Received December 9, 2020, accepted December 16, 2020, date of publication December 22, 2020,
date of current version January 4, 2021.

Digital Object Identifier 10.1109/ACCESS.2020.3046108

Real-Time Segmentation Method of Lightweight Network For Finger Vein Using Embedded Terminal Technique

JUNYING ZENG[1], (Member, IEEE), BOYUAN ZHU[1], YUJIE HUANG[1], CHUANBO QIN[1], JINGMING ZHU[1], FAN WANG[3], YIKUI ZHAI[1,2], (Member, IEEE), JUNYING GAN[1], (Member, IEEE), YUCONG CHEN[1], YINGBO WANG[1], RUGGERO DONIDA LABATI[2], (Member, IEEE), VINCENZO PIURI[2], (Fellow, IEEE), AND FABIO SCOTTI[2], (Senior Member, IEEE)

[1]Department of Intelligent Manufacturing, Wuyi University, Jiangmen 529020, China
[2]Departimento di Information, Universita degli Studi di Milano, 20133 Milano, Italy
[3]Cunjin College, Guangdong Ocean University, Zhanjiang 524000, China

Corresponding author: Chuanbo Qin (tenround@163.com)

This work was supported in part by NNSF under Grant 61771347, in part by the Special Project in key Areas of Artificial Intelligence in Guangdong Universities under Grant 2019KZDZX1017, in part by the Guangdong Basic and Applied Basic Research Foundation under Grant 2019A1515010716, in part by the Basic Research and Applied Basic Research Key Project in General Colleges and Universities of Guangdong Province under Grant 2018KZDXM073, and in part by the Open fund of Guangdong Key Laboratory of digital signal and image processing technology under Grant 2019GDDSIPL-03 and Grant 2020GDDSIPL-03.

ABSTRACT Because the existing finger vein segmentation networks are too large and not suitable for implementation in mobile terminals, the reduction of the parameters of the lightweight network leads to the reduction of the segmentation index, and the long-running time of deep network on hardware platforms; this paper proposes a lightweight real-time segmentation method for finger veins based on embedded terminal technique. In the preprocessing stage of the algorithm, the data is greatly expanded by randomly selecting the center to obtain sub-blocks on each image of the training set. The network first uses deep separable convolution to greatly reduce the U-Net parameters of a basic network and introduces an attention module to reorder the features to improve network performance, followed by a preliminary lightweight network Dinty-NetV1. Second, the Ghost module is added to the deep separable convolution, and the feature map of the network part is obtained through a cheap operation so that the network is further compressed to obtain Dinty-NetV2. After adding channel shuffle, all the characteristic channels are evenly shuffled and reorganized to obtain Dinty-NetV3. Finally, a study of the filter norm yields the distribution characteristics of the finger vein picture features. By using the geometric median pruning method, the network models for each stage of the algorithm proposed in this paper achieved better segmentation performance and shorter split time after pruning. The overall Dinty-NetV3 model size is only less than 9% of the U-Net and Mult-Adds is less than 2% of the U-Net with the same structure. After testing on two-finger vein datasets SDU-FV and MMCUBV-6000, we confirm that the performance of Dinty-NetV3 surpasses all previously proposed classic compression model algorithms and it is not inferior to more complex and huge networks such as U-Net, DU-Net, and R2U-Net. The proposed algorithm has advantages in terms of time needed to train the network, and we verify its universality using NVIDIA's full range of embedded terminals.

INDEX TERMS Channel shuffle, depth separable convolution, Dinty-Net, embedded terminal, filter pruning via geometric median, finger vein segmentation, GhostNet, lightweight network, U-Net.

I. INTRODUCTION

In recent years, biometric technology has been gaining more and more attention from the general public owing to its increasing need to in ensuring the safety and accuracy of biometric systems. At present, various biological characteristics such as fingerprints [1], palm-print [2], finger-vein [3], [4], hand-vein [5], palm-vein [6], face [7], iris [8], voice [9], gait [10], and signature [11] have been employed in recognition and verification applications. Finger vein recognition

The associate editor coordinating the review of this manuscript and approving it for publication was Hiu Yung Wong.

has been gaining tremendous attention in the research community due to its advantages of non-contact collection, live detection, not easy to forge, and low cost compared to other biometric recognition technologies [12]. Because the quality of the segmentation effect will directly affect the accuracy of subsequent finger vein feature extraction and recognition, segmentation of finger vein images has become a key step in finger vein recognition technology. The quality of the segmentation effect will directly affect the accuracy of subsequent finger vein feature extraction and recognition scheme. Although previously proposed segmentation networks for finger veins such as FCN [13], SegNet [14], RefineNet [15], U-Net [16], etc., have achieved better performance, it is worthy of note that these methods require a huge amount of storage resources in the device's space. Further, huge computing resource requirements make it difficult to effectively apply these methods in many current mobile hardware platforms. At present, domestic and foreign researchers have done a lot of research on finger vein segmentation and recognition and even on various network lightweight methods, however there are only a few embedded terminal implementation analysis algorithms.

Nowadays, artificial intelligence (AI), machine learning (ML) and deep learning (DL) all play important roles in many applications. In the ImageNet challenge [17], computers have surpassed the ability of humans to classify images. Since the invention of AlexNet (convolutional neural network) in 2012, computer vision has entered a new era. Deep neural networks are the most advanced technology used in applications including computer vision, natural language processing, speech recognition, etc. They enable computers to perform tasks that were once considered impossible. AI is becoming increasingly popular. The new industrial revolution dedicated to contributing to smart cities is called Industry 4.0, smart medical systems, agriculture, and education. The list continues. Internet connection has become an important feature of every household device. Internet-connected devices are growing explosively, and there are currently 20 million IoT devices in the world [18].

