matching is done using the distance classifier which is simply the difference between features of different or same images of finger vein and the difference is compared with the threshold value. If the difference is less than or equal to the threshold then the images are likely to be matched and if it is greater than the threshold then it is not the image of the same personality. The features fractal dimensions, lacunae, and Gabor features are used for matching.

The Network in [2] consists of 5 convolution layers and 2 fully connected layers. Input images are grey with a fixed size of 60*175 pixels. Convolution layer C1 is the first layer of the net, in which we first do convolution of the image followed by ReLU, max pooling, and LRN. The second, third, fourth, and fifth layer of the net is called convolution layer.

In [3], same images of the finger vein and the difference is compared with the threshold value. If the difference is less than or equal to the threshold then the images are likely to be matched and if it is greater than the threshold then it is not the image of the same personality.

An algorithm is proposed in [4] that uses band-limited phase-only correlation(BLPOC) to improve finger vein recognition performance. phase-only correlation (POC) is an efficient matching method using the phase component of the two-dimensional discrete Fourier transform (2D DFT) of the image. The image is converted into the frequency domain, and the registered image is compared with the correlation value to perform recognition. In addition to finger vein recognition, it is used for biometric recognition and has the advantage that the amount of computation is small and the time taken for recognition is short.

A scenario in [5] where algorithm and database are fixed, we think every user has a "best" finger, but the "best" finger for each user may be different. For example, in the first case mentioned in section 2, middle fingers perform better than index fingers. This means when all of the users use middle fingers, the performance of the system is better than when they all use index fingers.

CNN is a multilayer perceptron (MLP) network with a special topology containing more than one hidden layer CNNs are primarily used for object recognition in image processing, handwritten character recognition, and speech recognition, as they automatically extract discriminative features inside their layers from raw input information in [6].

They have employed an improved deep network, named Merge Convolutional Neural Network (Merge CNN), in [7] which uses several CNNs with short paths. The scheme is based on the use of multiple identical CNNs with different input image qualities, and the unification of their outputs into a single layer. To achieve this, we designed different networks and trained them with the FV-USM dataset.

The architecture of LCNN is employed [8] experiments (MFM part is excluded to maintain the clarity). The network contains 9 convolutional layers (Conv), 4 pooling layers (pooling) and 2 fully connected layers (Fc), and some assistant layers.

They introduce a unique and lightweight image enhancement method in [9] for person identification using Convolutional Neural Networks (CNN). As preprocessing steps, Contrast Limited Adaptive Histogram Equalization (CLAHE) followed by gamma correction is applied. Afterward, the image is sharpened and then passed through the median filter.

They propose a new finger vein image encryption scheme in [10], which applies Rivest–Shamir– Adleman encryption technology to finger vein image encryption. In addition, a complete cancelable finger vein Recognition system with template protection is proposed to ensure the security of the user's vein template while maintaining the recognition performance.

From the above papers, we observed that, as the number of layers of convolution layer increases, the accuracy of the model also increases. It is observed that we get more accuracy in the pre-trained model since the number of parameters considered is more compared to the model built from the scratch.

III. METHODOLOGY

A. Image pre-processing (Region of interest extraction):

A captured image contains not only the finger but also the background which is the capture machine. The purpose of extracting ROI is to save the finger part and remove the background. The upper and lower boundaries have to be found to capture the ROI.

B: Image pre-processing (Region of interest extraction):

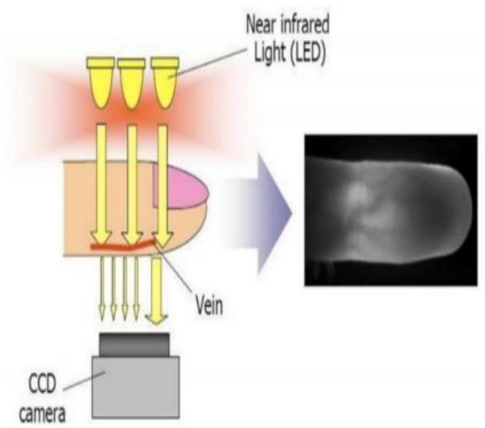
In general, CNN models which provide satisfying results on large datasets have enormous parameters. If we want to train a complicated CNN model from scratch, it is very time-consuming and resource consuming. Also usually, we do not have enough data to train from scratch. We should think about the overfitting issue too. If the model is too complicated and our datasets are rather small, it gets a great chance to overfit. Transfer Learning solves this problem. Transfer Learning aims to adapt an existing model (pre-trained models on large amounts of data) to other domains or other kinds of tasks. For example, we can use a model which was pre-trained on a large cat and dog dataset to classify elephants and monkeys or to classify cartoon cats and dogs. As we can imagine, a direct application of a pre-trained model to other domains or other tasks may not work well because the model does not see information from this domain when it was trained. We usually have two choices. On the one hand, the pre trained CNN model can be treated as a feature extractor. A linear classifier can be built by using extracted features as input. On the other hand, the fine-tuning method is often carried out to fine-tune some high-level layers. Features in the early layers are more generic. While features in the later layers contain more specific information of original datasets. Freezing early layers can bring us general and useful features for many tasks. And fine-tuning the following layers can generate more particular features existing in our datasets.

