

Article

Finger Vein Recognition Using DenseNet with a Channel Attention Mechanism and Hybrid Pooling

Nikesh Devkota and Byung Wook Kim * 

Department of Information and Communication Engineering, Changwon National University, Changwon 51140, Republic of Korea; 20227085@gs.cwnu.ac.kr

* Correspondence: bwkim@changwon.ac.kr

Abstract: This paper proposes SE-DenseNet-HP, a novel finger vein recognition model that integrates DenseNet with a squeeze-and-excitation (SE)-based channel attention mechanism and a hybrid pooling (HP) mechanism. To distinctively separate the finger vein patterns from their background, original finger vein images are preprocessed using region-of-interest (ROI) extraction, contrast enhancement, median filtering, adaptive thresholding, and morphological operations. The preprocessed images are then fed to SE-DenseNet-HP for robust feature extraction and recognition. The DenseNet-based backbone improves information flow by enhancing feature propagation and encouraging feature reuse through feature map concatenation. The SE module utilizes a channel attention mechanism to emphasize the important features related to finger vein patterns while suppressing less important ones. HP architecture used in the transitional blocks of SE-DenseNet-HP concatenates the average pooling method with a max pooling strategy to preserve both the most discriminative and contextual information. SE-DenseNet-HP achieved recognition accuracy of 99.35% and 93.28% on the good-quality FVUSM and HKPU datasets, respectively, surpassing the performance of existing methodologies. Additionally, it demonstrated better generalization performance on the FVUSM, HKPU, UTFVP, and MMCBNU_6000 datasets, achieving remarkably low equal error rates (EERs) of 0.03%, 1.81%, 0.43%, and 1.80%, respectively.

Keywords: biometrics; channel attention; deep neural network; DenseNet; finger vein recognition; hybrid; pooling



Citation: Devkota, N.; Kim, B.W. Finger Vein Recognition Using DenseNet with a Channel Attention Mechanism and Hybrid Pooling. *Electronics* **2024**, *13*, 501. <https://doi.org/10.3390/electronics13030501>

Academic Editors: Luca Mesin, Antonio Lanata, Irena Galić and Marija Habijan

Received: 20 December 2023

Revised: 17 January 2024

Accepted: 23 January 2024

Published: 25 January 2024



Copyright: © 2024 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

With the increasing preference for secure recognition systems, the use of biometric traits for human recognition has received a great deal of attention in recent years. Biometric techniques, as opposed to traditional authentication methods such as secret keys and passwords, are less vulnerable to theft or duplication. These techniques utilize either physiological and/or behavioral characteristics to authenticate a person through partially or completely automated methods [1,2]. Biometric characteristics can be divided into two categories: extrinsic characteristics and intrinsic characteristics. Biometric systems that use extrinsic characteristics usually rely on external features of the human body, such as fingerprints, faces, and irises [3]. Even though biometric systems that utilize extrinsic characteristics are less susceptible to theft than conventional methods, they are vulnerable in practice to forged input, raising privacy and security concerns [4]. In contrast, biometric systems use intrinsic characteristics such as finger veins and palm veins that are hidden beneath the skin, which are difficult to forge [5–7]. Particularly, finger vein authentication systems have demonstrated promising applications in consumer electronics, banking, and airports [8].

Finger vein recognition is an intrinsic biometric technique that uses human finger vein patterns captured under near-infrared (NIR) illumination as the basis for biometric authentication [9]. During the image acquisition process, a finger is placed between the

NIR light source and a camera [10]. When NIR light is directed at the finger, it is either absorbed or reflected depending on the thickness of the skin and the blood flow in the finger. As hemoglobin in the blood flowing through finger veins absorbs more NIR light than other tissues, the amount of NIR light transmitted through the blood vessels is reduced. Hence, the finger veins appear as dark lines in the acquired image. Even though finger vein recognition has been researched for many years, it is still a challenging task. During real-time acquisition, the process of acquiring finger vein images is susceptible to quality degradation. Degradation occurs for many reasons: light scattering when imaging the finger tissue, uneven muscle thickness within the finger, and physiological changes like inaccurate placement of the finger [11]. Hence, finger vein images need to be preprocessed using multiple image processing methods to acquire significant information [12].

The existing finger vein recognition models are categorized as nonlearning models and learning models [13]. Nonlearning models [14–17], which rely on rule-based approaches, mainly depend on input image quality and texture deletion. Additionally, these methods use handcrafted features and fixed hyperparameters, which may not generalize well to data acquired in different environmental settings, leading to lower recognition accuracy [18]. To address these issues and enhance generalization performance, researchers started to develop finger vein recognition algorithms based on learning models, which are further classified into machine learning (ML) [19–22] and deep learning (DL) [23–26]. Conventional ML-based methods, such as principal component analysis (PCA) and linear discriminant analysis (LDA), transform an image into low-dimensional space before classification [27]. Finger vein feature extraction using these techniques is often not robust to local appearance variations that are caused by rotation, scale, skin scattering, uneven illumination, and other factors [28]. DL methods consisting of convolutional neural networks (CNNs) overcome these limitations by automatically learning discriminative features from finger vein images [29]. However, a more advanced DL structure, such as DenseNet [30], and additional processing steps are required to precisely learn the detailed information of finger veins that represent complex patterns.

In this paper, we propose SE-DenseNet-HP architecture, which integrates DenseNet with a squeeze-and-excitation (SE)-based channel attention mechanism and a hybrid pooling (HP) mechanism for robust feature extraction and recognition of finger veins. As acquired finger vein images contain unsatisfactory illumination and noise distortions, a preprocessing stage is required before feeding input images into a DL network. To observe clear vein patterns in finger images, our preprocessing stage consists of region-of-interest (ROI) extraction, contrast enhancement, median filtering, adaptive thresholding, and morphological operations. The preprocessed image is then fed into the SE-DenseNet-HP architecture for robust feature extraction. The DenseNet backbone in SE-DenseNet-HP improves finger vein information flow and the gradients between neural network layers by combining the feature maps from all preceding levels that have the same spatial dimension. The channel attention mechanism based on the SE module [31] enhances the feature extraction process of the SE-DenseNet-HP architecture by putting more emphasis on the feature map channels that contain discriminative characteristics of the finger vein patterns. The HP process in SE-DenseNet-HP optimizes the feature extraction performance of the architecture by combining average pooling and max pooling to acquire generalized global information as well as detailed local information. From the simulation results, we show that the proposed SE-DenseNet-HP model can achieve recognition accuracies of up to 99.35% and 99.25%, respectively, for good- and poor-quality images of the Finger Vein University Sains Malaysia (FVUSM) [32] dataset and 93.28% and 88.16%, respectively, for good- and poor-quality images of the Hong Kong Polytechnic University (HKPU) [16] dataset. In addition, it is able to achieve an equal error rate (EER) of 0.03%, 1.81%, 0.43%, and 1.80% for the FVUSM, HKPU, University of Twente Finger Vascular Pattern (UTFVP) [33], and MMCBNU_6000 [34] datasets, respectively.

The motivation for the proposed combination of image preprocessing techniques is to create a preprocessing pipeline that can adapt to various environmental conditions for the acquisition of finger vein images. In addition, the specific adaptation and combination of DenseNet, the SE block, and HP architecture in the context of finger vein recognition have not been thoroughly explored in the existing literature; thus, we conducted extensive simulations with various datasets to design DNN architecture with these components, resulting in significant enhancements in accuracy and generalization performance. Overall, the major contributions of this study to improve finger vein recognition are as follows:

- To successfully separate vein patterns from the background in finger images, we develop an image preprocessing stage that consists of ROI extraction, contrast enhancement, median filtering, adaptive thresholding, and morphological operations. The quality of the preprocessed image is compared with conventional preprocessing;
- To enhance the performance of the DenseNet backbone model, the SE-based channel attention mechanism and the HP strategy are integrated into the DenseNet network structure. The SE module emphasizes the important features related to finger vein patterns while suppressing less important ones. The HP process used in the transitional blocks of SE-DenseNet-HP concatenates the average pooling method with a max pooling strategy to preserve the most discriminative and contextual information;
- To show the novelty of our proposed system, we compare the finger vein recognition performance of SE-DenseNet-HP with existing recognition approaches in terms of recognition accuracy, the receiver operating characteristic (ROC) curve, and the EER.

The rest of this paper is organized as follows: Section 2 describes previous works related to finger vein recognition. Section 3 provides a detailed explanation of our SE-DenseNet-HP model. Section 4 presents the experiments and results of our work. Finally, Section 5 provides the conclusion of our study.

2. Related Works

A finger vein recognition system is composed of four components: image acquisition, image preprocessing, feature extraction, and matching [35]. After acquiring vein images with NIR optical imaging techniques, quality enhancement is accomplished through several preprocessing steps, such as image filtering, ROI extraction, and image normalization. Then, feature extraction is performed on the enhanced images to extract discriminative features from individual vein images. Using the feature-matching technique from the extracted vein features, biometric recognition can be performed. Several studies have been conducted in this regard to develop finger vein recognition systems using rule-based methods, ML, and DL.