The problem of finger vein segmentation in embedded terminals is similar to the multi-objective optimization problem and segmentation performance. Further, network size and running time also need to be considered. Most of the previous lightweight research goals are to achieve relatively good accuracy with a network with small parameters on a computer terminal and considers only segmentation performance optimization and network size. However, in deep networks, it is necessary to reduce the number of parameters. The lightweight method often introduces a large number of operations that are more complicated and time-consuming than convolution. Although the results of this operation are reduced on the PC side and the accuracy is not reduced or even improved. To perform the above simple operations in an embedded terminal, longer calculation time is often needed, which is contrary to the original purpose of the lightweight method. Han published deep compression and EIE [19] in 2015. This article is a review-type article on model compression methods and successfully examined cutting, quantifying, and techniques for sharing weights which are used in model compression. It achieved quite satisfactory results and ultimately was the best paper of ICLR2016. Afterward, a research on model compression methods showed that DL mainly has the following branches: (i) Refined design for the model. At present, a large number of networks are very large in depth and width, resulting in a large number of redundant parameters. Further, there are many studies on model design, such as MobileNet [20] and Inception [21] that used more detailed and efficient model design techniques to achieve better performance results while reducing the number of model parameters. Although these models can achieve excellent performance with few parameters, they have never fully exploited the correlation and redundancy between feature maps. (ii) Tailoring for smaller neurons. Algorithms with complex structures and large neural networks usually have better performance. As result, there will be more redundancy in neurons, thus, a standard can be used to limit the unimportant connections or filters of the trained network. It is tailored to reduce the redundancy of the model, as reported in [22]–[24].

Since U-Net was proposed, due to its excellent performance in medical image segmentation, it is often used as infrastructure in other networks such as R2U-Net [25], DU-Net [26], etc. Most of these networks greatly increase the size and parameters of the model in pursuit of higher segmentation performance. Although this article chooses U-Net as the infrastructure network, the purpose is to compress the network. In the refined design of compression, the deep separable convolution in MobileNet was first used to reduce the parameters of the network initially, however it did not reach the ideal goal of this article. GhostNet [27] is also a kind of compact model design. It proposes that a feature map can be used as a ''Ghost'' of another and the ''Ghost'' can be generated by a cheaper operation. Using this algorithm can reduce the total number of parameters and computational complexity of the network that has been compressed earlier. This article uses the ''GhostNet'' idea to further compress the network. The hypothesis of the depth separable structure in MobileNet is that correlation and spatial correlation between the channels of the convolutional layer can be decoupled by ignoring the correlation and spatial correlation between them; however, it is not applicable in finger vein segmentation. This article uses ShuffleNet [28] proposed to use channel shuffle to reorganize the feature map after group convolution, so that the information can be transferred between different groups.

Generally speaking, a filter with a smaller norm is expected to make an absolutely small contribution to the network, rather than a relatively small but positive contribution, however this is not the case. After checking the features extracted by the network for redundancy, it is found that the conventional clipping method for smaller neurons can only speed up the network, but Filter Pruning via Geometric Median (FPGM) [29] proposed a filter pruning method selects the

filter with the greatest replaceability for cutting, which not only accelerates the network but also eliminates harmful features of the network results to improve the segmentation performance of the network. It is also necessary while compressing the model to consider that the segmentation performance of the network itself cannot be reduced due to a large reduction in parameters. Squeeze-and-Excitation Network [30] proposes attention and gating mechanisms (SE). The module can reorder the convolutional information, model the relationship in the feature map channel in an efficient calculation method, and is designed in the network to enhance the expressive ability of the network module. This module can improve the segmentation performance of the network at alow calculational cost.

Inspired by MobileNet, GhostNet, ShuffleNet, FPGM, Squeeze-and-Excitation Network and U-Net, we refine a U-Net-based lightweight fusion segmentation algorithm for finger veins.

The main contributions of this paper are as follows:

(i)First, this article aims at the problem of the traditional network being too large and requiring too many parameters, thus we employ deep separable convolution to lighten the network. Aiming at the feature map redundancy problem found in network visualization, GhostNet mapping is used to further compress the network and add the algorithm ''SE'' module that can improve the accuracy is improved, and the performance of finger vein segmentation reduced due to parameter reduction is improved.

(ii)Breaking the depth separable convolution is the hypothesis that the correlation and spatial correlation between the channels of the convolutional layer can be decoupled, and channel shuffle is added to the depth separable convolution with Ghost to group all feature channels and disrupt and reorganize evenly, the network is further compressed, and the performance is further improved.

(iii)Because of the excessively concentrated characteristics of the norm distribution of the finger vein filter, FPGM not only enables the network to be further accelerated on the test platform, but also the network models proposed at each stage achieved a better performance after pruning.

The rest of the paper is organized as follows: Section 2 introduces related work, Section 3 describes our method in detail, Section 4 gives experimental results and analysis, and Section 5 gives conclusions and future work.

II. RELATED WORK

The GhostNet[27] idea comes from Huawei's Noah's Ark Laboratory. It proposes that a well-trained deep neural network usually contains rich or even redundant feature maps, and one feature map can be transformed from another feature map through certain operations. It has been reported that once obtained, it can be assumed that one of the feature maps is a ''phantom''of the other. Therefore, GhostNet proposed that not all feature maps need to be obtained by convolution operation, and ''phantom'' feature maps can be generated by cheaper operations. Praticularly, the ordinary convolutional layer in the deep neural network will be divided into two parts. The first part involves ordinary convolutions, but the total number of them will be strictly controlled. Given the inherent feature map of the first part, a series of simple linear operations are applied to generate more feature maps. Without changing the size of the output feature map, the total number of parameters and the computational complexity required in the Ghost module have been reduced when compared with the ordinary convolutional neural network. The article shows that the Ghost module can reduce the computational cost of the general convolutional layer while maintaining similar recognition performance, and GhostNet can surpass advanced and efficient deep models such as MobileNetV3 to perform rapid inference on mobile devices.

ShuffleNet proposes the depth separable convolution uses an independent feature channel extraction to make a $1*1$ convolution in MobileNet, which is equivalent to the limit of the Inception structure. The hypothesis of this refined operation is that the correlation and spatial correlation between the convolutional layer channels can be decoupled. Map them separately and use channel shuffle technique instead of pointwise convolution to reorganize the feature map after group convolution so that the information can be transferred between different groups.

FPGM believes that the previous pruning work uses the ''smaller norm less important'' criterion to prune filters with smaller norms in convolutional neural networks, pointing out that its effectiveness depends on two requirements that are not always met: (i) The specification deviation of the filter should be large; (ii) The minimum specification of the filter should be small. There are two prerequisites for using this ''smaller, less standardized importance'' standard. First, the deviation of the filter specification should be large enough. This requirement makes the search space of the threshold T wide enough so that separating those filters that need to be pruned will be an easy task. Second, the specifications of the filters that can be pruned should be arbitrarily small, that is, close to zero; in other words, filters with smaller norms are expected to make an absolute small contribution to the network, rather than a relatively small but positive contribution, and it is not the case. Therefore, a novel filter pruning method is proposed, called ''filter pruning via geometric median (FPGM)''. FPGM selects the filter with the greatest replaceability unlike the previous method, which trims the filter with less contribution.

