Contrastive Learning-Based Finger-Vein Recognition with Automatic Adversarial Augmentation

Shaojiang Deng[1], Huaxiu Luo[1], Huafeng Qin[2], and Yantao Li[1](✉)

[1] College of Computer Science, Chongqing University, Chongqing, China
yantaoli@cqu.edu.cn
[2] Chongqing Technology and Business University, Chongqing, China

Abstract. In finger-vein recognition tasks, obtaining large labeled datasets for supervised deep learning is often difficult. To address this challenge, self-supervised learning (SSL) provides a solution by first pre-training a neural network using unlabeled data and subsequently fine-tuning it for downstream tasks. Contrastive learning, a variant of SSL, enables effective learning of image-level representations. To address the issue of insufficient labeled data for vein feature extraction and classification, we propose CL3A-FV, a Contrastive Learning-based Finger-Vein image recognition approach with Automatic Adversarial Augmentation in this paper. Specifically, CL3A-FV consists of the dual-branch augmentation network, Siamese encoder, discriminator, and distributor. The training process involves two steps: 1) training the Siamese encoder by updating its parameters while keeping other components fixed; and 2) training the dual-branch augmentation network with a fixed Siamese encoder, integrating a discriminator to distinguish views generated by the two branches, and a distributor to constrain the distribution of the augmented data. Both networks are updated adversarially using the stochastic gradient descent. We conduct extensive experiments to evaluate CL3A-FV on three finger-vein datasets, and the experimental results show that the proposed CL3A-FV achieves significant improvements compared to traditional self-supervised learning techniques and supervised methods.

Keywords: Contrastive learning · Automatic adversarial augmentation · Finger-vein recognition

1 Introduction

Due to its high security and reliability, finger-vein recognition has garnered considerable attention as a promising biometric identification technology. Finger-vein patterns are unique and highly distinctive, which can be captured by non-invasive imaging techniques, making finger-vein recognition a convenient and user-friendly biometric identification method [1].

In recent years, there has been a surge in the development of finger-vein recognition techniques. These approaches encompass both traditional methods that rely on handcrafted features [2,3], as well as deep learning-based methods [4,5]. Traditional

H. Gao et al. (Eds.): CollaborateCom 2023, LNICST 562, pp. 495–513, 2024.
https://doi.org/10.1007/978-3-031-54528-3_27

methodologies, such as repeated line tracking [6], maximum curvature points [7], and local binary pattern [8], have been proposed and demonstrated impressive recognition performance for finger-vein image recognition. However, compared to the rapid evolution of deep learning methods, these traditional handcrafted methods are constrained by their reliance on the prior knowledge. Consequently, an increasing number of deep learning techniques are employed in vein recognition. Recent works have explored various approaches for finger-vein recognition, such as those based on deep separable convolution [9], frequency-spatial coupling network [10], and the utilization of locality-constrained consistent to fuse multiple features [11]. Within deep learning-based methodologies, contrastive learning has risen as a potent technique for acquiring meaningful feature representations in the absence of negative samples [12–14]. Contrastive learning-based approaches have demonstrated exceptional performance in diverse computer vision areas, i.e., object detection, segmentation, and image classification, achieving state-of-the-art results. The typical approach in contrastive learning is to apply two different transformations to the original image, generate two views as inputs, and then learn the feature representation by minimizing the distance between them in a latent space. The success of contrastive learning heavily relies on predefined transformations [15]. However, for finger-vein data, deformation changes may cause the loss of some fine-grained details in vein images after transformations, potentially impacting the performance of contrastive learning methods.

In order to tackle the aforementioned issues, we present a novel Contrastive Learning-based Finger-Vein image recognition with Automatic Adversarial Augmentation, namely CL3A-FV. Instead of the predefined data augmentation operations, our method utilizes a neural network to learn augmented views by training the network in an adversarial manner between the contrastive learning network and the augmentation network. In the proposed CL3A-FV, we use a neural network-based data augmentation method that learns to generate more effective and diverse augmented samples for finger-vein images. Specifically, we train two different generator networks to create two augmented images from an input image, and a discriminator network to distinguish views by two different generator networks. The primary goal of optimizing the generator network is to generate samples that can effectively evade detection by the discriminator network. Conversely, the main objective of optimizing the discriminator network is to precisely classify images originating from two separate enhancement branches. The output of the generator network is used as the augmented input for the contrastive learning network, which learns to produce effective feature representations for finger-vein recognition. We assess the effectiveness of CL3A-FV on three publicly-available finger-vein recognition datasets, and the experimental results demonstrate that CL3A-FV achieves the state-of-the-art performance, and outperforms existing contrastive learning methods that rely on predefined data augmentation operations, thereby improving the accuracy and robustness of finger-vein recognition. The proposed method has several advantages over existing contrastive learning methods. First, it has the capability to acquire enhanced data augmentation techniques tailored specifically for finger-vein images, thereby enhancing the robustness and generalization capabilities of finger-vein recognition models. Second, it can reduce the manual effort required to define data augmentation operations, which is particularly important for large-scale

datasets. Finally, it can seamlessly integrate into current contrastive learning frameworks with minimal adjustments or modifications.

This work can be summarized by the following contributions:

- We propose CL3A-FV, a contrastive learning-based finger-vein recognition with automatic adversarial augmentation to address the issue of insufficient labeled data for vein feature extraction and classification. CL3A-FV consists of four components, namely the dual-branch augmentation network, Siamese encoder, discriminator, and distributor.
- CL3A-FV trains the encoder and augmentation network alternately by stochastic gradient descent in an adversarial manner, thereby improving the recognition accuracy and robustness of the model.
- We conduct extensive experiments on three public finger-vein datasets, by comparing CL3A-FV with traditional supervised learning methods as well as contrastive learning methods. The experimental findings substantiate that CL3A-FV attains the highest level of performance in tasks related to finger-vein image recognition.

The rest of this work is organized as follows. We review the related work in Sect. 2. In Sect. 3, we detail the proposed CL3A-FV in terms of the contrastive learning, dual-branch augmentation network, augmentation discriminator, distribution divergence minimization, and adversarial training, and access the performance of CL3A-FV in Sect. 4. We conclude this work in Sect. 5.