Rule-based methods often extract finger vein patterns using handcrafted feature extraction methods, and they utilize matching techniques based on extracted finger vein features. Miura et al. used a repeated line tracking (RLT) [14] feature extraction method to recognize finger veins. In that method, finger vein patterns were extracted using randomly varying start points. The same authors later proposed a new method for extracting vein patterns by detecting local maximum curvatures in the cross-sectional profiles of finger vein images with varying widths and brightness levels [15]. Kumar et al. [16] demonstrated a method for extracting finger veins using Gabor filters. Lee et al. [17] proposed a feature fusion method for finger veins that employs simple binarization, a local binary pattern (LBP), and a local derivative pattern (LDP). Features are extracted based on binary codes acquired by comparing the intensity value of the center pixel to the neighboring pixels. After extracting the vein features, the above approaches employ distance-based feature-matching algorithms, such as Hamming distance, Euclidean distance cross-correlation matching, and template matching. However, finger vein recognition based on the rule-based approach can work well only with specific NIR images and related system parameters [18]. As various NIR imaging devices and image-capture environments should be considered in biometric applications, rule-based approaches show limitations in generalization performance.

Several methods in ML algorithms have been used for finger vein recognition. He et al. [19] used PCA to extract key features of finger veins, such as shape and orientation and then constructed a multilayer neural network classifier for finger vein recognition. Khellat-Kihel et al. [20] proposed a support vector machine (SVM)-based finger vein recognition system. In their paper, they utilized a Gabor filtering method for feature extraction and an SVM algorithm for both classification and recognition. A finger vein recognition algorithm combining kernel principal component analysis (KPCA) and a weighted k-nearest centroid neighbor (WKNCN) was proposed by Mobarakeh et al. [21]. Kumar et al. [22] proposed integrated responses of texture (IRT) for feature extraction and a multi-SVM for classification. However, these methods are still inferior to DL methods because they are less robust to variations in acquired finger vein patterns caused by rotation, scale, skin scattering, uneven illumination, and other factors [26].

Recently, DL techniques have had a huge impact in various fields, including biometrics system recognition, like finger vein authentication. DL can learn the finger vein patterns from acquired images in a reliable manner, even with variations in the shape and orientation of the acquired finger vein images. Wenming et al. [23] used a generative adversarial network (GAN) for finger vein extraction and recognition. Qin and El-Yacoubi [24] utilized a deep neural network (DNN) for finger vein quality assessment and recognition. Das et al. [25] used a CNN to produce steady and highly accurate results when dealing with finger vein images of varying quality. However, the CNN they used had only five convolutional layers, which is not sufficient for robust feature extraction. To obtain better recognition accuracy, FVR-Net (a CNN with an HP technique) was used by Tamang and Kim [26]. The HP strategy used in their study exploits max pooling and average pooling in parallel. Max pooling generates sharp features that identify high activation values within the feature map. In contrast, average pooling is a more generalized computation that encourages networks to identify the complete extent of their input volume. However, the image preprocessing steps used in their study cannot guarantee high-quality vein patterns. Additionally, the FVR-Net lacks sufficient learning capabilities to recognize complex patterns in the finger vein images owing to the shallow network depth. Therefore, advanced preprocessing methods and a DNN architecture need to be introduced to enhance the performance of finger vein recognition.

3. Proposed Method

3.1. Data Preprocessing

Due to the restrictions of imaging environments, captured finger vein images easily degrade due to low contrast and image blur. This degradation occurs because of light scattering when imaging finger tissue, uneven muscle thickness within the finger, and physiological changes such as incorrect finger placement [28]. As a result, acquired images contain unsatisfactory illumination or strong noise distortions. Therefore, the finger vein images must be preprocessed before being sent to the neural network for training and testing [26]. The preprocessing stage is designed to remove any nonideal entities that appear in the vein images, such as noise, shadows, and low contrast, which are caused by a rotational or translational variant property of the finger or a faulty acquisition device.

A block diagram of the image preprocessing stage used in the proposed scheme is illustrated in Figure 1. First, ROI extraction is carried out by identifying the pixel areas where finger vein patterns are located. By obtaining the ROI that contains the most salient finger vein information from the input image and eliminating unnecessary regions such as the background, finger vein recognition systems can increase the accuracy in pattern recognition [36]. The contrast between finger vein patterns and the background is then improved using the contrast limited adaptive histogram equalization (CLAHE) contrast enhancement method [37], making the finger vein pattern more distinct from the background and easier to extract [26]. The contrast-enhanced ROI image is then subjected to a median blur filter, which eliminates unnecessary salt-and-pepper noise that might be present in finger vein images [38,39]. In the subsequent image preprocessing step,

adaptive thresholding [40] is applied to the noise-free enhanced ROI image to separate the finger vein patterns from the background. The extraction of vein patterns from finger vein images can be challenging due to uneven lighting and shadows created during the image acquisition process. The adaptive thresholding technique overcomes this difficulty by determining the threshold value locally for each pixel or region, which enables the threshold to adapt dynamically to illumination fluctuations present in various regions of the image [41]. Hence, adaptive thresholding facilitates a more precise separation of finger vein patterns from the background. Finally, morphological closing, which incorporates dilation followed by erosion, is applied to the binarized image to enhance and refine the acquired finger vein patterns. These processes aid in enhancing the quality of the binary image produced by the adaptive thresholding method and make it easier to extract vein features with more reliability and accuracy.

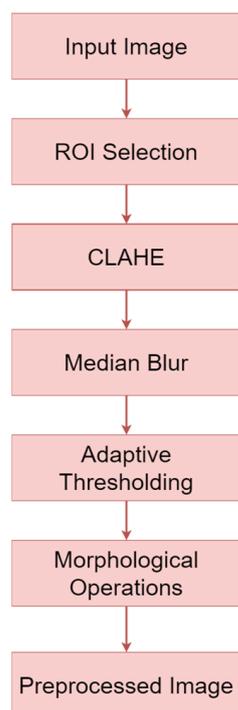


Figure 1. Block diagram of the image preprocessing stage used in finger vein recognition.

The ROI extraction process to remove unnecessary background from a sample image obtained during image acquisition is shown in Figure 2. The original finger vein image is first subjected to a Sobel edge detector, which identifies the ROI in the image of the finger. A global thresholding technique is then used to convert the detected edges into a binary image format. This binarization of edge detection from the image provides a clear separation of edge and nonedge regions more accurately. The finger vein patterns may become distorted or misaligned when finger vein images are acquired at various rotation angles during the image acquisition process. Due to this misalignment, the accuracy of finger recognition can be reduced. Hence, in the subsequent process, rotation correction is applied to the binarized image to standardize the orientation of finger vein patterns and minimize the impact of rotational variations during finger vein recognition. Finally, the rotated image is cropped to a dimension of 100×300 pixels for extracting the ROI.

The extracted ROI is further preprocessed to obtain clear finger vein patterns, as shown in Figure 3. After extracting the ROI from the original input image, the contrast between the finger vein patterns and surrounding background is increased using CLAHE. A median filter is then applied to the contrast-enhanced image, using the median value of adjacent pixels to remove unnecessary salt-and-pepper noise that may be present in the finger vein images. The median filter preserves edge features and fine details in images better than a

mean filter because it replaces each pixel value with the median value of the neighboring pixels rather than the mean [38]. Subsequently, we apply an adaptive thresholding method to the noise-free image. Based on changes in local image intensity, adaptive thresholding divides images into foreground and background areas [40,41]. This is particularly useful in finger vein recognition systems because it separates the finger vein patterns from their surroundings. As it computes the threshold value locally at each pixel based on neighboring pixel values, it can adapt to local variations in the illumination and contrast seen in the finger vein images. Following adaptive thresholding, finger vein images might have small gaps in the connectivity of finger vein patterns. A morphological closing operation closes these gaps by extending veins to cover the gaps and then contracting them to the previous size. This enhances the finger vein patterns obtained from adaptive thresholding by enhancing connectivity in the patterns. Hence, the binarized finger vein patterns obtained after adaptive thresholding are subsequently enhanced using morphological closing, which incorporates dilation followed by erosion. After the completion of the image preprocessing stage, the obtained finger vein images are passed to the SE-DenseNet-HP model for robust feature extraction of the preprocessed finger vein patterns.

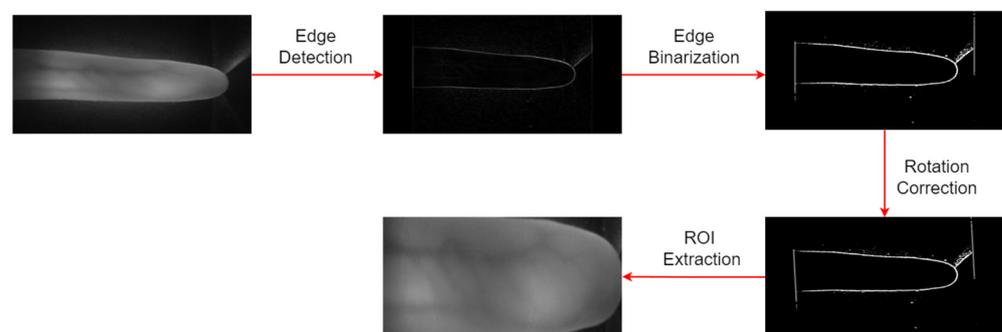


Figure 2. ROI image extraction to remove unnecessary background in a sample image from the FVUSM dataset.

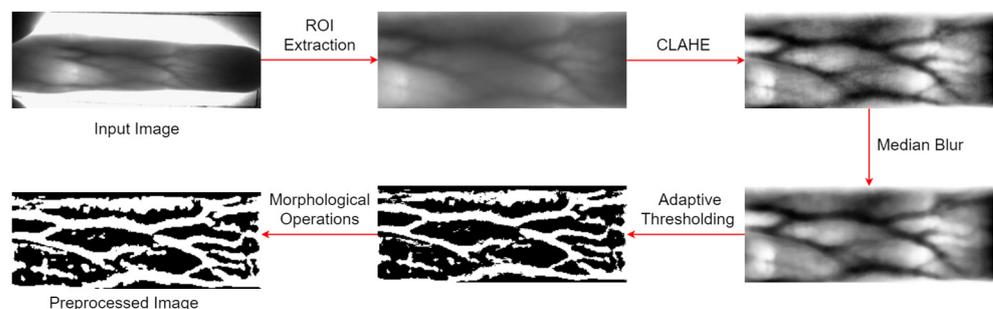


Figure 3. Image preprocessing steps to obtain clear finger vein patterns from an original finger image in HKPU dataset.