C: Finger-vein Recognition:

It proposes two cases of finger-vein recognition. The first case uses finger- vein images as inputs. The classifier allows classifying different fingers. The other case aims to use different images to classify authentic matching (matching between input and enrolled finger-vein images of the same class) and imposter matching (matching between input and enrolled finger-vein images of different classes). Both cases use a pre-trained CNN model and fine-tune the model with finger-vein datasets. As we have previously introduced, a biometric authentication system has two possible modes, identification, and verification. The first case for experiments corresponds to the identification mode, and the second one is for the verification mode.

Dataset details

FV-USM database: The images in the database 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: 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. In the first session, a total of 2952 (123 x 4 x 6) images were collected. Therefore, from two sessions, we obtained a total of 5904 images from 492 finger classes. The spatial and depth resolution of the captured finger images were 640 x 480 and 256 grey levels, respectively.



Capturing of a vein

ML/DL Techniques used

A convolutional neural network, or CNN, is a deep learning neural network designed for processing structured arrays of data such as images. Convolutional neural networks are widely used in computer vision and have become the state of the art for many visual applications such as image classification, and have also found success in natural language processing for text classification. There are 4 layers of CNN namely: 1. Convolution layer 2. Activation function 3. Pooling layer 4. Fully connected layer the task of these layers is assigned task of reducing the parameters, in order to reduce overfitting that reduces the accuracy. 1. Convolution layer is the first layer. It provides meaningful, low dimensional, invariant feature. 2. The activation function is important because without having an activation function even if there are about 10 layers they are as good as a single layer. 3. The pooling layer is usually used for down sampling the features. This will reduce the number of parameters making it easier for learning. 4. The final layer is a fully connected layer in which every node is connected to every other node in the next layers. This layer has deep 3D structure and lot of kernels as the last 3 layers can be applied many numbers of times. It accepts 3D feature map as input and gives out 1D feature vector as output. For the conversion purpose the flattening is used.

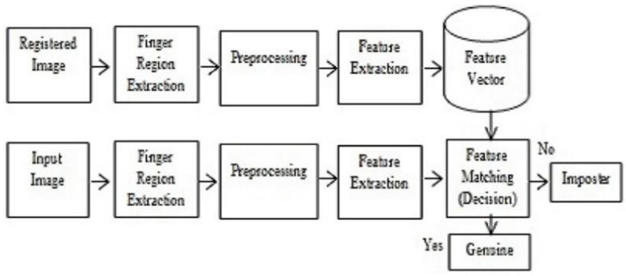
Use Cases

Use case 1: Confirm a payment A lot of financial services now have mobile apps. With many, you can open an app on your phone and quickly authenticate yourself by means of a password, or with your social media login. This is convenient to quickly check your account balance, or check if a refund has been credited etc. However, if you're going to complete a higher-risk transaction - e.g. transfer money or make a big payment - you want to make sure that this is secure. Such a transaction requires stronger authentication than your password or social media login, so you can 'step up' the authentication to require finger vein biometric authentication at this point only. This reduces friction for simple tasks, while better securing high risk tasks.

Use case 2: Identity delegation In this scenario, an administrator user gives power (delegate authority and access/rights) to another user to accomplish a task or take over a role. But before the second user takes this power, they have to prove their identity using strong authentication. This is where finger vein authentication comes into play. A remote employee receives an invitation to take the role of 'purchasing assistant' in an e commerce service. They will first use finger vein biometrics to enroll in the system, then in the future they will complete purchases. (according to the level of rights delegated to their identity) by holding their hand up to the camera and authenticating with finger vein authentication.