2 Related Work

2.1 Deep Learning Based Vein Recognition

In recent years, deep learning has demonstrated remarkable expertise in extracting meaningful features, resulting in its successful deployment across diverse computer vision tasks. Image classification stands out as a notable instance of its application [16]. Upon this success, researchers have employed deep learning techniques to analyze finger-vein images for various tasks. These tasks encompass the image classification [17–22], feature extraction [23–29], image enhancement [30–35], and image segmentation [36,37]. For *vein image classification*, various approaches have been explored. In [17], ResNet is directly applied to the images for classification, and [19] combines the deep belief network with histograms of uniform local binary patterns derived from curvature gray images for finger-vein recognition. Notably, a joint attention module is introduced in [22] to enhance the vein pattern contribution in the feature extraction. In addition to Convolutional Neural Networks (CNNs), alternative neural network architectures have been employed. For instance, [20] utilizes the Graph Neural Network (GNN) for classification, taking advantage of the graph-like structure inherent in intricate vein textures. Moreover, [18] employs the supervised discrete hashing to improve the matching speed, while [21] incorporates bias field correction and spatial attention to enhance the optimization of CNN-based finger-vein image recognition. For *vein feature extraction*, researchers have employed different strategies. Some studies [23,25,27] follow a similar workflow, where a CNN structure is employed to

extract features from finger-vein images, and then are analyzed by traditional machine learning algorithms. On the other hand, novel approaches have also been explored. For instance, [26] utilizes a combination of autoencoder and convolutional network to learn feature codes from finger vein images. In [28], a capsule neural network-based method is proposed to extract the region of interest (ROI) in finger veins. [24] introduces a lightweight two-channel network with three convolution layers to extract image features efficiently, followed by the verification using a support vector machine. Lastly, [29] presents a multi-modal biometric authentication system based on the deep fusion of electrocardiogram and finger-vein image data. For *vein image enhancement*, Generative Adversarial Networks (GANs) are commonly employed to recover missing vein patterns caused by various factors during the image capture process [32]. For severely-damaged finger-vein images, [33] introduces a modified GAN that utilizes neighbors-based binary pattern texture loss. To address the issue of motion blur in the finger-vein image recognition, [34] proposes a modified DeblurGAN to restore motion-blurred finger-vein images, thereby enhancing the identification performance. Apart from GANs, other modules are also utilized for image restoration in finger-vein image recognition. For instance, [30] presents a method that utilizes a deep CNN with a deconvolution sub-net to recover the original image and a modified linear unit to enhance finger vein texture details. Another study [31] employs a convolutional autoencoder (CAE) to restore venous networks in finger vein images by extracting effective features. In terms of enhancing finger vein image quality, [35] introduces a novel network architecture based on the pulse-coupled neural network. This architecture aimed to improve image quality and make finger vein image recognition more practical. For *vein image segmentation*, [36] proposes a finger vein segmentation algorithm based on LadderNet. This algorithm leverages the concatenation of feature channels from the expanding and contracting paths in the network to obtain comprehensive semantic information from vein images. Furthermore, traditional finger vein segmentation networks often have excessive parameters, making them challenging to use in mobile terminals. To address this issue, [37] proposes a lightweight real-time segmentation network specifically designed for finger vein image recognition on embedded terminals. This network achieves comparable performance to more complex networks while meeting the requirements of embedded mobile terminals.

The application of deep learning methods has further advanced the development of vein recognition, but there are still some drawbacks. For example, from a modeling perspective, due to the complexity of deep learning models, they often perform well on the training set, but their performance on the test set may decrease, which constrains the capacity of model generalization. This may be due to the difference between the training set and test set, such as different acquisition devices or environmental conditions. From a data perspective, issues such as poor quality of finger-vein images (e.g., blur, uneven lighting, spatial position changes, etc.) and sparse single-class images can also affect the robustness of feature extraction.

2.2 Contrastive Learning

Contrastive learning is a specific form of self-supervised learning, which strives to acquire meaningful representations by comparing samples that are similar with ones that are dissimilar [38–40]. The criterion for defining positive and negative samples is an important part of the contrast paradigm, which directly determines whether a good representation can be learned. Traditionally, contrastive learning uses a set of positive and negative pairs to train the model. Positive pairs are samples that are similar, while negative pairs are samples that are dissimilar. The model learns to push the negative pairs apart in the embedding space while pull the positive pairs together. This approach has been used in many successful contrastive learning methods, such as InfoNCE [41] and SimCLR [39]. More recently, some researchers have proposed to use only positive samples in contrastive learning, without negative samples. These methods rely on the idea that samples from the same class are inherently similar, therefore can be used as positive examples. Examples of such methods include SimSiam [13] and BYOL [12], which have achieved the state-of-the-art performance on several benchmark datasets. Overall, the choice of positive and negative samples is an important aspect of contrastive learning, and both approaches have shown promising results in various applications.

2.3 Automatic Augmentation

Automatic augmentation has recently gained popularity in the field of supervised learning [42–46]. [42] uses reinforcement learning to learn data augmentation policies and searches for the optimal augmentation strategy. [43] proposes a search algorithm based on weak data augmentation policies, which significantly reduces the search time. In [44], a search algorithm based on backpropagation finds a balance between computation time and performance. [45] proposes a straightforward yet effective data augmentation strategy that leverages random combinations of operations to enhance data. [46] proposes a population-based optimization algorithm that efficiently learns data augmentation policies. Typically, these methods establish a search space in advance and then search for augmentation policies within that space for the target tasks. However, the search cost of these methods is typically high and not suitable for contrastive learning. To address this limitation, we propose a new self-supervised learning (SSL) strategy in this paper.

3 Proposed Method

This paper introduces CL3A-FV, a contrastive learning-based approach for finger-vein recognition that employs automatic adversarial augmentation to enhance the performance of finger-vein recognition in scenarios with limited labeled data. Figure 1 illustrates the architecture of the proposed CL3A-FV, which consists of four components, namely the Dual-Branch Augmentation Network, Siamese Encoder, Discriminator, and Distributor. The training process is divided into two steps: 1) train the Siamese Encoder, where the other parts of CL3A-FV are fixed and only the parameters of the encoder are updated. The Siamese Encoder learns feature representations by minimizing the similarity between different views of the same image, generated by two learnable neural

networks; and 2) train the Dual-Branch Augmentation Network, where the Siamese Encoder is fixed and a Discriminator is introduced to distinguish the views generated by the two branches. To constrain the distribution of the generated data, a Distributor is introduced to produce discriminative augmentations, i.e., more dissimilar sample pairs. Both networks are updated by the stochastic gradient descent in an adversarial manner. The proposed CL3A-FV can be incorporated into any contrastive learning method, which will further improve the performance of the corresponding contrastive learning method. In the following, we first introduce some prior knowledge about contrastive learning, and then detail each component of CL3A-FV.