3.2. SE-DenseNet-HP Architecture

Due to the limitations on the generalization performance of rule-based and ML approaches, we adopted a DL-based finger recognition system. For feature extraction and classification, we use the backbone network of a DenseNet169 architecture [30] that incorporates an HP strategy and channel attention mechanism. DenseNet optimizes the flow of information and gradients between neural network layers by concatenating feature maps from all preceding layers with the same spatial dimension. The SE [31] module-based channel attention mechanism enables the network to adaptively recalibrate the relevance of different feature channels, allowing it to focus on more discriminative characteristics of finger vein patterns. HP combines the average pooling method, which achieves improved feature localization, with the max pooling strategy, which activates the most discrete features of the input.

In the DenseNet structure, each layer receives additional input from all the layers that precede it and passes on its own feature maps to all subsequent layers with the same spatial dimension. Consider an input image, x_0 , that passes through a neural network of n layers. Every l th layer in the network consists of a nonlinear transformation, $H_l(\cdot)$, which includes pooling, batch normalization (BN), a rectified linear unit (ReLU), or convolution. The output of the $(l - 1)$ th layer, denoted as x_{l-1} , is connected as an input to the l th layer in traditional convolutional feed-forward networks, as shown in (1):

$$x_l = H_l(x_{l-1}). \quad (1)$$

It is known that ResNet [42] bypasses nonlinear transformations with an identity function by incorporating a skip connection. In ResNet, the gradient can flow directly from latter layers to earlier layers through the identity function, as shown in (2):

$$x_l = H_l(x_{l-1}) + x_{l-1}. \quad (2)$$

However, the summation is used to combine the identity function and the output of H_l , which may impede information flow in the network. To improve information flow between layers even further, DenseNet utilizes direct connections from any layer to all subsequent layers. As a result, the feature maps of all preceding layers, x_0, \dots, x_{l-1} , are passed to the l th layer as the input, which is shown in (3):

$$x_l = H_l([x_0, x_1, \dots, x_{l-1}]), \quad (3)$$

where $[x_0, x_1, \dots, x_{l-1}]$ denotes the concatenation of feature maps produced in the $0, \dots, l - 1$ layers. In DenseNet, the number of channels generated by feature map concatenation in deeper layers increases significantly if the size of the feature maps remains the same. This might result in an increase in the computational and memory costs of the network. Hence, to address this issue, downsampling layers are used in DenseNet to decrease the size of feature maps. The downsampling layers divide the DenseNet architecture into numerous dense blocks, each consisting of feature maps of different sizes.

Figure 4 illustrates feature map concatenation within a single dense block. As shown in Figure 4a, each layer inside a dense block is connected to its preceding layers to improve information flow between layers and resolve the vanishing gradient problem. However, as the number of layers within a dense block increases, the number of channels in the concatenated feature maps significantly increases. This can result in a network that is both computationally expensive and memory-intensive. To solve this issue, a bottleneck layer is introduced in the dense block. The bottleneck layer consists of a BN, an ReLU, and a 1×1 convolution, followed by another BN, ReLU, and 3×3 convolution, as shown in Figure 4. First, the input feature maps are fed into the BN layer, and nonlinearity is added to the normalized input by the ReLU activation function. Then, the 1×1 convolution operation is used to reduce the amount of input feature maps and improve computational efficiency. This is followed by another BN layer and ReLU activation. The 3×3 convolution operation applied to the input feature maps helps in the extraction of high-level features from the normalized input data.

The SE module in Figure 4c is utilized after the bottleneck layer to recalibrate the importance of different feature map channels. The reason for using an SE module lies in the fact that different channels within feature maps carry different amounts of meaningful information. Standard DNNs treat all channels equally and independently, and they may be unable to capture and highlight the most discriminating elements of feature maps. By explicitly modeling the channel dependencies and recalibrating their relative importance, the SE module shows robust performance in distinguishing important characteristics of finger vein patterns. The SE module's initial squeeze operation, which uses global average pooling, compresses each feature channel's spatial dimensions to a single value. The channel descriptor, which is the result of this pooling operation, gathers channel-

specific information. Hence, each feature channel's significance or relevance inside a feature map is described by the channel descriptor. The excitation mechanism takes the channel descriptor and learns a set of adaptive weights that represent the importance of each channel. These weights capture the relevance of each channel in relation to the others. The original feature map is then rescaled using the adaptive weights that were generated during the excitation mechanism. Hence, by using the SE module inside a dense block, the SE-DenseNet-HP architecture selectively emphasizes the differentiating characteristics of finger vein patterns while suppressing less important ones during the feature extraction process by multiplying each channel by its appropriate weight. Lastly, the input feature maps of all preceding layers are concatenated with the output feature maps of each layer inside a dense block, as shown in Figure 4. This feature map concatenation improves gradient flow through the network, resulting in higher performance accuracy of the finger vein recognition system.

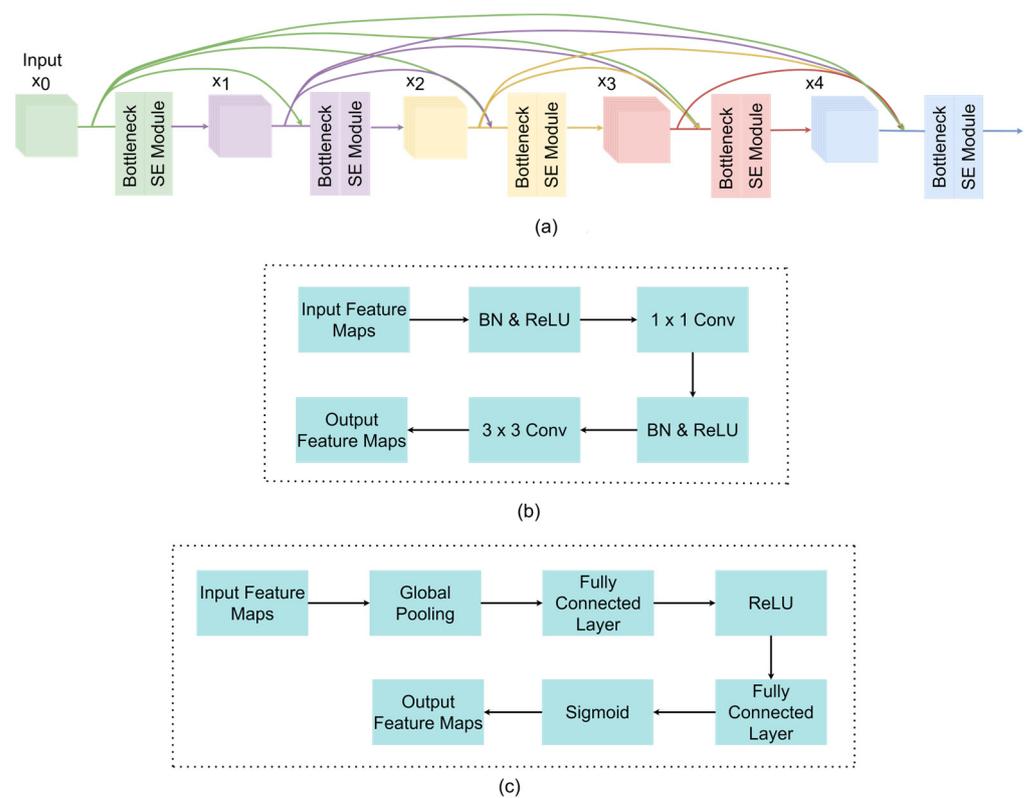


Figure 4. Denseblock architecture inside SE-DenseNet-HP. (a) Feature map concatenation inside a dense block of the SE-DenseNet-HP architecture. (b) Detailed structure of bottleneck layer. (c) Detailed structure of SE module.

As the depth of the DenseNet-based architecture is large, the number of channels generated by feature map concatenation is substantially huge, resulting in an increase in the computational and memory cost of the network. Hence, transitional blocks are placed between the dense blocks in SE-DenseNet-HP to reduce the size of feature maps obtained from the dense blocks. Figure 5 illustrates a block diagram of the transitional blocks used in SE-DenseNet-HP. The output of the prior dense block is fed as an input to the transitional block. First, the BN layer and ReLU activation function are used to normalize the input and provide nonlinearity, respectively. Then, an HP layer is followed by a 1×1 convolution layer to reduce the size of the feature maps. The lower resolution output of the transitional block is fed to the subsequent dense block. The original DenseNet [30] uses 2×2 average pooling layers for the pooling strategy in transitional layers. However, average pooling uses the average of all the values in each pooling region [41], which can lead to the loss of

discriminative information in finger vein patterns. Therefore, in our research, we propose an HP method for the pooling strategy in transitional layers.