Use case 3: Single Sign-On (SSO) Today it is very rare that an organization has just one internet service. The opposite is commonplace: many web services and mobile applications from one organization. Using single sign-on with finger vein biometrics would allow users to seamlessly use SSO to easily switch between a financial planning application, an e-commerce site and another application without a new login. They don't need a different set of credentials for each service. SSO is particularly powerful for financial services as it allows customers to be strongly authenticated and ready to access several services with one authentication.

Workflow Diagram



The flow of finger vein recognition

A finger vein recognition system consist of the following steps: (a) Image Acquisition (b) Finger Region Extraction (c) Preprocessing (d) Feature Extraction (e) Matching

1. ***Image acquisition***

Image accusation is the first most step in the finger vein recognition system. The finger vein image is acquired using the finger vein scanner or the finger vein reader. The finger vein scanner consists of the NIR (Near Infra-red) sensors. The finger vein scanner works on the principle of light transmission phenomenon. The finger vein scanner consists of the acryl which serves as the platform for placing the finger. The CCD preprocessor camera is used for capturing the image of finger veins.

2. ***Finger Region Extraction***

The Finger region extraction i.e. ROI (Region of Interest) from the finger vein image is extracted. The ROI is obtained by thresholding the image. The multidimensional filtering is done for thresholding. The ROI extracted image is obtained by convolving the cropped and thresholder image.

3. ***Preprocessing***

In the image preprocessing the image enhancement is done by certain image resizing rule in order to get the finger vein patterns to be visible properly. First the original image is resizing to 1/4th of its original size and again it is restoring back to original. Again, the restored image is resizing to 1/3rd of its size.

4. ***Feature Extraction***

The feature extraction is most important step in the recognition process. In the finger vein recognition system, the important features are texture and edges. The algorithms used for texture extraction is Fractal Dimension and Lacunae and for extracting edges the Gabor filter is used

5. Matching: The matching is same as classification. The matching is done using the distance classifiier which is simply the difference between features of different or same images of finger vein and the difference is compared with the threshold value. If the difference is less than or equal to threshold then the images are likely to be matched and if it is greater than threshold then it is not the image of the same personality. The features fractal dimensions, lacunae and gabor features are used for matching.

Preprocessing

Image pre-processing (Region of interest extraction): A captured image contains not only the finger but also the background which is the capture machine. The purpose of extracting ROI is to save the finger part and remove the background. The upper and lower boundaries have to be found to capture the ROI. We have established our ROI system with the following steps:

- Cropping images to 240x240 pixels
- Separating the image into two halves and applying filters to detect upper and lower boundaries
- Keeping only the region between the upper and lower limits
- Doing linear stretching.

Validation Methodology

The lower the loss, the better a model (unless the model has over-fitted to the training data). The loss is calculated on training and validation and its interpretation is how well the model is doing for these two sets. Unlike accuracy, loss is not a percentage. It is a summation of the errors made for each example in training or validation sets. In the case of neural networks, the loss is usually negative log-likelihood and residual sum of squares for classification and regression respectively. Then naturally, the main objective in a learning model is to reduce (minimize) the loss function's value with respect to the model's parameters by changing the weight vector values through different optimization methods, such as backpropagation in neural networks. Loss value implies how well or poorly a certain model behaves after each iteration of optimization. Ideally, one would expect the reduction of loss after each, or several, iteration(s). The accuracy of a model is usually determined after the model parameters are learned and fixed and no learning is taking place. Then the test samples are fed to the model and the number of mistakes (zero-one loss) the model makes are recorded, after comparison to the true targets. Then the percentage of misclassification is calculated. The below figure shows the accuracy rate of the model for 12 epochs which are automatically calculated based on the dataset size

```
Instructions for updating:
Use the `axis` argument instead

Training Epoch 1/12 --- Training Accuracy:   0.0%, Validation Accuracy:   0.0%,  Validation Loss: 6.258
Training Epoch 2/12 --- Training Accuracy:   0.0%, Validation Accuracy:   0.0%,  Validation Loss: 6.177
Training Epoch 3/12 --- Training Accuracy:   0.0%, Validation Accuracy:   0.0%,  Validation Loss: 6.254
Training Epoch 4/12 --- Training Accuracy:   0.0%, Validation Accuracy:   0.0%,  Validation Loss: 6.218
Training Epoch 5/12 --- Training Accuracy:  15.0%, Validation Accuracy:   0.0%,  Validation Loss: 5.882
Training Epoch 6/12 --- Training Accuracy:  60.0%, Validation Accuracy:  40.0%,  Validation Loss: 4.247
Training Epoch 7/12 --- Training Accuracy:  90.0%, Validation Accuracy:  45.0%,  Validation Loss: 2.239
Training Epoch 8/12 --- Training Accuracy: 100.0%, Validation Accuracy:  60.0%,  Validation Loss: 1.231
Training Epoch 9/12 --- Training Accuracy: 100.0%, Validation Accuracy:  50.0%,  Validation Loss: 1.819
Training Epoch 10/12 --- Training Accuracy: 100.0%, Validation Accuracy:  75.0%,  Validation Loss: 2.252
Training Epoch 11/12 --- Training Accuracy: 100.0%, Validation Accuracy:  85.0%,  Validation Loss: 0.551
Training Epoch 12/12 --- Training Accuracy: 100.0%, Validation Accuracy:  90.0%,  Validation Loss: 0.777

Execution Time: 4.462147446473439 minutes
```