3.1 Contrastive Learning

Contrastive learning is a commonly-used image self-supervised learning method, which learns generic features of images by minimizing the distance between different augmented versions of the same image. Specifically, for a given encoder network $f(\cdot)$ and an exponential moving average (EMA) encoder $f'(\cdot)$, the similarity loss used in SSL does not exploit negative samples, which can be written as Eq. (1):

$$\mathcal{L}_C = \frac{1}{2}\mathcal{D}(f(x_1), f'(x_2)) + \frac{1}{2}\mathcal{D}(f(x_2), f'(x_1)), \tag{1}$$

In the given context, $\mathcal{D}$ represents a similarity distance, while x_1 and x_2 refer to two transformed versions of the image, specifically $x_1 = t_1(x)$ and $x_2 = t_2(x)$. These transformations, as previously demonstrated in [39,40,47,48], involve techniques such as cropping the image followed by the application of color transformations.

3.2 Dual-Branch Augmentation Network

The augmentation types can be categorized into illumination augmentation, geometric augmentation, and noise augmentation, which address the uneven illumination or occlusion, incomplete images or positional displacement of the vein curve, and noise introduced by device-related factors during image acquisition, respectively. Inspired by these categories, we construct our dual-branch augmentation network. As illustrated in Fig. 2, one vein image is first normalized and resized to obtain x, which is then fed into two augmentation networks, namely T_{θ_1} and T_{θ_2}. Each augmentation network applies three enhancement blocks to x, resulting in two transformed samples: $x_1 = T_{\theta_1}(x)$ and $x_2 = T_{\theta_2}(x)$. Finally, the contrastive loss between x_1 and x_2 is calculated by Eq. (2):

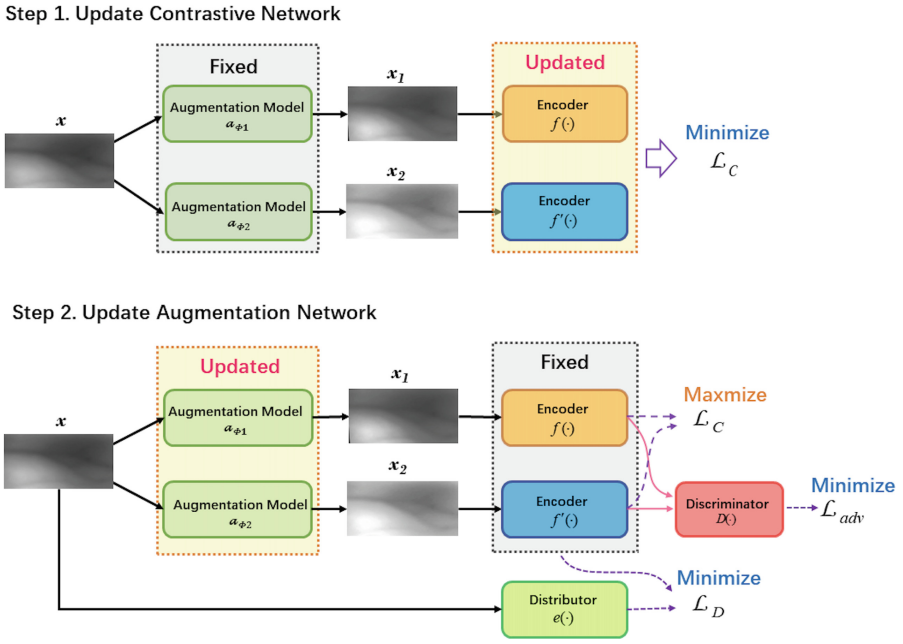

Fig. 1. Architecture of CL3A-FV.

$$\mathcal{L}_C = \frac{1}{2}\mathcal{D}(f(T_{\theta_1}(x)), f'(T_{\theta_2}(x))) + \frac{1}{2}\mathcal{D}(f(T_{\theta_2}(x)), f'(T_{\theta_1}(x))). \tag{2}$$

In particular, the augmentation network T_θ is composed of three modules: a geometric transformation module g_θ, an illumination transformation module c_θ, and a noise transformation module p_θ, which are designed to learn transformations suitable for vein images. The three modules are elaborated as follows:

Geometric Augmentation: Spatial transformation functions commonly employed at present, including rotation, scaling, and translation, can be generalized into more comprehensive functions known as affine transformations. Inspired by STN [49], we design a geometric network g_θ to learn the affine transformation parameters. The network structure can be a multilayer perceptron (MLP) or a convolutional network with the input of an image itself or Gaussian noise. Since geometric transformations are universal for different images, we use MLP due to the generality and transferability as expressed in Eq. (3):

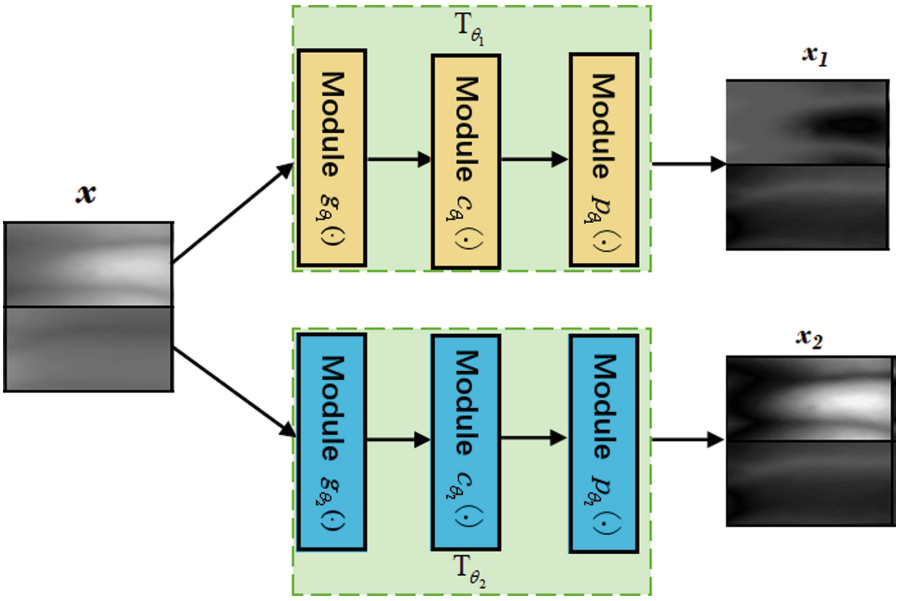

Fig. 2. Dual-branch augmentation network.

$$x' = Augg(x, A + I), A = g_\theta(z), \tag{3}$$

where $Augg(x, A + I)$ is an affine transformation that includes the most geometric transformations applicable to vein images and $A \in R^{2\times3}$ is the learned transformation parameters. $I \in R^{2\times3}$ is represented by a matrix, where $I_{ii} = 1$ or 0, making the affine transformation an identity mapping, and z is an N-dimensional Gaussian noise.