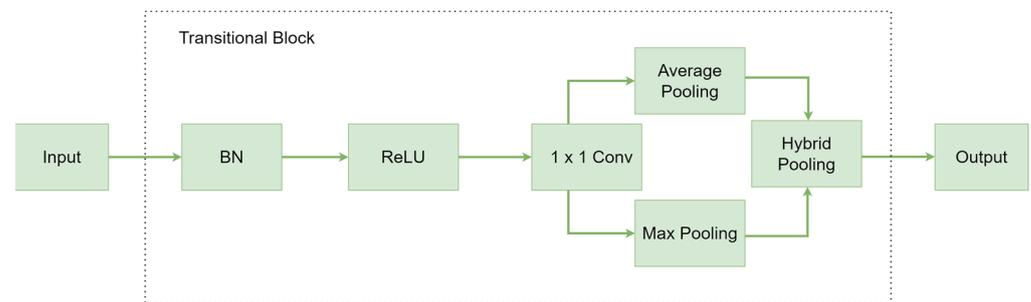


Figure 5. Block diagram of transitional block used in SE-DenseNet-HP architecture.

HP is a feature pooling strategy that integrates the advantages of both max pooling and average pooling. In contrast to average pooling, which selects the average of all values in each pooling region, max pooling is a technique that takes the maximum value in each pooling region [43]. Thus, the HP approach can capture both the overall structure of the input feature maps and the most discriminative features by combining both pooling techniques. In our experiment, the HP strategy combines the output from average pooling and max pooling layers, which are concatenated and then passed to the subsequent layer.

Using the above-mentioned dense block and transitional block, the overall architecture of SE-DenseNet-HP is presented in Figure 6. As can be seen, the proposed network consists of four dense blocks, three transitional blocks, and a classification layer. First, the input image passes through a 5×5 convolution layer, followed by an HP operation. The SE-DenseNet-HP architecture creates an initial feature map representation of the input finger vein image using the first convolution operation. The HP strategy reduces the dimension of the feature maps obtained after the first convolution operation. The output from the HP operation is used as an input for the first dense block. Dense blocks are the main blocks of the SE-DenseNet-HP architecture. Each dense block consists of multiple bottleneck layers and SE modules. The bottleneck layer reduces the amount of input feature maps and improves computational efficiency. During the feature extraction process, the SE module selectively emphasizes distinctive characteristics of finger vein patterns while suppressing less significant ones. Finally, each layer's output feature maps obtained after the SE operation are concatenated with the input feature maps of every subsequent layer inside a dense block to improve network performance by enhancing information flow between these layers. Transitional blocks in the SE-DenseNet-HP architecture are used as connectors between dense blocks, changing the size of feature maps and managing the flow of information throughout the network. After Dense Block 4, the feature map is fed into the classification layer to generate a one-dimensional vector as output. The output vector consists of the probabilities that the input data belong to one of the S classes. Then, the model selects the class with the highest probability from the output vector as the prediction result for finger vein recognition. The detailed architecture of SE-DenseNet-HP is presented in Table 1.

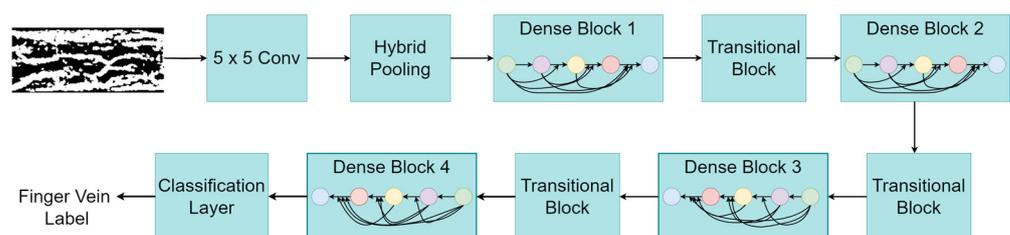


Figure 6. SE-DenseNet-HP network architecture for finger vein extraction and recognition. The color circles represent feature map concatenation between subsequent layers inside a dense block.

Table 1. Architecture of SE-DenseNet-HP.

| Layers | | Input Size | Output Size |
|--------------------------------|--|---------------------------|---------------------------|
| Input | | $100 \times 300 \times 1$ | $100 \times 300 \times 1$ |
| 5×5 conv | | $100 \times 300 \times 1$ | $48 \times 148 \times 64$ |
| 3×3 HP | | $48 \times 148 \times 64$ | $23 \times 73 \times 128$ |
| Dense Block 1 | $\begin{pmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{pmatrix} \times 6$ | $23 \times 73 \times 128$ | $23 \times 73 \times 320$ |
| Transitional Block | $\begin{pmatrix} 1 \times 1 \text{ conv} \\ 2 \times 2 \text{ HP} \end{pmatrix}$ | $23 \times 73 \times 320$ | $11 \times 36 \times 320$ |
| Dense Block 2 | $\begin{pmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{pmatrix} \times 12$ | $11 \times 36 \times 320$ | $11 \times 36 \times 704$ |
| Transitional Block | $\begin{pmatrix} 1 \times 1 \text{ conv} \\ 2 \times 2 \text{ HP} \end{pmatrix}$ | $11 \times 36 \times 704$ | $5 \times 18 \times 704$ |
| Dense Block 3 | $\begin{pmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{pmatrix} \times 32$ | $5 \times 18 \times 704$ | $5 \times 18 \times 2240$ |
| Transitional Block | $\begin{pmatrix} 1 \times 1 \text{ conv} \\ 2 \times 2 \text{ HP} \end{pmatrix}$ | $5 \times 18 \times 2240$ | $2 \times 9 \times 2240$ |
| Dense Block 4 | $\begin{pmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{pmatrix} \times 32$ | $2 \times 9 \times 2240$ | $2 \times 9 \times 3264$ |
| Classification Layer | Global average pooling, 2D | $2 \times 9 \times 3264$ | (None, 3264) |
| Fully connected layer, SoftMax | | (None, 3264) | S |

4. Experiments and Results

In this section, we evaluate the performance of finger vein recognition from the proposed model on four publicly available datasets: HKPU [16], FVUSM [32], UTFVP [33], and MMCBNU_6000 [34]. As the number of images per class is not enough to train the model, we utilized a data augmentation technique to increase the number of training images. Initially, we used recognition accuracy for good- and poor-quality finger vein images for HKPU and FVUSM, as in [26]. To prove that our proposed model can generalize better in additional datasets compared to the previous model used in [26], we used UTFVP and MMCBNU_6000, along with HKPU and FVUSM datasets, for computing the EER and receiver operating characteristic (ROC) curve. All experiments were conducted using Tensorflow 2.10.1 in Python on a Windows 10 operating system with Nvidia GeForce RTX 3080 Ti GPU and 32 GB memory.

4.1. Experiment Datasets

The effectiveness of our proposed finger vein model was tested on the HKPU, FVUSM, MMCBNU_6000, and UTFVP datasets. The HKPU dataset comprises finger vein images from male and female volunteers obtained from a contactless imaging device. The dataset was largely compiled between April 2009 and March 2010. It contains 3132 images from 156 classes, with the first 105 images captured in two separate sessions. The shortest, longest, and average time between sessions was one month, more than six months, and 66.8 days, respectively. Each of the 105 participants gave six samples of their index and middle fingers. The remaining 51 classes' images were captured in a single session. In our study, we used all 156 classes—a single session—from the first 105 classes in addition to the single session from the remaining 51 classes—yielding a total of 1872 images. Each finger was treated as a separate class, yielding a total of 312 classes. Each class consists of six

finger vein images. Figure 7a,b illustrate sample original HKPU images along with their preprocessed images, respectively.

The FVUSM dataset contains finger vein images from 123 participants with ages ranging from 20 to 52 years. Each person provided images of four fingers—left index, left middle, right index, and right middle—for a total of 492 finger classes. Each finger was imaged six times in one session, and each person took part in two sessions separated by more than two weeks. From the two sessions, 5904 images from 492 finger classes were obtained. In our study, we used only one session with a total of 2952 images. Sample original images, along with their corresponding preprocessed images, are shown in Figure 7c,d, respectively.