III. PROPOSED METHOD

A. DATA EXPANSION PART

The experiments in this paper were performed using the public datasets of SDU-FV [31] and MMCBNU_6000 [32]. One of the biggest characteristics of the data set of finger vein images is that the number of classifications is large or the training period is unknown, and the number of training samples for a single classification is very small. In order to increase the number of training samples and make full use of the limited data, at this stage, we use a block for

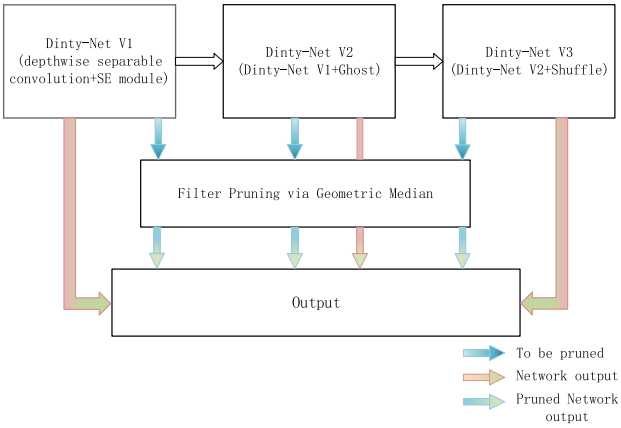


FIGURE 1. Algorithm improvement process.

data expansion. It worthy of note that the basic network U-Net used in this article has been down-sampled four times, so the block size is 64*64. According to this method, 100 training pictures of data can be divided into 200,000 pictures.

B. THE OVERALL IDEA OF THE ALGORITHM

The problem of finger vein segmentation in the embedded terminal is similar to a multi-objective optimization problem, the segmentation performance, network size and the running time of the embedded terminal with different computing performance should be considered at the same time. Therefore, the research in this paper cannot just add various modules to the network to improve accuracy like traditional research methods. The proposed algorithm flow is shown in Figure 1, and the overall network architecture of this paper is shown in Figure 2.

The first step is to build a U-Net based lightweight finger vein fusion segmentation network Dinty-NetV1 through MobileNetV2[33] according to the traditional lightweight

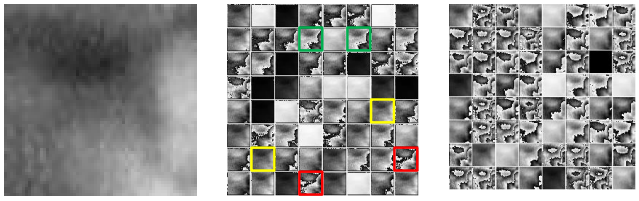


FIGURE 3. Database training original image and 64 feature maps generated by Dinty-NetV1 and V2.

network research process. Then integrate "SE" module to improve the accuracy of the model, but there are many redundant feature maps in the current network.

Aiming at the redundancy problem of feature maps, GhostNet and deep separable convolution are fused, so that part of the feature maps in the network get Dinty-NetV2 through GhostNet's "cheap operation".

Figure 3 shows the visual observation of the feature map output by the middle layer of the Dinty-NetV1 network. It can be seen that compared with the original image, when 64 feature maps are output for the first convolution, not all feature channels have the characteristics of the original image. The extracted is relatively complete. A considerable part of the feature maps cannot clearly see the characteristics of the veins, and there are many pairs of feature maps with similar features. They do not need to be generated by depth separable convolution but can be obtained by linear transformation with less calculation. Therefore, the mapping idea proposed in Ghost is used to share part of the calculation of generating feature maps to further compress and accelerate the network. Select half of the feature map of each layer of convolution as the mapped feature map. The same generated feature map is shown in Figure 3(c). You can see that the Ghost's feature map has more items to extract than Dinty-NetV1 (Figure 3(b)). Feature map of deep features.

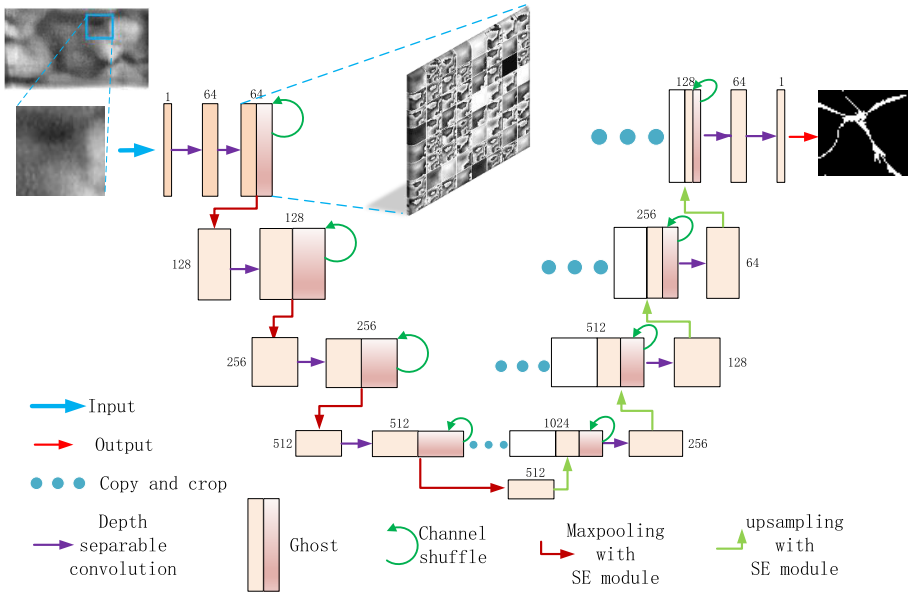


FIGURE 2. Network architecture of the proposed method.

TABLE 1. Deep separable convolution structure.