Illumination augmentation: For a given vein image x, the illumination transformation module is defined as Eq. (4):

$$x'' = Augc(x', c), c = c_\theta(x, z), \tag{4}$$

where $Augc$ is a color transformation function with parameters c, and c_θ is a small convolutional network. $z \sim N(0, I_N)$, where $N(0, I_N)$ is an N-dimensional Gaussian distribution.

Noise Augmentation: Additive noise is a commonly-used tool for generating adversarial examples [50,51], and thus we explore it in the image data augmentation. While adversarial examples can be utilized as augmented data during adversarial training, their primary objective revolves around enhancing the model's robustness rather than solely focusing on improvement. Some evidence [52] suggests that adversarial training often sacrifices the model's generalization to clean data. However, our adversarial augmentation model can improve its generalization performance through the online training. Recently, AdvProp [50] has successfully used PGD attack [51] and OnlineAugment [53] has used a learning perturbation for data augmentation. In this work,

we design an adversarial augmentation model to generate additive noise. Specifically, the adversarial augmentation network p_θ takes the original image x as its input and generates additive noise x_z. By adding the generated noise to the original image, we obtain an adversarial perturbed image x'''. We use a variational autoencoder (VAE) network to learn to generate noise, as shown in Eq. (5):

$$x''' = Augp(x'', x_z), x_z = p_\theta(x), \tag{5}$$

where the shape of x_z is the same with the input x.

The above three types of augmentations basically cover all possible variations of the samples. In order to further improve the performance of this method, we also adopt a strong augmentation strategy in this paper, which applies multiple augmentations to the original image. The learning objective of the two augmentation networks is to obtain pairs of samples with large difference. Therefore, we update the augmentation networks by maximizing the distance between the generated sample pairs, which is opposite to the learning objective of the encoder. At the same time, inspired by the approach of generating images with GANs, we introduce a discriminator to distinguish images generated by two different branches, further achieving the goal of generating dissimilar pairs of samples.

3.3 Augmentation Discriminator

In the existing GAN models, a discriminator is introduced to supervise the generator by performing binary classification on generated images and real images, in order to train a generator that can create images close to the real distribution. In the proposed CL3A-FV, we adopt a similar approach to GANs to ensure that the augmentation network produces two "dissimilar" augmented images. Specifically, we introduce a discriminator to perform binary classification on the augmented images from the two branches. The output loss of the discriminator is used as the adversarial loss to update the augmentation network.

We assign different pseudo-labels to the augmented images from the two branches. The input space is denoted as $X = \{X^{aug1}, X^{aug2}\}$, and the set of possible labels is denoted as $Y = 0, 1$, where 0 and 1 are the labels assigned to the augmented images from the two branches. Therefore, the adversarial loss for the two augmentation networks is defined as Eq. (6):

$$\mathcal{L}_{adv}(D(f(x_i)), d_i) = -[d_i log\frac{1}{D(f(x_i))} + (1 - d_i)log\frac{1}{1 - D(f(x_i))}], \tag{6}$$

where $D(\cdot)$ is the introduced discriminator, and d_i denotes the binary variable (augmentation label) for the ith example, which indicates whether x_i comes from augmentation $T_{\theta_1}(x_i \sim X^{aug1},\ if\ d_i = 0)$ or from the augmentation $T_{\theta_2}(x_i \sim X^{aug2},\ if\ d_i = 1)$.

3.4 Distributional Divergence Minimization

To align the images before and after augmentation and prevent harmful augmentation that may affect downstream tasks, we introduce the maximum mean discrepancy

(MMD) to constrain the distribution. The MMD is utilized to quantify the discrepancy between two distributions within the reproducing kernel Hilbert space. The distance between two distributions is calculated as Eq. (7):

$$L_{MMD}(x^{original}, x^{aug}) = \|\frac{1}{n}\sum_{i=1}^{n} g(x_i^{original}) - \frac{1}{m}\sum_{j=1}^{m} g(x_j^{aug})\|_H^2, \tag{7}$$

where $x^{original}$ represents the image before enhancement, and x^{aug} represents the image after augmentation. m and n are the batch sizes, and g is the Gaussian kernel function which maps x to the reproducing Hilbert space. It is worth noting that for the three augmentation networks, we only need this regularization term to ensure that the augmentation networks perform image transform without destroying the original data distribution. Specifically, we input both the original image x and augmented images x_1 and x_2 into a distributor $e(\cdot)$, which is an encoder version of $f(\cdot)$ without the MLP head. During the training, its weights are not directly updated, but inherited from the encoder $f(\cdot)$. We calculate the distribution difference $\mathcal{L}_D$ between the original distribution and the transformed distributions, and use it as the distribution supervision signal for the augmentation network to avoid excessive transformation. The term used to constrain the augmentation networks can be represented as Eq. (8):

$$\mathcal{L}_D = \frac{1}{2}(L_{MMD}(e(x), e(x_1)) + L_{MMD}(e(x), e(x_2))). \tag{8}$$

3.5 Adversarial Training

For the augmentation network, its task is to generate dissimilar pairs of samples, which can force the contrastive network to learn more robust representations. Therefore, when guiding the learning of the augmentation network, we consider three aspects: maximizing the contrastive loss, minimizing the discriminator loss, and minimizing the distribution loss, which can be formulated as Eq. (9):

$$\mathcal{L}_{aug} = \lambda\mathcal{L}_D + \mathcal{L}_{adv} - \mathcal{L}_C, \tag{9}$$

where λ is a hyper-parameter used to constrain the augmentation network. To generate "dissimilar" pairs of samples, we train the model in an adversarial manner. The training process is partitioned into two components: training the augmentation network and training the encoder. These two parts are updated in an alternating manner. The encoder's parameter update is achieved by minimizing $\mathcal{L}_C$, which minimizes the distance between two transformed views. The augmentation network's parameter update is achieved by minimizing $\mathcal{L}_{aug}$, which encourages the two branches to generate more "dissimilar" views and better helps the learning of the encoder.

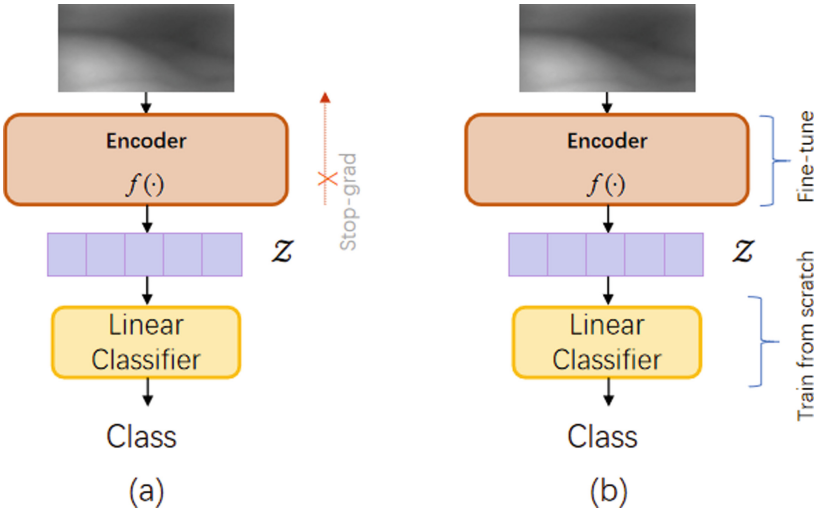

Fig. 3. Linear evaluation protocol (a) and fine-tuning evaluation (b).