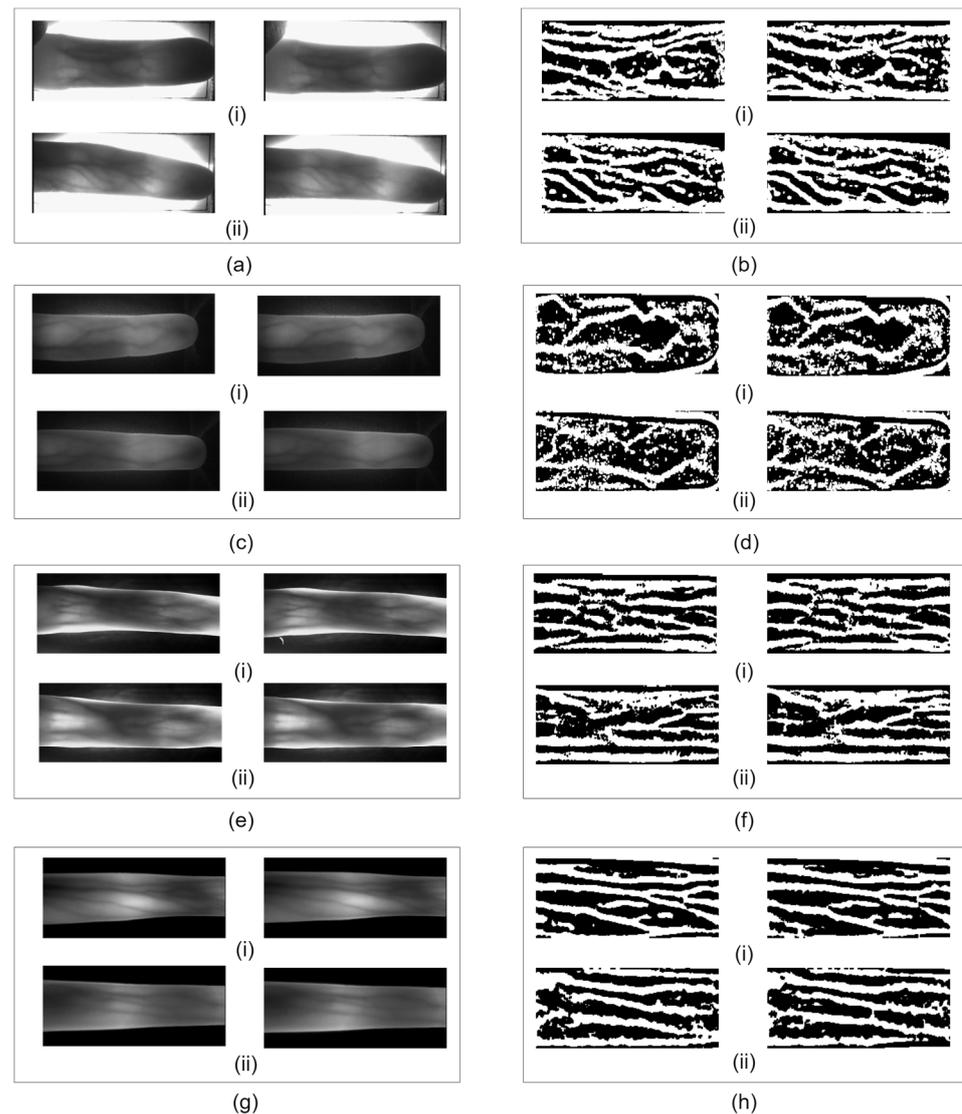


Figure 7. Original images along with their corresponding preprocessed images for (a) original HKPU images; (b) preprocessed HKPU images; (c) original FVUSM images; (d) preprocessed FVUSM images; (e) original UTFVP images; (f) preprocessed UTFVP images; (g) original MMCBNU_6000 images; (h) preprocessed MMCBNU_6000 images. The labels (i) and (ii) denote images selected from two separate classes within each respective dataset.

A total of 1440 finger vascular pattern images are included in the UTFVP dataset. These images were gathered from 60 volunteers at the University of Twente. The collection process involved capturing images in two separate sessions, with an average time gap of 15 days. The vascular patterns of the index, ring, and middle fingers of both hands were collected twice during each session. In each session, two images were collected for

each finger. The captured images had a resolution of 672×380 pixels. In this study, all 1440 images from both sessions resulted in a total of 360 finger classes, with each class having 4 image samples. Figure 7e,f display the original sample images alongside their corresponding preprocessed images, respectively, for the UTFVP dataset.

The MMCBNU_6000 dataset comprises finger vein images that were captured from a group of 100 volunteers originating from 20 different countries. Each participant was requested to provide images of their index finger, middle finger, and ring finger from both hands during the image-capturing process. To ensure an adequate number of samples, each person's collection of six fingers was repeated ten times, resulting in a total of ten finger vein images per individual. Consequently, the finger vein database contains a total of 6000 images. Each image had dimensions of 480×640 pixels. In this study, we used all 6000 images from the MMCBNU_6000 database, resulting in 360 finger classes. Figure 7g,h illustrate the sample original images with their corresponding preprocessed images for MMCBNU_6000.

The quality of finger vein patterns is crucial for obtaining high accuracy from a finger vein recognition system. To analyze the performance of SE-DenseNet-HP against finger vein images of varied quality, the HKPU and FVUSM datasets were further divided into good-quality and poor-quality images based on the image quality of the captured finger vein image and the preprocessed image. Out of 312 classes from the HKPU dataset, 66 were classified as good quality, and the rest were classified as poor quality. Similarly, based on the visual quality of the preprocessed images from the FVUSM dataset, 231 classes were classified as good quality, and the rest were classified as poor quality. As shown in Figure 8, good-quality and poor-quality images from both datasets had different levels of contrast between finger vein patterns and their surroundings. The finger veins in good-quality original images from both datasets were more distinctive from their surroundings compared to poor-quality original images, as shown in Figure 8a,c,e,g. Therefore, the finger vein patterns can be more clearly separated from their background in the preprocessing stage for good-quality original images compared to poor-quality original images, as seen in Figure 8b,d,f,h.

4.2. Training, Validation, and Test Datasets

In all datasets, there are a limited number of images per class, which is insufficient for training the proposed SE-DenseNet-HP model. To increase the number of image samples per class in the training stage, an intraclass data augmentation technique was used. The primary goal of data augmentation is to expand and enrich the examples of finger vein images so that deformations in different vein patterns can be reflected in a real-world scenario. First, four original images per class from HKPU and FVUSM, three original images per class from UTFVP, and five original images per class from MMCBNU_6000 were selected for training, and the rest were separated for testing. Then, we applied 0.01 width and height pixel shifts (left to right and up to down) on the training images. To cross-validate the proposed model performance during the training stage, the training data were further divided into training and validation sets at a ratio of 7:3.

4.2.1. Training Stage

To train the model, the preprocessed finger vein images for both datasets were fed into the SE-DenseNet-HP architecture. A cross-entropy loss function was used to calculate the dissimilarity between model predictions and actual ground truth labels. The loss function was minimized using the Adam optimizer with an initial learning rate of 0.001 to achieve faster convergence. A batch size of 16 and 50 epochs was used during training. Furthermore, early stopping was used with a patience of 10 steps to stop the learning mechanism when there is a saturation in validation loss.

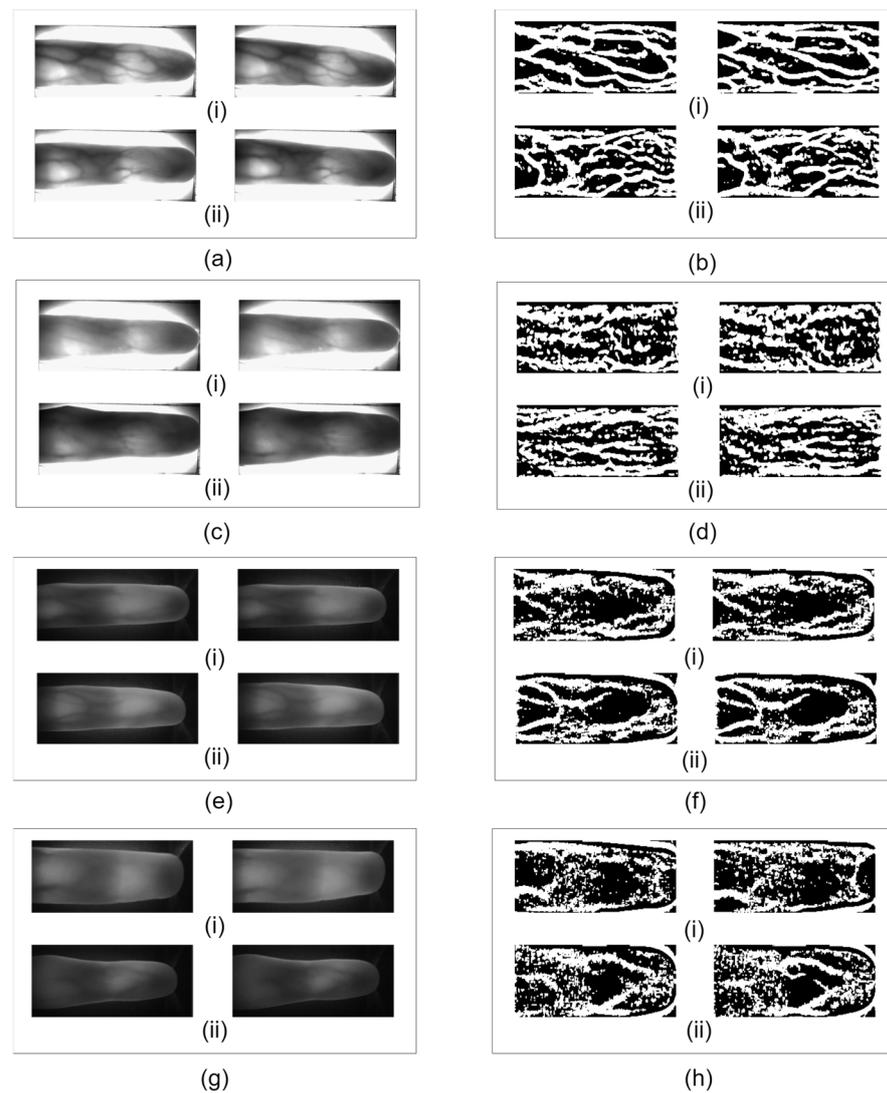


Figure 8. Comparison of good-quality and poor-quality finger vein patterns obtained from the HKPU and FVUSM datasets: (a) good-quality original HKPU images; (b) preprocessing of good-quality original HKPU images; (c) poor-quality original HKPU images; (d) preprocessing of poor-quality original HKPU images; (e) good-quality original FVUSM images; (f) preprocessing of good-quality original FVUSM images; (g) poor-quality original FVUSM images; (h) preprocessing of poor-quality original FVUSM images. The labels (i) and (ii) denote images selected from two separate classes within the good-quality and poor-quality categories of the HKPU and FVUSM datasets, respectively.