Input	Operator	Output
$h \times w \times k$	1×1 conv2d, ReLU6	$h \times w \times (tk)$ expand
$h \times w \times tk$	3×3 dwise s=s, ReLU6	$\frac{h}{s} \times \frac{w}{s} \times (tk)$ conv
$\frac{h}{s} \times \frac{w}{s} \times tk$	Linear 1×1 conv2d	$\frac{h}{s} \times \frac{w}{s} \times k'$ compress

Dinty-NetV3 breaks the assumption that the correlation and spatial correlation between channels in the convolutional layer can be decoupled based on Dinty-NetV1. Channel shuffle is added in the convolution process to evenly shuffle and reorganize all feature channels to make their characteristics. The channel information and spatial information communicate with each other to complete the final lightweight model in the model structure design.

Additionally, this article also proposes cutting out some of the feature maps that are useless or even unfavorable for the subsequent extraction of features for ensuring the redundancy of feature maps. The pruning of the model is actually a very direct method to accelerate the model. The pruning of the filter for the redundancy of the feature map is not only accelerated, but the network performance of each stage is further improved after the pruning.

C. A LIGHTWEIGHT SEGMENTATION ALGORITHM BASED ON U-NET–DINTY-NETV1

Dinty-NetV1 is composed of three steps as a whole. The first step is to follow the U-Net architecture as the basic network. U-Net is mainly composed of a contracted path and an expanded path. The contracted path is used to capture the contextual information in the picture while the expanded path is used to accurately locate the part that is needed to be segmented in the picture. The down-sampling process in the left half of the structure can gradually reveal environmental information while the up-sampling process in the corresponding right structure can be combined with the down-sampled input information to gradually restore the image accuracy, convolution and convolution in the network after adding ReLU and normalization.

The second step is the lightweight processing of U-Net. This is done by replacing the ordinary convolution layer in U-Net with the depth separable convolution proposed by MobileNetV2. The depth separable convolution structure of MobileNetV2 is shown in Table 1. Where t represents the ''expansion'' multiple, k represents the number of output channels, k' represents the number of repetitions, and s represents the step size. Among them, operation 1 and operation 3 are called point convolution, and operation 2 is called depth convolution. The spatial structure is shown in Figure 4.

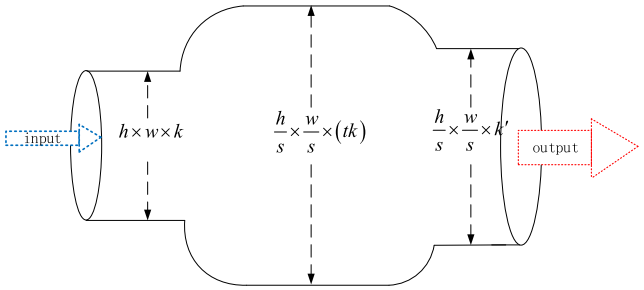


FIGURE 4. Depth separable convolutional space structure diagram.

For a feature image of size $D_G \times D_G \times M$ input to the neural network, the output feature image is $D_F \times D_F \times M$,the size of the convolution kernel $D_K \times D_K \times M$. The amount of calculation for the standard convolution operation is

$$D_K \times D_K \times M \times N \tag{1}$$

For depth separable convolution, the amount of calculation for point convolution is

$$1 \times 1 \times M \times N \tag{2}$$

Depth convolution:

$$D_K \times D_K \times M \times 1 \tag{3}$$

The overall amount of calculation is

$$D_K \times D_K \times M \times 1+1 \times 1 \times M \times N \tag{4}$$

The depth separable convolution is compared with the standard convolution parameters. Assuming that the input and output are feature images of the same size, we get:

$$\frac{1}{N} + \frac{1}{D_K^2} \tag{5}$$

Further, in a situation where the value of N is generally larger, the size of the previous item can be ignored. Assuming that the size of the convolution kernel we set is 3*3, only in this convolution process compared to the standard convolution, the parameter amount can be reduced to one-ninth of the original. Taking this as an example, U-Net achieves model reduction through depth separable convolution.

The attention mechanism is introduced after the deep separable convolution. Further, we use the ''SE'' module proposed by the Squeeze-and-Excitation Network[30]. The ''SE'' module can learn the correlation between channels and improve the accuracy of the model by sacrificing a small amount of calculation. Figure 5 shows the structure diagram of the ''SE'' module. First, perform the overall average pooling of C ×H ×W to obtain a feature map of 1 × 1×C size. This feature map can be understood as having a global receptive field, followed by two full connection layer structure diagram to model the correlation between the roads, then obtain the normalized weight between 0 and 1 through a sigmoid, and finally use a Scale operation to weight the normalized weight to the features of each channel to complete the original feature in the channel dimension. Calibration makes the network

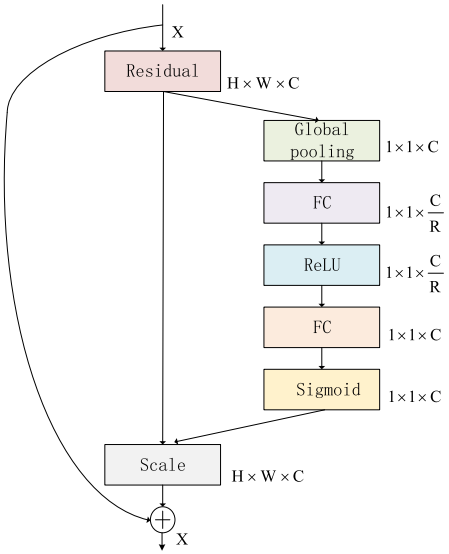


FIGURE 5. "SE" module structure.

have the ability of targeted learning. Overall, the integration of the above several steps algorithms completes the core construction of the network.

The total parameters of the built network are 5.36M, the model calculation cost Mult-Adds is 0.17G, and the U-Net training parameter of the same structure and the number of channels is 13.39M, which is 2.5 times that of the model in this article, Mult-Adds It is 1.93G, which is 11.4 times that of the current model, so this article refers to the current network as Dinty-NetV1.

D. GHOST IS INTEGRATED INTO DEEP SEPARABLE CONVOLUTION–DINTY-NETV2

GhostNet follows the assumption that a well-trained deep neural network usually contains rich or even redundant feature maps to ensure a comprehensive understanding of the input data. The feature map extracted by the network can be obtained by transforming another feature map through a cheap operation. It can be considered that one of the feature maps is the ''Ghost'' of the other, and not all feature maps need to be obtained by a convolution operation as shown in figure6(a). We used deep separable convolution to complete the first part of the feature map generation, then merge the GhostNet module and use this part of the feature map to generate their corresponding ''Ghost'' as shown in figure6(b).