4 Experiments

4.1 Experiment Setting and Datasets

Experiment Settings. We conduct extensive linear evaluation experiments on CL3A-FV using the commonly-adopted evaluation methods for self-supervised methods. In addition, to demonstrate the superiority of CL3A-FV over supervised methods, we conduct fine-tuning experiments. In the linear evaluation experiments, we first pre-train the model on an unlabeled dataset, and then further evaluate the trained encoder on a labeled dataset with the fixed encoder weights. In the fine-tuning experiments, we further fine-tune the trained encoder on a labeled dataset with a very small learning rate. We evaluate the proposed CL3A-FV with three different SSL objectives: BYOL [12], SimSiam [13], and Barlow Twins (BT) [14]. In our experiments, the state-of-the-art finger-vein classifiers, i.e. FVRAS-Net [54], FV-CNN [55], PV-CNN [56], and LW-CNN [57] are used to test our approach. We split each dataset into a training set and a testing set with a ratio of $6:4$ for evaluation.

Datasets. To assess the effectiveness of the proposed CL3A-FV, we conduct extensive experiments on three finger-vein databases:

(1) HKPU-FV: The Hong Kong Polytechnic University finger-vein database [58] comprises a total of 3,132 contactless and opening images from 156 subjects. These images include 2,520 finger images taken from 105 subjects across two separate sessions, with an average interval of 66.8 d between sessions. Each subject provided 12 images per session, with six images captured from their index and middle fingers, respectively. The remaining 612 images were captured from 51 subjects in the first session only. For testing purposes, we utilized the sub-database containing the first 105

Table 1. Top-1 Classification Accuracy (%) on HKPU-FV, MMCBNU-FV, and CTBU-FV.

Method	FVRAS-Net	FV-CNN	PV-CNN	LW-CNN
Dataset: HKPU-FV				
SimSiam(+Ours)	90.68(93.56)	90.88(93.75)	92.46(93.05)	85.03(87.41)
BYOL(+Ours)	94.84(95.14)	92.17(93.16)	95.34(96.52)	96.63(98.00)
BT(+Ours)	86.56(87.80)	91.67(91.97)	88.50(90.48)	88.40(90.00)
Dataset: MMCBNU-FV				
SimSiam(+Ours)	93.80(97.04)	96.29(96.95)	96.08(97.20)	97.95(98.16)
BYOL(+Ours)	97.75(97.98)	97.04(97.69)	93.36(94.23)	98.20(98.25)
BT(+Ours)	92.25(94.56)	96.33(97.25)	97.04(97.84)	98.54(98.85)
Dataset: CTBU-FV				
SimSiam(+Ours)	89.12(91.23)	89.25(92.56)	88.16(90.78)	91.54(93.02)
BYOL(+Ours)	92.83(94.56)	89.12(91.25)	90.83(91.96)	90.08(91.33)
BT(+Ours)	89.98(90.75)	89.00(90.56)	90.37(92.02)	94.12(94.86)

subjects' finger images from the first session, which more accurately replicates a two-session scenario.The images are grayscale and have a resolution of 256×513.

(2) MMCBNU-FV: The MMCBNU6000 fingerprint database [59] comprises 6,000 images, derived from 100 volunteers who provided six fingers each. During the data capture process, each volunteer placed their fingers - the index, middle, and ring fingers of both the left and right hands on the sensor 10 times, resulting in 10 images being collected for each finger.

(3) CTBU-FV: The Chongqing Technology and Business University finger-vein dataset [60] comprises 6,000 images obtained from 100 subjects (66 males and 34 females). Each subject contributed six fingers, namely the index, middle, and ring fingers from both the left and right hands. For each finger, there were ten images with a size of 240×320 pixels. These images were cropped to extract the region of interest (ROI) images, which were then normalized to 55×127 pixels.

4.2 Evaluation of Representations

The commonly employed method for assessing learned representations during the SSL pre-training process is the linear evaluation protocol [61]. In this protocol, a linear classifier, such as SVM or Logistic Regression, is trained on the frozen backbone network. The image representations, obtained from the final or penultimate layers of the backbone network as a feature vector, are utilized. The test accuracy serves as a proxy for the quality of the representations (Fig. 3(a)). In this work, to assess the learned representations of the proposed method, we conduct experiments on three datasets employing four classifiers within the linear evaluation protocol of self-supervised learning. The experimental results are presented in Table 1.

In Table 1, the results in parentheses represent the performance of CL3A-FV architecture applied to the corresponding SSL objectives, while the results outside the paren-

theses represent the baseline results of the original SSL methods. For instance, in the first row, the results are shown for four different backbones on three original SSL methods (outside the parentheses) and on our corresponding method (inside the parentheses) in the HKPU-FV dataset. From Table 1, it can be observed that there is an improvement across all four classifiers for the HKPU-FV dataset, with the highest margin in recognition accuracy being approximately 3%. However, the improvement effects are generally less pronounced for the MMCBNU-FV dataset. In contrast, for the CTBU-FV dataset, the proposed CL3A-FV method performs better compared to the MMCBNU-FV dataset under the same settings. We speculate that this difference may be attributed to variations in the sample quality. Our approach might be more effective for coarsely-processed data. Models using the proposed framework consistently outperform their respective SSL baselines beyond the margin of error.

4.3 Fine-Tune Evaluation

Fine-tuning [62] is a method that allows us to reuse a pre-trained model and adapt it to a new task. It involves unfreezing some of the top layers of the pre-trained neural network model, which is initially used for feature extraction. As a result, we can jointly train these unfrozen layers along with the newly-added part of the model, such as a fully-connected classifier. This process helps the model to specialize in and adapt to the specific requirements of the new task. This technique is referred to as fine-tuning because it involves making subtle adjustments to the higher-level representations of the pre-trained model (Fig. 3(b)). These adjustments are made to align the model more closely with the specific problem being addressed, thereby increasing its relevance and effectiveness for the task. Only the top layers of the convolutional base are possible to be fine-tuned once the classifier on top has already been trained so as to train a randomly-initialized classifier, freezing of pre-trained convolutional networks like VGG16 [63]. The solution lies in utilizing a smaller learning rate for the weights undergoing fine-tuning, and a higher one for the randomly initialized weights, such as those in the softmax classifier. Pre-trained weights are already good, but they need to be fine-tuned.