4.2.2. Testing Stage

To test the recognition performance of SE-DenseNet-HP, we compared the recognition accuracy, ROC curve, and EER with existing works conducted on the HKPU, FVUSM, UTFVP, and MMCBNU_6000 datasets. And we performed an ablation study, which aims to address the advancements in the use of the DenseNet model in finger vein recognition by incorporating attention mechanisms and custom pooling strategies and assessing their impact on the model's performance. For this, we assessed the recognition accuracy performance of SE-DenseNet-HP in comparison with the original DenseNet and DenseNet-HP, which consist of DenseNet with HP but without SE. Among the various components that can be considered to perform an ablation study, an SE block is considered to have a decisive impact on finger vein recognition performance. The integration of the SE mechanism introduces a channel attention mechanism that selectively highlights features that are relevant to finger vein patterns while suppressing those that are less important. This combination

leads to a more distinct representation of finger vein patterns and contributes to the enhanced recognition performance of our model. In addition, the recognition accuracy of SE-DenseNet-HP was compared with existing works including a DNN [24], FVR-Net [26], P-SVM [44], and NN [45], as shown in Table 2. Recognition accuracy values for P-SVM, NN, DNN and FVR-Net were adapted from [26]. As shown in Table 2, SE-DenseNet-HP has an overall better performance compared to DenseNet-HP and DenseNet, especially for the HKPU dataset. It outperforms DenseNet (86.57%) and DenseNet-HP (92.53%), with a recognition accuracy of 93.28% for high-quality vein images in HKPU. It outperforms DenseNet (80.20%) and DenseNet-HP (84.08%) in recognition accuracy, even in the poor-quality images of HKPU. The obtained results highlight that the integration of SE and the HP mechanism improves the overall performance of the original DenseNet. The channel attention mechanism based on the SE module enhances the feature extraction process of the SE-DenseNet-HP architecture by putting more emphasis on the feature map channels that contain discriminative characteristics of the finger vein patterns. The HP architecture used in the transitional blocks of SE-DenseNet-HP concatenates the average pooling method with the max pooling strategy to preserve both the most discriminative and contextual information.

Table 2. Recognition accuracy comparison of SE-DenseNet-HP with existing methods.

| Method | Datasets | Recognition Accuracy (%) | |
|----------------|----------|--------------------------|--------------|
| | | Good Quality | Poor Quality |
| P-SVM [44] | FVUSM | 66.48 | 65.71 |
| | HKPU | 80.57 | 79.04 |
| NN [45] | FVUSM | 62.35 | 55.00 |
| | HKPU | 81.44 | 81.82 |
| DNN [24] | FVUSM | 69.33 | 68.57 |
| | HKPU | 84.59 | 83.64 |
| FVR-Net [26] | FVUSM | 97.84 | 97.22 |
| | HKPU | 93.18 | 88.97 |
| DenseNet | FVUSM | 99.56 | 98.00 |
| | HKPU | 86.57 | 80.20 |
| DenseNet-HP | FVUSM | 99.56 | 99.23 |
| | HKPU | 92.53 | 84.08 |
| SE-DenseNet-HP | FVUSM | 99.35 | 99.23 |
| | HKPU | 93.28 | 88.16 |

For good-quality vein patterns from the FVUSM dataset, SE-DenseNet-HP obtained 32.87%, 37.00%, 30.02%, and 1.51% higher accuracy than PS-VM, NN, DNN, and FVR-Net, respectively. With the HKPU dataset, SE-DenseNet-HP gained an increment of 12.71%, 11.84%, 8.69%, and 0.10% in recognition accuracy compared with the above-mentioned schemes. For poor-quality vein patterns from the FVUSM dataset, SE-DenseNet-HP outperformed the previously mentioned baseline models and obtained 33.52%, 44.23%, 30.66%, and 2.01% higher recognition accuracy. The higher recognition accuracy obtained by SE-DenseNet-HP is explained by its robust image preprocessing and feature extraction mechanism.

Figure 9 illustrates the comparison of the preprocessed images from the SE-DenseNet-HP and FVR-Net [26]. The original sample images corresponding to three classes in the HKPU dataset are shown in Figure 9a(i,ii,iii). These images illustrate the raw input data prior to any image preprocessing techniques being applied. Figure 9b,c show the finger vein patterns obtained after preprocessing the raw input images corresponding to Figure 9a in SE-DenseNet-HP and FVR-Net, respectively. Similarly, Figure 9d–f show the original sample images for three classes from the FVUSM dataset and the corresponding finger vein patterns obtained after preprocessing in SE-DenseNet-HP and FVR-Net. As shown in Figure 9b,c,e,f, the finger veins patterns obtained after preprocessing in SE-DenseNet-HP

have more detailed information compared to preprocessing in FVR-Net for both the HKPU and FVUSM datasets. This shows that the preprocessing steps in SE-DenseNet-HP are better at extracting the vein patterns compared to the preprocessing method employed in FVR-Net, resulting in enhanced finger vein recognition.

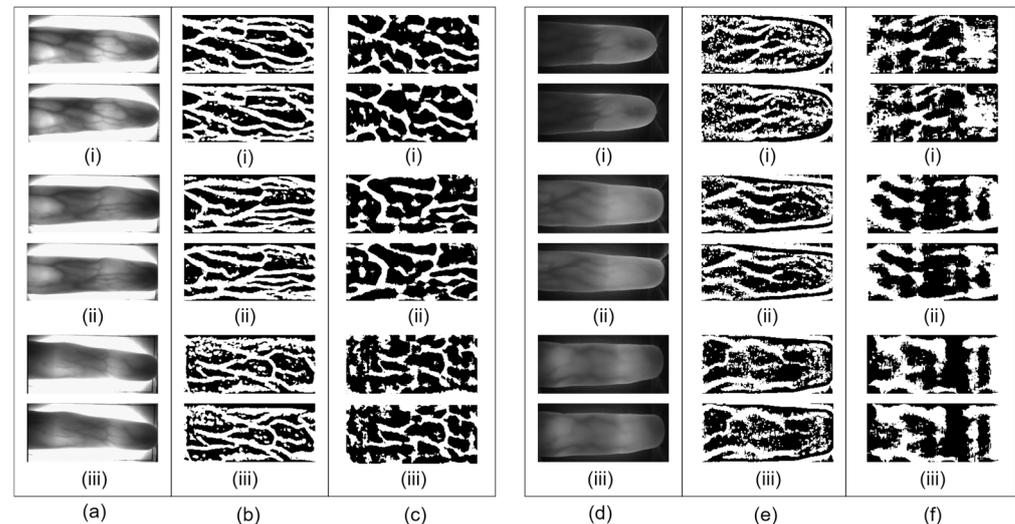


Figure 9. Comparison of preprocessed images from SE-DenseNet-HP and FVR-Net [22]: (a) original sample images from the HKPU dataset; (b) preprocessed images from the HKPU dataset (SE-DenseNet-HP); (c) preprocessed images from the HKPU dataset (FVR-Net); (d) original sample images from the FVUSM dataset; (e) preprocessed images from the FVUSM dataset (SE-DenseNet-HP); (f) preprocessed images from the FVUSM dataset (FVR-Net). The labels (i), (ii) and (iii) indicate images taken from three distinct classes within each respective dataset.

To construct the ROC curve and calculate the EER, the true acceptance rate (TAR), which is $TAR = 1 - \text{the false rejection rate (FRR)}$, is plotted against the false acceptance rate (FAR) at different threshold settings. Note that a decision threshold in a finger vein recognition system is the boundary threshold determining whether a sample finger vein pattern matches the previously stored pattern obtained from a person's finger veins. EER is the threshold value at which the system achieves an equal balance of false positives and false negatives. It denotes the point at which the FRR and FAR are equal. The FRR is the percentage of authorized users that the system rejects, and the FAR is the percentage of imposters that the system accepts. When the decision threshold is lower than the EER, a finger vein recognition system has a low level of security owing to a high FAR. However, when the decision threshold is higher than the EER, the FRR increases as more authorized users are identified as imposters. Therefore, the EER is a trade-off between security and acceptability, which determines the most optimal decision-making threshold in the system. For a finger vein recognition application, lower EER values are preferable because the system is more accurate at identifying the difference between genuine and imposter finger vein patterns. To compare the EER obtained from the proposed SE-DenseNet-HP model with FVR-Net, we used the same image preprocessing technique as above in this study.

Figure 10 illustrates the ROC curve depicting the TAR against the FAR for the FVUSM, HKPU, UTFVP, and MMCBNU_6000 datasets. A curve that is positioned nearer to the upper-left corner of the ROC graph signifies the superior performance of a system. As seen in Figure 10, SE-DenseNet-HP has higher TAR values compared to FVR-Net for FAR values less than 10^{-1} . The above ROC curve suggests that the proposed SE-DenseNet-HP model performed better than FVR-Net for all datasets. This is further justified by the EER obtained with all datasets. For example, the proposed model was able to obtain an EER of 0.03% for FVUSM, which is significantly lower than the EER obtained with FVR-Net (0.68%).

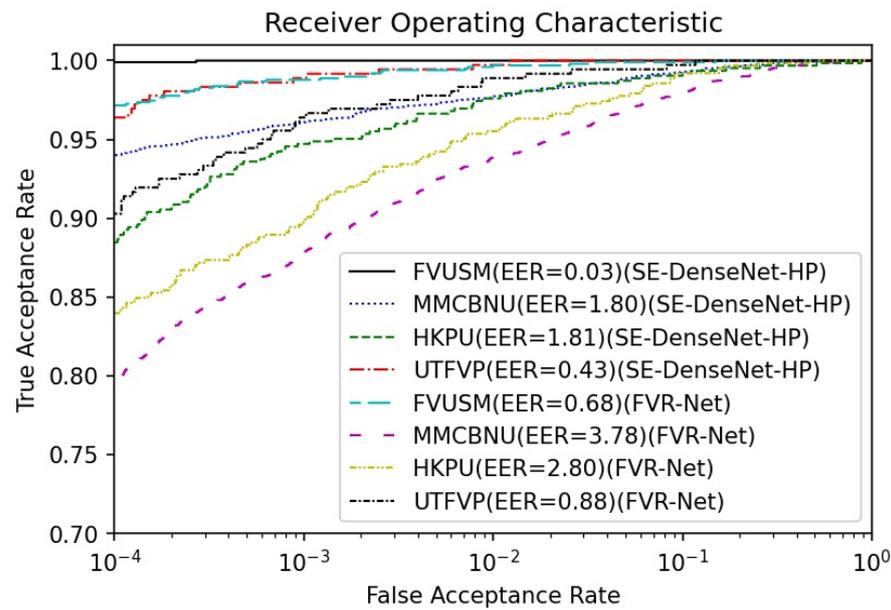


Figure 10. Comparison of EERs (%) obtained from SE-DenseNet-HP and FVR-Net.