The profit analysis of Dinty-Net fusion Ghost module memory usage and theoretical acceleration is as follows:

For a feature image of size $D_G \times D_G \times M$ input to the neural network, the output feature image is $D_F \times D_F \times M$, the size of the linear mapping is $d \times d$. The calculation is divided into two parts:

first part is

$$\frac{D_K \times D_K \times M \times N}{R} \tag{6}$$

second part is

$$\frac{d \times d \times N \times (R-1)}{R} \tag{7}$$

the convolution kernel $D_K \times D_K \times M$, Ghost mapping ratio is R, that is, the N/R output feature map is obtained by convolution, the number of feature maps in the linear mapping part is $N \times (R\text{-}1)/R$ part is $N \times (R\text{-}1)/R$, the average kernel

Total calculation is

$$\frac{D_K \times D_K \times M \times N}{R} + \frac{d \times d \times N \times (R-1)}{R} \tag{8}$$

The compression ratio can be obtained by dividing standard convolution operation $D_K \times D_K \times M \times N$

$$\begin{aligned} R_C &= \frac{D_K \times D_K \times M \times N}{\frac{D_K \times D_K \times M \times N}{R} + \frac{d \times d \times N \times (R\text{-}1)}{R}} \\ &= \frac{R \times M}{R+M\text{-}1} \approx R \end{aligned} \tag{9}$$

Since the average kernel of the mapping $d \times d$ is similar to the size of $D_K \times D_K$, so the compression ratio R_C is approximately equal to the mapping ratio R. If the number of feature maps mapped in the output image is half of N, that is, R=2, the network calculation amount is reduced to half of the previous one.

Integrate Ghost into Dinty-NetV1 and the improved module performs accelerated revenue analysis as follows:

$$D_K \times D_K \times M \times 1{+}1 \times 1 \times M \times \frac{N}{R} \tag{10}$$

The second part of the calculation is still the formula (7)

At this time compression ratio is

$$\begin{aligned} R'_c &= \frac{D_K \times D_K \times M \times N}{D_K \times D_K \times M \times 1{+}1 \times 1 \times M \times \frac{N}{R} + \frac{d \times d \times N \times (R\text{-}1)}{R}} \\ &\approx \frac{R \times M}{R + \frac{M}{D_K^2}} \end{aligned} \tag{11}$$

If R=2 and $D_K \times D_K = 9$, the theoretical compression ratio is $R'_c = \frac{18M}{M+18}$ when the number of input channels M is 1, it is the first feature extraction. We believe that the network is shallow at this time and the number of channels is not too redundant, so the source network is still almost uncompressed; when the number of input channels is 64, this the theoretical compression ratio is 14, and the compression ratio will increase as the network deepens.

E. GROUP CONVOLUTIONAL SHUFFLES–DINTY-NETV3

Depth separable convolution is to use an independent feature channel extraction to make a 1*1 convolution, which is equivalent to the limit of the Inception structure. The fundamental hypothesis behind Inception is that cross-channel correlations and spatial correlations are sufficiently decoupled that it is preferable not to map them jointly. Sometimes better results may be achieved, but the finger vein image features studied in this article may violate this assumption. Therefore, this article adds channel shuffle to Dinty-NetV2 to conduct new experiments and propose Dinty-NetV3. After channel shuffle undergoes the same deep convolution, it does not separate the spatial features from the convolution channel like the depth separable convolution, but groups all the feature channels

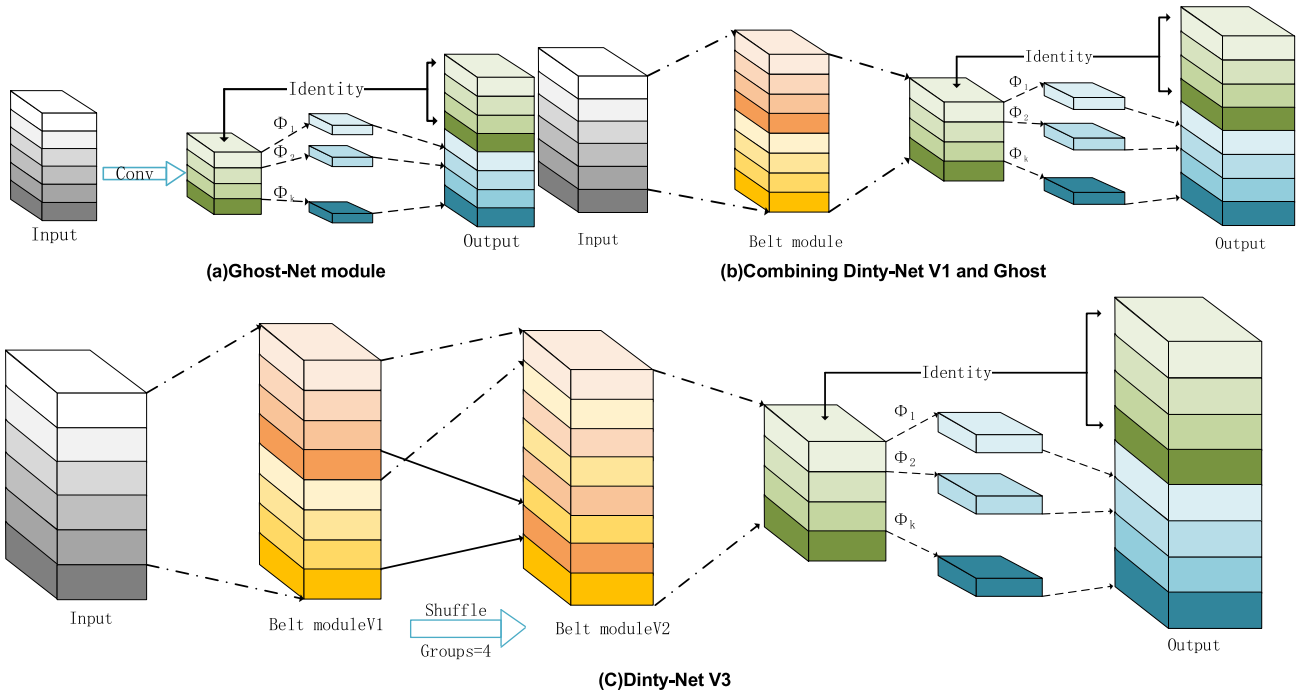


FIGURE 6. Ghost, Dinty-Net V2 and Dinty-Net V3 convolutional process.