To demonstrate that SSL methods can improve the classifier performance without increasing the amount of data, we compare the results of training the entire network from scratch on a labeled dataset with the results of fine-tuning the pre-trained feature extractor using our method with a smaller learning rate. Fine-tuning evaluation results are listed in Table 2. From the table, it can be observed that all classifiers show improved recognition accuracy on different datasets after comparative learning with pre-training and fine-tuning, including the top-performing LW-CNN classifier. Particularly the CTBU-FV dataset has lower sample quality, because the supervised training with LW-CNN exhibits overfitting. However, this issue is significantly alleviated through pre-training and fine-tuning. To more intuitively observe the difference between the results with and without pre-training, we present the fine-tuning results in Fig. 4 on datasets of HKPU-FV, MMCBNU-FV, and CTBU-FV. In Fig. 4, the CTBU-FV dataset may suffer from overfitting due to its more homogeneous sample data and the powerful LW-CNN classifier. Pre-training and fine-tuning can significantly alleviate the overfitting. For the other two datasets, most of them show a significant improvement compared to traditional fully-supervised one-stage training under the same settings, while a small

Table 2. Classification Accuracy (%) of Fine-tuning on HKPU-FV, MMCBNU-FV, and CTBU-FV.

Method	HKPU-FV	MMCBNU-FV	CTBU-FV
Classifier: FVRAS-Net			
SimSiam(+Ours)	95.54(96.23)	96.91(97.58)	95.00(95.70)
BYOL(+Ours)	96.13(97.02)	98.20(98.25)	96.33(96.98)
BT(+Ours)	95.35(96.33)	94.62(97.50)	95.58(96.70)
Scratch	95.44	96.20	93.37
Classifier: FV-CNN			
SimSiam(+Ours)	94.05(95.34)	96.12(96.75)	90.33(91.30)
BYOL(+Ours)	93.85(96.43)	96.04(96.98)	91.08(91.79)
BT(+Ours)	94.74(94.84)	96.54(97.00)	91.20(91.83)
Scratch	94.15	95.95	89.62
Classifier: PV-CNN			
SimSiam(+Ours)	95.56(96.63)	97.05(97.75)	92.79(93.50)
BYOL(+Ours)	95.90(96.33)	97.56(97.91)	94.29(95.05)
BT(+Ours)	92.48(95.44)	96.33(97.08)	93.87(94.52)
Scratch	90.78	95.91	91.86
Classifier: LW-CNN			
SimSiam(+Ours)	98.81(99.00)	98.29(99.13)	93.95(94.88)
BYOL(+Ours)	98.61(99.23)	98.33(98.60)	93.50(95.14)
BT(+Ours)	98.70(98.80)	98.45(98.86)	94.87(95.35)
Scratch	98.00	97.87	85.12

portion may have limited improvement due to factors, such as the model and the dataset itself.

4.4 Ablation Study

To investigate the effectiveness of the proposed distributor, we conduct four comparative experiments on the HKPU-FV dataset for each of the three methods (SimSiam+Ours, BYOL+Ours, and BT+Ours), by removing the distribution constraint MMD or replacing it with Jensen-Shannon (JS) distance, Kullback-Leibler (KL) divergence, and Wasserstein distance, respectively. The results are shown in Table 3. The results in the second row of Table 3 correspond to the unconstrained augmentation network, and when compared to the first row, it can be observed that the MMD distance improves the performance across all three frameworks, where the BYOL+Ours showing the most significant improvement. However, the other three constraint conditions do not yield improvements in every method. We speculate that this may be due to excessive constraints, which results in the augmentation network learning ineffectively.

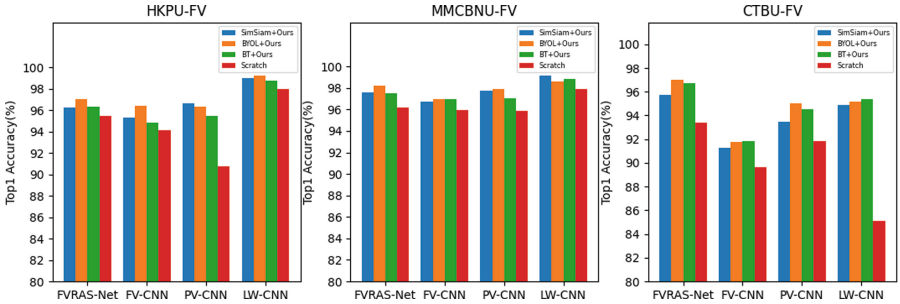

Fig. 4. Fine-tuning evaluation on HKPU-FV, MMCBNU-FV, and CTBU-FV.

Table 3. Classification Accuracy (%) of Ablation Study on HKPU-FV.

Condition		Method	
	SimSiam+Ours	BYOL+Ours	BT+Ours
MMD (Ours)	93.56	95.14	87.80
None	93.06	91.17	87.02
JS	93.16	87.61	86.35
Wasserstein	91.77	90.28	87.88
KL	93.26	89.79	87.65

5 Conclusion

In this paper, we proposed a novel contrastive learning-based finger-vein image recognition with automatic adversarial augmentation, namely CL3A-FV. Our method addresses the limitations of traditional contrastive learning methods by using a neural network-based data augmentation method that generates more effective and diverse augmented samples of finger-vein images. The proposed CL3A-FV achieves the state-of-the-art performance on three finger-vein datasets, outperforming the existing contrastive learning methods that rely on predefined data augmentation operations. The proposed CL3A-FV has several advantages: 1) it is capable of learning more effective data augmentation operations specifically designed for finger-vein images; 2) it reduces the manual efforts required to define data augmentation operations; and 3) it is easy to be integrated into existing contrastive learning frameworks. Moreover, our approach enables the learning of adaptive enhancement pairs for finger-vein images, which improves the performance and robustness of the model. We conduct extensive experiments on three finger-vein datasets and show that CL3A-FV achieves state-of-the-art performance and robustness in vein image recognition tasks, compared to traditional supervised learning methods and the existing contrastive learning methods. Overall, the proposed method provides a novel and effective solution for improving finger vein recognition in scenarios with limited labeled data. We believe that this work will inspire further research in the field of self-supervised methods for biometric identification and have a significant impact on the development of more accurate and reliable biometric identification systems.

However, training the model and generating augmented samples may require significant computational resources, particularly when dealing with large-scale target datasets. Furthermore, we did not consider the inclusion of negative samples. To further enhance the performance of our model, we will investigate methods to introduce negative samples at a lower computational cost in the future.