Fig. Epoch and accuracy

```
9654
Epoch 95/100
93/93 [==============================] - 516s 6s/step - loss: 0.0631 - accuracy: 0.9835 - val_loss: 0.0977 - val_accuracy: 0.
9705
Epoch 96/100
93/93 [==============================] - 540s 6s/step - loss: 0.0751 - accuracy: 0.9776 - val_loss: 0.0749 - val_accuracy: 0.
9746
Epoch 97/100
93/93 [==============================] - 539s 6s/step - loss: 0.0619 - accuracy: 0.9864 - val_loss: 0.0866 - val_accuracy: 0.
9766
Epoch 98/100
93/93 [==============================] - 538s 6s/step - loss: 0.0630 - accuracy: 0.9825 - val_loss: 0.0393 - val_accuracy: 0.
9868
Epoch 99/100
93/93 [==============================] - 526s 6s/step - loss: 0.0543 - accuracy: 0.9844 - val_loss: 0.0377 - val_accuracy: 0.
9909
Epoch 100/100
93/93 [==============================] - 527s 6s/step - loss: 0.0968 - accuracy: 0.9671 - val_loss: 0.1062 - val_accuracy: 0.
9634
```

Fig.Accuracy rate of the VGG16 model which is trained for 100 epochs.

```
model.save('vgg_model.h5')
model.save_weights('vgg_model_weights.h5')
```

```
Epoch 100/100
93/93 [==============================] - 527s 6s/step - loss: 0.0968 - accuracy: 0.9671 - val_loss: 0.1062 - val_accuracy: 0.
9634
```

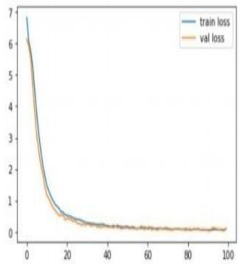


Fig. The graph of training and validation loss

IV. RESULTS

The following is obtained as a result of the finger vein recognition process:

Fig. Folder of the pre-processed image

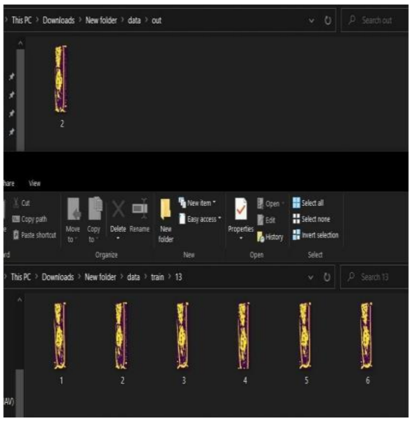
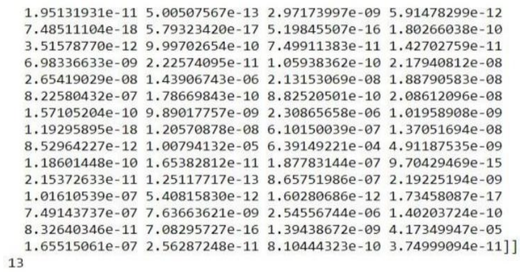