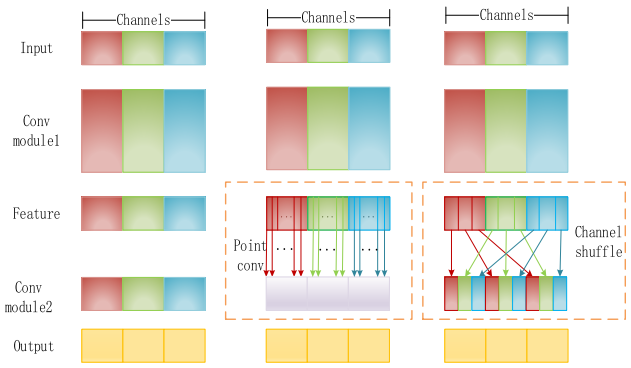


FIGURE 7. The operation process of ordinary convolution, depth separable volume and channel shuffle.

and "evenly shuffles" them to regroup their feature channels. Figure 7 shows the exchange of information as well as spatial information and convolutional process is shown in figure6(c). Compared with the previous Dinty-NetV2, the number of 1∗1 convolutions will be greatly reduced, thus reducing the params and multy-adds of the network. The size of the Dinty-NetV3 model is only less than a percentage of the same structure U-Net Ninth, Mult-Adds is less than 2% of the U-Net with the same structure, and after experimentation, it is found that the segmentation performance of Dinty-NetV3 is improved again and the time is reduced again compared with Dinty-NetV2.

F. FPGM FOR THE CHARACTERISTICS OF FINGER VEIN FILTERS

The usual filter pruning method following the "smaller standard filter is less important" when performing filter pruning. However, such a standard requires two prerequisites as support. First, the deviation of the filter specification should

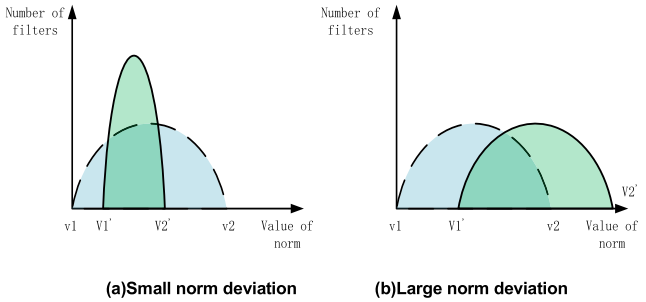


FIGURE 8. Ideal and reality of the norm-based criterion.

be sufficiently big. This requirement makes the search space of the threshold T wide enough such that separating those filters that need to be pruned will be an easy task. Second, the specifications of those filters that can be pruned should be small, that is, close to zero; in other words, filters with a smaller norm are expected to have a minor contribution to the network, rather than a relatively small but active contribution. Figure 8(a) shows the ideal norm distribution when these two requirements are met. The blue curve represents the ideal norm distribution of the network, and v1 and v2 are the minimum and maximum values of the norm distribution, respectively. To choose an appropriate threshold T (shaded in red), two requirements should be met, that is, the specification deviation should be large, and the minimum specification should be as small as any. As shown in Figure 8, the actual filter norm will have a special situation. Figure 8 (a) means that the deviation of the filter norm distribution is too small, and the norm value is concentrated in a small interval, which makes it difficult to find a suitable one. The threshold to select the filter to be trimmed. Figure 8(b) shows that V1'≫V1——>0, those filters considered to be the least important still make a significant contribution to the network, which means

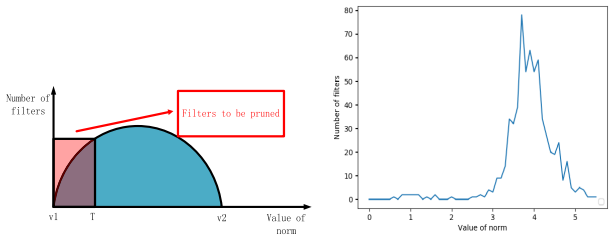


FIGURE 9. Requirements for norm-based filter pruning criterion and visualization of filter distribution.

that each filter has high information significance. Therefore, pruning those filters with the minimum norm will have a negative impact on the network. FPGM(Filter Pruning via Geometric Median) can solve these two problems shown in Figure 8. The central idea of FPGM is as followes: given a set of n points $a^{(1)},\ldots,a^{(n)}$ with each $a^{(i)} \in R^d$, find $x^{(i)} \in R^d$ that minimizes the sum of Euclidean distances to them:

$$x^* = \arg\min_{x\in R^d} f(x) \text{where}$$
$$f(x) \stackrel{\text{def}}{=} \sum_{i\in[1,n]} ||x - a^{(i)}||^2 \tag{12}$$

Here use the geometric median to get the common information of all the filters within the single ith layer:

$$x^{GM} = \arg\min_{x\in R^{N_i\times K\times K}} \sum_{j'\in[1,N_{i+1}]} ||x - F_{i,j'}||^2 \tag{13}$$

In the ith layer, find the filter(s) nearest to the geometric median in that layer:

$$F_{i,j^*} = \arg\min_{F_{i,j'}} ||F_{i,j'} - x^{GM}||^2 \tag{14}$$

then F_{i,j^*} can be represented by the other filters in the same layer, and therefore, pruning them has little negative impacts on the network performance.

We visualized the norm distribution on the lowest layer of the encoder in Dinty-NetV1, the network layer with 1024 feature channels, and obtained the norm distribution as shown in Figure 9(b). It can be seen that the deviation of the norm distribution of the filter of this layer is too small, which basically conforms to the situation that the norm values of

Figure 8(a) are concentrated in a small interval. Therefore, according to the characteristics of the finger veins, the geometric median value is used in this article filter and trim. After experimenting with different pruning rates, we selected a pruning rate of 20% without reducing the network performance to make the network run faster.