Acknowledgment. This work was supported in part by the National Natural Science Foundation of China under Grants 62072061 and 61976030, and in part by the Funds for Creative Research Groups of Chongqing Municipal Education Commission under Grant CXQT21034.

References

1. Shaheed, K., Liu, H., Yang, G., Qureshi, I., Gou, J., Yin, Y.: A systematic review of finger vein recognition techniques. Information **9**(9), 213 (2018)
2. Yang, J., Zhang, X.: Feature-level fusion of global and local features for finger-vein recognition. In: IEEE 10th International Conference On Signal Processing Proceedings, pp. 1702–1705. IEEE (2010)
3. Liu, Y., Ling, J., Liu, Z., Shen, J., Gao, C.: Finger vein secure biometric template generation based on deep learning. Soft. Comput. **22**, 2257–2265 (2018)
4. Zhang, D., Zuo, W., Yue, F.: A comparative study of palmprint recognition algorithms. ACM Comput. Surv. (CSUR) **44**(1), 1–37 (2012)
5. Yang, L., Yang, G., Wang, K., Hao, F., Yin, Y.: Finger vein recognition via sparse reconstruction error constrained low-rank representation. IEEE Trans. Inf. Forensics Secur. **16**, 4869–4881 (2021)
6. Miura, N., Nagasaka, A., Miyatake, T.: Feature extraction of finger-vein patterns based on repeated line tracking and its application to personal identification. Mach. Vis. Appl. **15**, 194–203 (2004)
7. Miura, N., Nagasaka, A., Miyatake, T.: Extraction of finger-vein patterns using maximum curvature points in image profiles. IEICE Trans. Inf. Syst. **90**(8), 1185–1194 (2007)
8. Lee, E.C., Jung, H., Kim, D.: New finger biometric method using near infrared imaging. Sensors **11**(3), 2319–2333 (2011)
9. Shaheed, K., et al.: Ds-cnn: a pre-trained xception model based on depth-wise separable convolutional neural network for finger vein recognition. Expert Syst. Appl. **191**, 116288 (2022)
10. Huang, J., Zheng, A., Shakeel, M.S., Yang, W., Kang, W.: Fvfsnet: frequency-spatial coupling network for finger vein authentication. IEEE Trans. Inf. Forensics Secur. **18**, 1322–1334 (2023)
11. Yang, L., Liu, X., Yang, G., Wang, J., Yin, Y.: Small-area finger vein recognition. IEEE Trans. Inf. Forensics Secur. **18**, 1914–1925 (2023)
12. Grill, J.B., et al.: Bootstrap your own latent-a new approach to self-supervised learning. Adv. Neural. Inf. Process. Syst. **33**, 21271–21284 (2020)
13. Chen, X., He, K.: Exploring simple siamese representation learning. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 15750–15758 (2021)
14. Zbontar, J., Jing, L., Misra, I., LeCun, Y., Deny, S.: Barlow twins: self-supervised learning via redundancy reduction. In: International Conference on Machine Learning, pp. 12310–12320. PMLR (2021)
15. Zhang, J., Ma, K.: Rethinking the augmentation module in contrastive learning: learning hierarchical augmentation invariance with expanded views. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 16650–16659 (2022)

16. Krizhevsky, A., Sutskever, I., Hinton, G.E.: Imagenet classification with deep convolutional neural networks. In: Advances in Neural Information Processing Systems 25 (2012)
17. Kim, W., Song, J.M., Park, K.R.: Multimodal biometric recognition based on convolutional neural network by the fusion of finger-vein and finger shape using near-infrared (nir) camera sensor. Sensors **18**(7), 2296 (2018)
18. Xie, C., Kumar, A.: Finger vein identification using convolutional neural network and supervised discrete hashing. Pattern Recogn. Lett. **119**, 148–156 (2019)
19. Fang, Z.M., Lu, Z.M.: Deep belief network based finger vein recognition using histograms of uniform local binary patterns of curvature gray images. Inter. J. Innovative Comput. Inform. Control **15**(5), 1701–1715 (2019)
20. Li, J., Fang, P.: Fvgnn: a novel gnn to finger vein recognition from limited training data. In: 2019 IEEE 8th Joint International Information Technology and Artificial Intelligence Conference (ITAIC), pp. 144–148. IEEE (2019)
21. Huang, Z., Guo, C.: Robust finger vein recognition based on deep cnn with spatial attention and bias field correction. Int. J. Artif. Intell. Tools **30**(01), 2140005 (2021)
22. Huang, J., Tu, M., Yang, W., Kang, W.: Joint attention network for finger vein authentication. IEEE Trans. Instrum. Meas. **70**, 1–11 (2021)
23. Qin, H., El-Yacoubi, M.A.: Deep representation for finger-vein image-quality assessment. IEEE Trans. Circuits Syst. Video Technol. **28**(8), 1677–1693 (2017)
24. Fang, Y., Wu, Q., Kang, W.: A novel finger vein verification system based on two-stream convolutional network learning. Neurocomputing **290**, 100–107 (2018)
25. Nguyen, D.T., Yoon, H.S., Pham, T.D., Park, K.R.: Spoof detection for finger-vein recognition system using nir camera. Sensors **17**(10), 2261 (2017)
26. Hou, B., Yan, R.: Convolutional auto-encoder based deep feature learning for finger-vein verification. In: 2018 IEEE International Symposium on Medical Measurements and Applications (MeMeA), pp. 1–5. IEEE (2018)
27. Kamaruddin, N.M., Rosdi, B.A.: A new filter generation method in pcanet for finger vein recognition. IEEE Access **7**, 132966–132978 (2019)
28. Ma, N., Li, Y., Wang, Y., Ma, S., Lu, H.: Research on roi extraction algorithm for finger vein recognition based on capsule neural network. In: International Conference on Frontiers of Electronics, Information and Computation Technologies, pp. 1–5 (2021)
29. El-Rahiem, B.A., El-Samie, F.E.A., Amin, M.: Multimodal biometric authentication based on deep fusion of electrocardiogram (ecg) and finger vein. Multimedia Syst. **28**(4), 1325–1337 (2022)
30. Zhu, C., Yang, Y., Jang, Y.: Research on denoising of finger vein image based on deep convolutional neural network. In: 2019 14th International Conference on Computer Science & Education (ICCSE), pp. 374–378. IEEE (2019)
31. Guo, X.J., Li, D., Zhang, H.G., Yang, J.F.: Image restoration of finger-vein networks based on encoder-decoder model. Optoelectronics Lett. **15**(6), 463–467 (2019)
32. Yang, S., Qin, H., Liu, X., Wang, J.: Finger-vein pattern restoration with generative adversarial network. IEEE Access **8**, 141080–141089 (2020)
33. He, J., et al.: Finger vein image deblurring using neighbors-based binary-gan (nb-gan). IEEE Trans. Emerging Topics Comput. Intell. (2021)
34. Choi, J., Hong, J.S., Owais, M., Kim, S.G., Park, K.R.: Restoration of motion blurred image by modified deblurgan for enhancing the accuracies of finger-vein recognition. Sensors **21**(14), 4635 (2021)
35. Lei, L., Xi, F., Chen, S.: Finger-vein image enhancement based on pulse coupled neural network. IEEE Access **7**, 57226–57237 (2019)