Fig. Extraction of region of interest for one finger vein



```
 1.95131931e-11 5.00507567e-13 2.97173997e-09 5.91478299e-12
 7.48511104e-18 5.79323420e-17 5.19845507e-16 1.80266038e-10
 3.51578770e-12 9.99702654e-10 7.49911383e-11 1.42702759e-11
 6.98336633e-09 2.22574095e-11 1.05938362e-10 2.17940812e-08
 2.65419029e-08 1.43906743e-06 2.13153069e-08 1.88790583e-08
 8.22580432e-07 1.78669843e-10 8.82520501e-10 2.08612096e-08
 1.57105204e-10 9.89017757e-09 2.30865658e-06 1.01958908e-09
 1.19295895e-18 1.20570878e-08 6.10150039e-07 1.37051694e-08
 8.52964227e-12 1.00794132e-05 6.39149221e-04 4.91187535e-09
 1.18601448e-10 1.65382812e-11 1.87783144e-07 9.70429469e-15
 2.15372633e-11 1.25117717e-13 8.65751986e-07 2.19225194e-09
 1.01610539e-07 5.40815830e-12 1.60280686e-12 1.73458087e-17
 7.49143737e-07 7.63663621e-09 2.54556744e-06 1.40203724e-10
 8.32640346e-11 7.08295727e-16 1.39438672e-09 4.17349947e-05
 1.65515061e-07 2.56287248e-11 8.10444323e-10 3.74999094e-11]]
13
```

V. CONCLUSION AND FUTURE WORK

There has been many survey papers written on the project, Finger vein recognition though it is gaining popularity in in recent years. Each model proposed in the papers had a little different methodology followed. The models used in this project are usual CNN sequential model and VGG16 model.The dataset used is the FV-USM database. The validation loss of these are less compared to the model proposed in the base paper, which uses the dataset SDUMLA-FV database.

It shows that the performance of using CNN is better than using traditional algorithms. Besides, CNN is an end-to-end system that is simple for finger-vein recognition and has high robustness. Considering the superior performance of CNN, we will do more research with a deep learning method on finger vein recognition in the future. One important thing about finger vein recognition is to build a public database with a large quantity of data. In the future, studies on a lighter model for solving the issue of long processing time, caused using two input images, as well as that on the measures to shorten the processing time of the pre-processing algorithm, will be conducted. Furthermore, the possibility of applying the proposed algorithm to other types of biometric data (face, iris, and fingerprint) besides finger-vein will be investigated.

As we know that deep learning method is data-driven. Data quality is a key element to achieve successful experiment results. The dataset we used is not of good quality, but the results are still promising. Both final results of the identification model and the verification model beat the results in the paper to which is referred. Unlike other models, deep learning methods are fast to implement and more straightforward to build up without diving into too much complicated feature handling and engineering. However, compared to some results in other papers (not always using deep learning methods), we still have improvements to pursue in terms of the data pre-processing, architectures of models, and choices of hyper-parameters.

REFERENCES

[1] Yilong Yin, Lili Liu, and Xiwei Sun. SDUMLA-HMT: A Multimodal Biometric Database. The 6th Chinese Conference on Biometric Recognition (CCBR 2011), LNCS 7098, pp. 260– 268, Beijing, China, 2011.

[2] Yang G, Xi X, Yin Y. Finger vein recognition based on a personalized best bit map. Sensors (Basel). 2012.

[3] Shaheed, Kashif & Id, Hangang &, Liu & Yang, Gongping & Qureshi, Imran & Gou, Jie & Yin, Yilong. information A Systematic Review of Finger Vein Recognition Techniques. Information (Switzerland). 2018.

[4] Meng, G., Fang, P., & Zhang, B. Finger vein recognition based on convolutional neural network. In MATEC Web of Conferences EDP Sciences. 2017

[5]SDUMLA-HMT Database http://mla.sdu.edu.cn/info/1006/1195.htm

[6]https://www.analyticsvidhya.com/blog/2017/06/transfer-learning-the-art-of-fine-tuning-a pre-trained- model/

[7] Pham TD, Park YH, Nguyen DT, Kwon SY, Park KR. Nonintrusive Finger-Vein Recognition System Using NIR Image Sensor and Accuracy Analyses According to Various Factors. Sensors (Basel). 2015.

[8] Ferguson, Max & AK, Ronay & Tina Lee, Yung-Tsun & H. Law, Kincho. Automatic localization of casting defects with convolutional neural networks. 2017.

[9] Simonyan, Karen & Zisserman, Andrew. Very Deep Convolutional Networks for Large Scale Image Recognition. 2014.

[10] Hong HG, Lee MB, Park KR. Convolutional Neural Network-Based Finger-Vein Recognition Using NIR Image Sensors. Sensors (Basel). 2017.