IV. EXPERIMENTAND ANALYSIS

A. DATASETS

The data sets used in this paper are the SDU-FV data set created by Shandong University MLA Laboratory and the MMCBNU_6000 data set created by Chonbuk National University in South Korea. There are 106 subjects in the SDU-FV data set. Finger vein images of the index finger, middle finger and ring finger of each person's left and right hands are collected, and six pictures are collected for each finger. Thus, there are 636 categories (106 people × 6 fingers) 3816 (106 people × 6 fingers × 6 sample fingers) images in the library. The MMCBNU_6000 data set has a total of 100 subjects. Finger vein images of the index finger, middle finger, and ring finger of each person's left and right hands are collected. 10 pictures are collected for each finger, and the data set gives the extracted ROI area. Therefore, there are 600 categories (100 people × 6 fingers) and 6000 (600 people × 6 fingers × 10 sample fingers) images in the library. In the experiment we performed, 100 sheets of SDU-FV and MMCBNU_6000 were selected as the training set and 20 sheets as the test set. For the 100 images in the training set, 2000 patches were randomly selected for each image, a total of 200,000 patches were used for network training and one-fifth of the training set was selected as the verification set during each round of training. Twenty images are used as the test set. The size of the patch is the same as that the one used during training. When the width and height are both five steps, multiple consecutive overlapping blocks are extracted for each test image. The probabilities are averaged to obtain the probability that the pixel is a vein. To ensure that the memory limit and real-time performance of the hardware platform are not exceeded, a step size of five is selected in the tradeoff between indicators and time, that is, each 270*150 finger vein test image is divided into 16,340 patches and input to the network before re-splicing.

B. EXPERIMENT PLATFORM

In order to show the versatility of this algorithm on embedded platforms, this article runs on the PC's end environment: Win10, Intel Core i9-990000@3.20GHz CPU, memory 32GB, graphics card NVIDIA GeForce GTX 2080Ti (11GB/GIGABYTE), Pytorch1.4.0 and Python 3.6. The Adam optimizer is used for gradient descent, the learning rate is 0.001, and the batch size is 512.

Relevant experiments have been carried out on the full range of NVIDIA embedded terminals JETSON NANO[34], JETSON TX2, JETSON XAVIAR NXJETSON, JETSON AGX XAVIAR, and verify the structure and ideas of the algorithm proposed in this paper, as well as the feasibility of implementation on the terminal.

NVIDIA JETSON NANO Developer Kit is a powerful small computer that uses a quad-core 64-bit ARM CPU and a 128-core integrated NVIDIA GPU to support all major DL frameworks and tools, including TensorFlow, Pytorch, Caffe/Caffe2, Keras, MXNet and other famous frameworks.

Compared with the previous generation Jetson TX1, NVIDIA Jetson TX2 provides twice the power efficiency, faster calculation speed and stronger reasoning ability. Large deep neural networks can be run on edge devices to achieve higher accuracy. Power consumption was only 7.5 watts, and the energy efficiency is more than 25 times higher than that of a very advanced desktop CPU.

Jetson Xavier NX can use the complete NVIDIA software stack to run modern AI networks and frameworks through

TABLE 2. The performance of DINTY-NET and large networks on the SDU-FV dataset.

Model	Params	Mult-Adds	AUC	ACC	PRE	SPE
R2U-Net	48.92M	-	0.9029	0.9187	0.6218	0.9821
DU-Net	26.73M	-	0.9133	0.9199	0.6420	0.9726
U-Net	13.39M	1.928G	0.8434	0.9117	0.5379	0.9648
Dinty-NetV1 (pruning)	**5.29M**	**171.34M**	0.8620 0.8778	0.9136 0.9091	0.5402 0.5815	0.9549 0.9667
Dinty-NetV2 (pruning)	**3.34M**	**141.94M**	0.8906 0.8587	0.9156 0.9155	0.5528 0.5604	0.9567 0.9631
Dinty-NetV3 (pruning)	**1.156M**	**33.81M**	0.8914 0.8933	0.9156 0.9165	0.5535 0.5613	0.9576 0.9601

TABLE 3. The performance of DINTY-NET and large networks on the MMCBNU_6000 dataset.

Model	Params	Mult-Adds	AUC	ACC	PRE	SPE
R2U-Net	48.92M	-	0.9058	0.9294	0.5468	0.9722
DU-Net	26.73M	-	0.9125	0.9330	0.5882	0.9789
U-Net	13.39M	1.928G	0.8474	0.9103	0.4949	0.9570
Dinty-NetV1 (pruning)	**5.29M**	**171.34M**	0.8225 0.7744	0.9180 0.9134	0.5931 0.5709	0.9835 0.9924
Dinty-NetV2 (pruning)	**3.34M**	**141.94M**	0.7916 0.7949	0.9141 0.9151	0.5825 0.5949	0.9916 0.9905
Dinty-NetV3 (pruning)	**1.156M**	**33.81M**	0.7730 0.8469	0.9090 0.9125	0.4610 0.5166	0.9847 0.9763

acceleration libraries to meet the needs of DL, computer vision, computer graphics, multimedia, etc.

JETSON AGX XAVIAR can obtain help to realize excellent performance of GPU workstations on embedded modules below 30 watts. This developer kit is equipped with a brand-new Xavier processor, designed for autonomous machines, and its performance and energy efficiency are 2 times and 10 times higher than the previous generation NVIDIA Jetson TX2, respectively.

C. PERFORMANCE EVALUATION INDEX

This article uses accuracy, specificity and precision metrics to evaluate our model. Additionally, the receiver operating characteristic curve (ROC) and the area under the curve (AUC) are longer used in medical image segmentation tasks, so this paper takes AUC as the most important index for network performance comparison. We also use PRE and SPE as additional indicators for network performance measurement. PRE(precision) is the proportion of positive examples that are actually positive, and SPE(specificity) is the proportion of all negative examples that are matched. The segmentation of finger veins can be regarded as a binary classification problem. The classification targets can be divided into positive and negative : True positives(TP), False positives(FP), False negatives(FN), True negatives(TN). PRE=$\frac{TP}{TP+FP}$, SPE=$\frac{TN}{N}$.