36. Zeng, J., Wang, F., Qin, C., Gan, J., Zhai, Y., Zhu, B.: A novel method for finger vein segmentation. In: Yu, H., Liu, J., Liu, L., Ju, Z., Liu, Y., Zhou, D. (eds.) ICIRA 2019. LNCS (LNAI), vol. 11741, pp. 589–600. Springer, Cham (2019). https://doi.org/10.1007/978-3-030-27532-7_52
37. Zeng, J.: Real-time segmentation method of lightweight network for finger vein using embedded terminal technique. IEEE Access **9**, 303–316 (2020)
38. Wu, Z., Xiong, Y., Yu, S.X., Lin, D.: Unsupervised feature learning via non-parametric instance discrimination. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 3733–3742 (2018)
39. Chen, T., Kornblith, S., Norouzi, M., Hinton, G.: A simple framework for contrastive learning of visual representations. In: International Conference on Machine Learning, pp. 1597–1607. PMLR (2020)
40. Tian, Y., Krishnan, D., Isola, P.: Contrastive multiview coding. In: Vedaldi, A., Bischof, H., Brox, T., Frahm, J.-M. (eds.) ECCV 2020. LNCS, vol. 12356, pp. 776–794. Springer, Cham (2020). https://doi.org/10.1007/978-3-030-58621-8_45
41. Oord, A.v.d., Li, Y., Vinyals, O.: Representation learning with contrastive predictive coding. arXiv preprint arXiv:1807.03748 (2018)
42. Ho, D., Liang, E., Chen, X., Stoica, I., Abbeel, P.: Population based augmentation: efficient learning of augmentation policy schedules. In: International Conference on Machine Learning, pp. 2731–2741. PMLR (2019)
43. Lim, S., Kim, I., Kim, T., Kim, C., Kim, S.: Fast autoaugment. In: Advances in Neural Information Processing Systems 32 (2019)
44. Hataya, R., Zdenek, J., Yoshizoe, K., Nakayama, H.: Faster autoaugment: learning augmentation strategies using backpropagation. In: Vedaldi, A., Bischof, H., Brox, T., Frahm, J.-M. (eds.) ECCV 2020. LNCS, vol. 12370, pp. 1–16. Springer, Cham (2020). https://doi.org/10.1007/978-3-030-58595-2_1
45. Cubuk, E.D., Zoph, B., Shlens, J., Le, Q.V.: Randaugment: practical automated data augmentation with a reduced search space. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops, pp. 702–703 (2020)
46. Cubuk, E.D., Zoph, B., Mane, D., Vasudevan, V., Le, Q.V.: Autoaugment: learning augmentation policies from data. arXiv preprint arXiv:1805.09501 (2018)
47. Bachman, P., Hjelm, R.D., Buchwalter, W.: Learning representations by maximizing mutual information across views. In: Advances in Neural Information Processing Systems 32 (2019)
48. Dosovitskiy, A., Springenberg, J.T., Riedmiller, M., Brox, T.: Discriminative unsupervised feature learning with convolutional neural networks. In: Advances in Neural Information Processing Systems 27 (2014)
49. Jaderberg, M., Simonyan, K., Zisserman, A., et al.: Spatial transformer networks. In: Advances in Neural Information Processing Systems 28 (2015)
50. Xie, C., Tan, M., Gong, B., Wang, J., Yuille, A.L., Le, Q.V.: Adversarial examples improve image recognition. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 819–828 (2020)
51. Madry, A., Makelov, A., Schmidt, L., Tsipras, D., Vladu, A.: Towards deep learning models resistant to adversarial attacks. arXiv preprint arXiv:1706.06083 (2017)
52. Raghunathan, A., Xie, S.M., Yang, F., Duchi, J.C., Liang, P.: Adversarial training can hurt generalization. arXiv preprint arXiv:1906.06032 (2019)
53. Tang, Z., Gao, Y., Karlinsky, L., Sattigeri, P., Feris, R., Metaxas, D.: OnlineAugment: online data augmentation with less domain knowledge. In: Vedaldi, A., Bischof, H., Brox, T., Frahm, J.-M. (eds.) ECCV 2020. LNCS, vol. 12352, pp. 313–329. Springer, Cham (2020). https://doi.org/10.1007/978-3-030-58571-6_19

54. Yang, W., Luo, W., Kang, W., Huang, Z., Wu, Q.: Fvras-net: an embedded finger-vein recognition and antispoofing system using a unified cnn. IEEE Trans. Instrum. Meas. **69**(11), 8690–8701 (2020)
55. Das, R., Piciucco, E., Maiorana, E., Campisi, P.: Convolutional neural network for finger-vein-based biometric identification. IEEE Trans. Inf. Forensics Secur. **14**(2), 360–373 (2018)
56. Qin, H., El-Yacoubi, M.A., Li, Y., Liu, C.: Multi-scale and multi-direction gan for cnn-based single palm-vein identification. IEEE Trans. Inf. Forensics Secur. **16**, 2652–2666 (2021)
57. Shen, J., et al.: Finger vein recognition algorithm based on lightweight deep convolutional neural network. IEEE Trans. Instrum. Meas. **71**, 1–13 (2021)
58. Kumar, A., Zhou, Y.: Human identification using finger images. IEEE Trans. Image Process. **21**(4), 2228–2244 (2011)
59. Lu, Y., Xie, S.J., Yoon, S., Yang, J., Park, D.S.: Robust finger vein roi localization based on flexible segmentation. Sensors **13**(11), 14339–14366 (2013)
60. Qin, H., Hu, R., El-Yacoubi, M.A., Li, Y., Gao, X.: Local attention transformer-based full-view finger-vein identification. IEEE Trans. Circuits Syst. Video Technol. **33**(6), 2767–2782 (2023)
61. Kolesnikov, A., Zhai, X., Beyer, L.: Revisiting self-supervised visual representation learning. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 1920–1929 (2019)
62. Peng, P., Wang, J.: How to fine-tune deep neural networks in few-shot learning? arXiv preprint arXiv:2012.00204 (2020)
63. Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556 (2014)