To further observe the trade-off between security and acceptability, we compared the EERs with variations of the proposed SE-DenseNet-HP model and existing works. At first, we conducted another ablation study to further enhance the SE-DenseNet-HP architecture by incorporating the group convolution (cardinality) used in ResNext [46] and reducing the skip connections of DenseNet, called Log-DenseNet [47], inside the dense block. The SE-DenseNet-HP with reduced skip connection is named Log-SE-DenseNet-HP. By doing this, we can identify the most effective way to enhance the feature extraction performance of finger veins using DenseNet with an SE mechanism and HP strategy. In the ablation study, the SE-DenseNet-HPs with cardinality 16 and 32 are named SE-DenseNet-HP-16 and SE-DenseNet-HP-32, respectively. Table 3 illustrates the EER comparison of SE-DenseNet-HPs by modifying the cardinality and skip connection process. As illustrated in Table 3, SE-DenseNet-HP-16 and SE-DenseNet-HP-32 achieved a better performance by achieving lower EERs of 0.16% and 0.32% compared to SE-DenseNet-HP (0.43%) in the UTFVP dataset. Conversely, for the HKPU, FVUSM, and MMCBNU_6000 datasets, SE-DenseNet-HP-16 shows an increase in EER values by 0.07%, 0.3%, and 0.37%, while SE-DenseNet-HP-32 demonstrates increases of 0.07%, 0.86%, and 0.54% compared to SE-DenseNet-HP. The above results show that SE-DenseNet-HP has a better performance than SE-DenseNet-HP-16 and SE-DenseNet-HP-32.

Table 3. EER (%) comparison of SE-DenseNet-HP by modifying cardinality and skip connections.

| Method | Cardinality | FVUSM | HKPU | UTFVP | MMCBNU_6000 |
|--|-------------|-------|------|-------|-------------|
| SE-DenseNet-HP (no cardinality) | — | 0.03 | 1.81 | 0.43 | 1.80 |
| SE-DenseNet-HP-16 | 16 | 0.10 | 2.11 | 0.16 | 2.17 |
| SE-DenseNet-HP-32 | 32 | 0.10 | 2.67 | 0.30 | 2.34 |
| Log-SE-DenseNet-HP (no cardinality) | — | 0.13 | 2.26 | 0.42 | 1.95 |
| Log-SE-DenseNet-HP-16 | 16 | 0.06 | 2.59 | 0.59 | 2.29 |
| Log-SE-DenseNet-HP-32 | 32 | 0.07 | 2.56 | 0.44 | 2.34 |

Moreover, SE-DenseNet-HP exhibits superior performance compared to Log-SE-DenseNet-HP, Log-SE-DenseNet-HP-16, and Log-SE-DenseNet-HP-32. In the FVUSM dataset, SE-DenseNet-HP outperformed Log-SE-DenseNet-HP, Log-SE-DenseNet-HP-16, and Log-SE-DenseNet-HP-32, with a lower EER of 0.03% compared to EERs of 0.13%, 0.06%,

and 0.07%. In the HKPU and MMCBNU_6000 datasets, SE-DenseNet-HP exhibits EERs of 1.81% and 1.80%, which is 0.45% and 0.15% lower than Log-SE-DenseNet-HP. In addition, Log-SE-DenseNet-HP-16 and Log-SE-DenseNet-HP-32 also have higher EERs of 2.59% and 2.56% than SE-DenseNet-HP (1.81%) for the HKPU dataset. In the MMCBNU_6000 dataset, SE-DenseNet-HP has a lower EER of 1.80% compared to Log-SE-DenseNet-HP-16 (2.29%) and Log-SE-DenseNet-HP-32 (2.34%). The obtained results show that SE-DenseNet-HP has superior performance compared to Log-DenseNet-HP. SE-DenseNet-HP's dense connections allow each layer to receive direct input from all preceding layers. In finger vein recognition, this might be beneficial for the improved feature extraction of finger vein patterns with different variations in thickness and orientation.

Table 4 presents an EER comparison of SE-DenseNet-HP with FVR-Net, deep segmentation (DS) [4], BACS-LBP [6], Lu et al. [7], and BMSU-LBP [9]. EER values for DS, BACS-LBP, Lu et al. and BMSU-LBP were adapted from [4,6,7,9] respectively. When evaluated on the HKPU dataset, SE-DenseNet-HP surpasses other methods by achieving an EER of 1.81%. Similarly, it achieves a lower EER score in the FVUSM (0.03%), UTFVP (0.43%), and MMCBNU_6000 (1.80%) datasets. In comparison to other existing methodologies, such as deep segmentation (DS) [4], BACS-LBP [6], Lu et al. [7], BMSU-LBP [9], and FVR-Net [24], SE-DenseNet-HP outperforms them by consistently achieving superior EER performance. Note that these methodologies exhibit varying degrees of EERs across different datasets, whereas SE-DenseNet-HP consistently stands out with its lower EER values across all datasets, validating the effectiveness of our proposed finger vein recognition system. The proposed combination of image preprocessing techniques used in this study creates a preprocessing pipeline that can adapt to different environmental conditions specifically for finger vein recognition. By combining the image preprocessing methods and integrating DenseNet, SE block, and HP architecture, our system achieves the enhanced feature representation of finger veins, leading to improved accuracy and generalization performance across various datasets.

Table 4. EER (%) comparison of SE-DenseNet-HP with existing methods.

| Method | FVUSM | HKPU | UTFVP | MMCBNU_6000 |
|----------------------------|-------|------|-------|-------------|
| SE-DenseNet-HP | 0.03 | 1.81 | 0.43 | 1.80 |
| Deep segmentation (DS) [4] | 1.42 | 2.70 | – | – |
| BACS-LBP [6] | 1.16 | 2.86 | – | – |
| Lu et al. [7] | 0.67 | – | 1.18 | 3.33 |
| BMSU-LBP [9] | 1.89 | – | – | 2.60 |
| FVR-Net [26] | 0.68 | 2.80 | 0.88 | 3.78 |

5. Conclusions

In this paper, we proposed a DNN-based network called SE-DenseNet-HP that makes use of DenseNet with HP and an SE-based channel attention mechanism for robust feature extraction and recognition of finger veins. The preprocessed images are then fed into SE-DenseNet-HP for robust feature extraction and recognition. The feature map concatenation in DenseNet improves the information flow between network layers. The SE module added inside the dense blocks uses the channel attention mechanism to emphasize channels containing important finger vein feature information while suppressing other channels that contain less information. HP improves feature extraction performance by acquiring both global and local information from the preprocessed finger vein images. Our experiments show that the SE-DenseNet-HP model can achieve recognition accuracy of up to 99.35% and 93.28% from good-quality vein patterns in the FVUSM and HKPU datasets and outperforms existing work. In addition, it achieves EERs of 0.03%, 1.81%, 0.43%, and 1.80% for the FVUSM, HKPU, UTFVP, and MMCBNU_6000 datasets, respectively, proving its robustness in finger vein recognition compared to previous studies.

Author Contributions: Conceptualization, N.D. and B.W.K.; methodology, B.W.K.; software, N.D.; validation, N.D. and B.W.K.; formal analysis, B.W.K.; investigation, N.D.; resources, N.D.; data curation, B.W.K.; writing—original draft preparation, N.D.; writing—review and editing, N.D. and B.W.K.; visualization, N.D. and B.W.K.; supervision, B.W.K.; project administration, B.W.K.; funding acquisition, B.W.K. All authors have read and agreed to the published version of the manuscript.

Funding: This research was supported by a National Research Foundation of Korea (NRF) grant funded by the Korean government (NRF-2022R1A2B5B01001543), and a project for Industry-Academic Cooperation Based Platform R&D funded by Korea Ministry of SMEs and Startups in 2021 (S3014213).

Data Availability Statement: Dataset available on request from the authors.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Alay, N.; Al-Baity, H.H. Deep learning approach for multimodal biometric recognition system based on fusion of iris, face, and finger vein traits. *Sensors* **2020**, *20*, 5523. [[CrossRef](#)]
2. 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.* **2020**, *69*, 8690–8701. [[CrossRef](#)]
3. Hsia, C.-H.; Ke, L.-Y.; Chen, S.-T. Improved Lightweight Convolutional Neural Network for Finger Vein Recognition System. *Bioengineering* **2023**, *10*, 919. [[CrossRef](#)]
4. Qin, H.; El-Yacoubi, M.A. Deep representation-based feature extraction and recovering for finger-vein verification. *IEEE Trans. Foren. Secur.* **2017**, *12*, 1816–1829. [[CrossRef](#)]
5. Shaheed, K.; Mao, A.; Qureshi, I.; Kumar, M.; Hussain, S.; Zhang, X. Recent advancements in finger vein recognition technology: Methodology, challenges and opportunities. *Inf. Fusion* **2022**, *79*, 84–109. [[CrossRef](#)]
6. Zhang, Z.; Wang, M. A simple and efficient method for finger vein recognition. *Sensors* **2022**, *22*, 2234. [[CrossRef](#)]
7. Lu, H.; Wang, Y.; Gao, R.; Zhao, C.; Li, Y. A Novel ROI Extraction Method Based on the Characteristics of the Original Finger Vein Image. *Sensors* **2021**, *21*, 4402. [[CrossRef](#)]
8. Yao, Q.; Song, D.; Xu, X.; Zou, K. A novel finger vein recognition method based on aggregation of radon-like features. *Sensors* **2021**, *21*, 1885. [[CrossRef](#)] [[PubMed](#)]
9. Hu, N.; Ma, H.; Zhan, T. Finger vein biometric verification using block multi-scale uniform local binary pattern features and block two-directional two-dimension principal component analysis. *Optik* **2020**, *208*, 1–16. [[CrossRef](#)]
10. Kono, M.; Ueki, H.; Umemura, S. Near-infrared finger vein patterns for personal identification. *Appl. Opt.* **2002**, *41*, 7429–7436. [[CrossRef](#)] [[PubMed](#)]
11. Shaheed, K.; Liu, H.; Yang, G.; Qureshi, I.; Gou, J.; Yin, Y. A systematic review of finger vein recognition techniques. *Information* **2018**, *9*, 213. [[CrossRef](#)]
12. 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.* **2023**, *18*, 1322–1334. [[CrossRef](#)]
13. Tao, Z.; Zhou, X.; Xinru, X.; Zhixue, L.; Sen, H.; Yalei, W.; Wei, T. Finger-vein recognition using bidirectional feature extraction and transfer learning. *Math. Probl. Eng.* **2021**, *2021*, 1–11. [[CrossRef](#)]
14. Miura, N.; Nagasaka, A.; Miyatake, T. Feature extraction of finger-vein patterns based on repeated line tracking and its application to personal identification. *Mach. Vision Appl.* **2004**, *15*, 194–203. [[CrossRef](#)]
15. Miura, N.; Nagasaka, A.; Miyatake, T. Extraction of finger-vein patterns using maximum curvature points in image profiles. *IEICE Trans. Inf. Syst.* **2007**, *90*, 1185–1194. [[CrossRef](#)]
16. Kumar, A.; Zhou, Y. Human identification using finger images. *IEEE Trans. Image Process.* **2011**, *21*, 2228–2244. [[CrossRef](#)] [[PubMed](#)]
17. Lee, E.C.; Jung, H.; Kim, D. New finger biometric method using near infrared imaging. *Sensors* **2011**, *11*, 2319–2333. [[CrossRef](#)]
18. Hong, H.G.; Lee, M.B.; Park, K.R. Convolutional neural network-based finger-vein recognition using NIR image sensors. *Sensors* **2017**, *17*, 1297. [[CrossRef](#)]
19. He, C.; Li, Z.; Chen, L.; Peng, J. Identification of finger vein using neural network recognition research based on PCA. In Proceedings of the IEEE 16th International Conference on Cognitive Informatics & Cognitive Computing (ICCI* CC 2017), Oxford, UK, 26–28 July 2017.
20. Khellat-Kihel, S.; Cardoso, N.; Monteiro, J.; Benyettou, M. Finger vein recognition using Gabor filter and Support Vector Machine. In Proceedings of the International Image Processing, Applications and Systems Conference, Sfax, Tunisia, 3–5 November 2014.
21. Mobarakeh, A.K.; Rizi, S.M.; Khaniabadi, S.M.; Bagheri, M.A.; Nazari, S. Applying Weighted K-nearest centroid neighbor as classifier to improve the finger vein recognition performance. In Proceedings of the 2012 IEEE International Conference on Control System, Computing and Engineering, Penang, Malaysia, 23–25 November 2012.
22. Kumar, R.P.; Agrawal, R.; Sharma, S.; Dutta, M.K.; Travieso, C.M.; Alonso, J.B. Finger vein recognition using integrated responses of texture features. In Proceedings of the 4th International Work Conference on Bioinspired Intelligence (IWOB 2015), San Sebastian, Spain, 10–12 June 2015.

23. Yang, W.; Hui, C.; Chen, Z.; Xue, J.H.; Liao, Q. FV-GAN: Finger vein representation using generative adversarial networks. *IEEE Trans. Inf. Forensics Secur.* **2019**, *14*, 2512–2524. [[CrossRef](#)]
24. Qin, H.; El-Yacoubi, M.A. Deep representation for finger-vein image-quality assessment. *IEEE Trans. Circuits Syst. Video Technol.* **2017**, *28*, 1677–1693. [[CrossRef](#)]
25. Das, R.; Piciuccio, E.; Maiorana, E.; Campisi, P. Convolutional neural network for finger-vein-based biometric identification. *IEEE Trans. Inf. Forensics Secur.* **2018**, *14*, 360–373. [[CrossRef](#)]
26. Tamang, L.D.; Kim, B.W. FVR-Net: Finger Vein Recognition with Convolutional Neural Network Using Hybrid Pooling. *Appl. Sci.* **2022**, *12*, 7538. [[CrossRef](#)]
27. Sidiropoulos, G.K.; Kiratsa, P.; Chatzipetrou, P.; Papakostas, G.A. Feature Extraction for Finger-Vein-Based Identity Recognition. *J. Imaging* **2021**, *7*, 89. [[CrossRef](#)]
28. Li, H.; Yang, L.; Yang, G.; Yin, Y. Discriminative binary descriptor for finger vein recognition. *IEEE Access.* **2017**, *6*, 5795–5804. [[CrossRef](#)]
29. Zhang, R.; Yin, Y.; Deng, W.; Li, C.; Zhang, J. Deep learning for finger vein recognition: A brief survey of recent trend. *arXiv* **2022**, arXiv:2207.02148.
30. Huang, G.; Liu, Z.; Van Der Maaten, L.; Weinberger, K.Q. Densely connected convolutional network. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017.
31. Hu, J.; Shen, L.; Sun, G. Squeeze-and-excitation networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake, UT, USA, 18–23 June 2018.
32. Asaari, M.S.M.; Suandi, S.A.; Rosdi, B.A. Fusion of band limited phase only correlation and width centroid contour distance for finger based biometrics. *Expert Syst. Appl.* **2014**, *41*, 3367–3382. [[CrossRef](#)]
33. Ton, B.T.; Veldhuis, R.N. A high quality finger vascular pattern dataset collected using a custom designed capturing device. In Proceedings of the 6th IAPR International Conference on Biometrics (ICB 2013), Madrid, Spain, 4–7 June 2013.
34. Lu, Y.; Xie, S.J.; Yoon, S.; Wang, Z.; Park, D.S. An available database for the research of finger vein recognition. In Proceedings of the 6th International Congress on Image and Signal Processing (CISP 2013), Hangzhou, China, 16–18 December 2013.
35. Yin, Y.; Zhang, R.; Liu, P.; Deng, W.; He, S.; Li, C.; Zhang, J. Artificial Neural Networks for Finger Vein Recognition: A Survey. *arXiv* **2022**, arXiv:2208.13341.
36. Yao, Q.; Song, D.; Xu, X. Robust finger-vein ROI localization based on the 3σ criterion dynamic threshold strategy. *Sensors* **2020**, *20*, 3997. [[CrossRef](#)] [[PubMed](#)]
37. Reza, A.M. Realization of the contrast limited adaptive histogram equalization (CLAHE) for real-time image enhancement. *J. VLSI Signal Process. Syst. Signal Image Video Technol.* **2004**, *38*, 35–44. [[CrossRef](#)]
38. Justusson, B. Median filtering: Statistical properties. In *Two-Dimensional Digital Signal Processing II: Transforms and Median Filters*; Huang, T.S., Ed.; Springer: Berlin/Heidelberg, Germany, 2006; pp. 161–196.
39. Zhu, R.; Wang, Y. Application of Improved Median Filter on Image Processing. *J. Comput.* **2012**, *7*, 838–841. [[CrossRef](#)]
40. Bradley, D.; Roth, G. Adaptive thresholding using the integral image. *J. Graphics Tools* **2007**, *12*, 13–21. [[CrossRef](#)]
41. Roy, P.; Dutta, S.; Dey, N.; Dey, G.; Chakraborty, S.; Ray, R. Adaptive thresholding: A comparative study. In Proceedings of the 2014 International Conference on Control, Instrumentation, Communication and Computational Technologies (ICCCCT), Kanyakumari, India, 10–11 July 2014.
42. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep Residual Learning for Image Recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016.
43. Tong, Z.; Tanak, G. Hybrid pooling for enhancement of generalization ability in deep convolutional neural networks. *Neurocomputing* **2019**, *333*, 76–85. [[CrossRef](#)]
44. Platt, J.C. Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods. *Adv. Large Margin Classif.* **1999**, *10*, 61–74.
45. Hopfield, J.J. Neural networks and physical systems with emergent collective computational abilities. *Proc. Natl. Acad. Sci. USA* **1982**, *79*, 2554–2558. [[CrossRef](#)]
46. Xie, S.; Girshick, R.; Dollár, P.; Tu, Z.; He, K. Aggregated residual transformations for deep neural networks. In Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017.
47. Hu, H.; Dey, D.; Del, G.A.; Hebert, M.; Bagnell, J. Log-DenseNet: How to Sparsify a DenseNet. *arXiv* **2017**, arXiv:1711.00002.

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.