D. COMPARISON OF PROPOSED NETWORK WITH OTHER LARGE-SCALE NETWORK ON THE PC

To verify the performance of the proposed method on the PC side, we compare the network performance with the basic network U-Net and its modified networks R2U-Net and DU-Net in SDU-FV and MMCBNU_6000. The results are shown in Table 2,3.

Looking at the two databases as a whole, the network sizes of R2U-Net, DU-Net, and U-Net are 43, 23 and 12 times that of Dinty-NetV3, respectively. Dinty-NetV3 has been implemented in terms of network size and model parameters. From the longitudinal perspective of the proposed method in the results of the two tables (excluding pruning), the Dinty-NetV1 V2 V3 on the two databases has gradually reduced in terms of network size and model parameters. Considering the proposed method and its corresponding pruning results, in the six groups of experiments, most of the network indicators after pruning exceeded the original network and the best AUC indicators finally displayed in the two databases were also obtained from the pruning of Dinty-NetV3, which verifies that not all the feature maps in the network that we proposed have a positive effect on the final result and the result will be better if the unimportant features are cut out. For the SDU-FV database, from the perspective of four indicators, when we proposed Dinty-NetV1, three of the four indicators of the network have exceeded the basic network U-Net. When Dinty-NetV3, the three indicators are particularly It is AUC that far exceeds U-Net. Dinty-NetV3 did not exceed the R2U-Net and DU-Net indicators, but it can be seen that it is extremely close. Further, the difference in the number of training parameters is too large, but considering the network size and model parameters. With the huge difference in the training parameters and possibility to be implemented on embedded terminals, we believe that the performance of the proposed model is good enough. For the MMCBNU_6000 database, the performance of Dinty-NetV3 is not as average as in SDU-FV, but it also has its advantages. In the AUC indicator, the gap between Dinty-NetV3 and R2U-Net and DU-Net is slightly larger than that of SDU-FV, but for the Specificity indicator, the three proposed methods and pruning all surpass R2U-Net and DU-Net, the surpass of the precision index on Dinty-NetV2 is particularly obvious, but because this index is too much affected by the number of convolution channels, Dinty-NetV3 has a lower level of this index.

E. COMPARISON OF THE PROPOSED NETWORK WITH OTHER LIGHTWEIGHT NETWORKS ON EMBEDDED PLATFORMS

In addition to comparing U-Net and two large-scale U-Net based networks, this article also compares the model size, segmentation index, and running time of the classic lightweight network in recent years on NVIDIA's full range of embedded platforms. Comprehensive comparison, the results are shown in Table 4 and Table 5.

From Table 4 and Table 5, we can see that Dinty-NetV3 parameters are only 1.156M, which consists of 40%, 34%, and 17% of Squeeze_Unet, Mobile_Unet, and Ghost_Unet. Mult-adds have already demonstrated the superiority of the model in Dinty-NetV1, especially the Mult-adds of Dinty-NetV3 are 12%, 7% and 26% of Squeeze_Unet, Mobile_Unet, Ghost_Unet, reached the optimal model compression.

Tables 4 and 5 show the most important AUC evaluations, running time and other indicators at embedded platforms. Although the network performs equally well on other

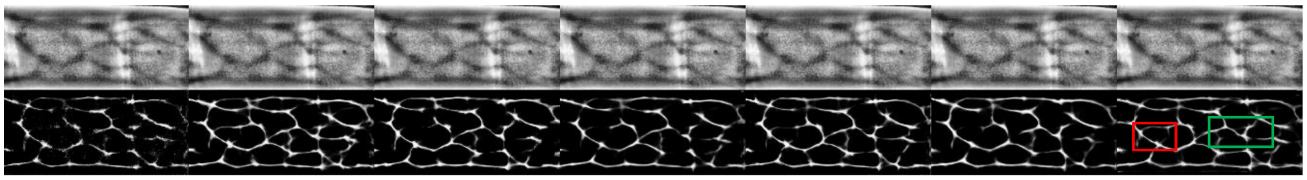


FIGURE 10. U-Net(a), Contrast lightweight network(b)(c)(d) and Dinty-Net(e)(f)(g) segmentation results output on SDU-FV.

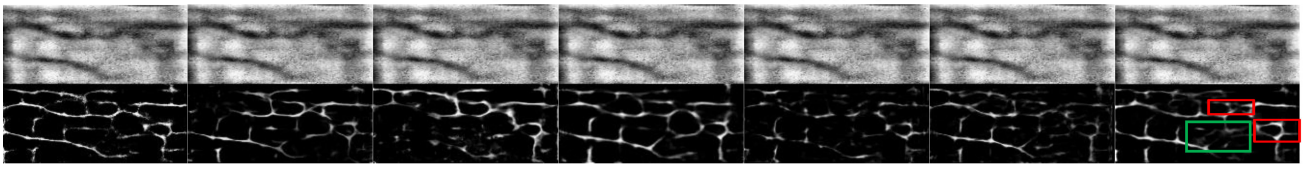


FIGURE 11. U-Net(a), Contrast lightweight network(b)(c)(d) and Dinty-Net(e)(f)(g) segmentation results output on MMCBNU_6000.

TABLE 4. Proposes network and other lightweight network indicators on the SDU-FV dataset.

Model	Params	Mult-adds	AUC	Time (s) /a picture				ACC	PRE	SPE
				Nano	Tx2	NX	AGX			
Squeeze_Unet	2.893M	287.61M	0.8630	1.3521	0.6853	0.6632	0.3881	0.9102	0.5190	0.9472
Mobile_Unet	3.392M	481.35M	0.8554	1.4248	**0.5194**	**0.5053**	**0.2771**	0.9131	0.5368	0.9534
Ghost_Unet	6.783M	128.97M	0.8865	0.6799	0.6839	0.7222	0.3586	0.9184	**0.5848**	**0.9675**
Dinty-NetV1 (pruning)	5.29M	171.34M	0.8620 (0.8778)	0.6898 **(0.6715)**	0.6459 (0.6304)	0.5983 (0.5718)	0.3470 (0.3401)	0.9136 (0.9182)	0.5402 (0.5815)	0.9549 (0.9667)
Dinty-NetV2 (pruning)	3.24M	141.94M	0.8872 (0.8954)	0.7738 (0.7687)	0.6925 (0.6805)	0.6439 (0.6240)	0.3911 (0.3812)	0.9170 **(0.9193)**	0.5598 (0.5786)	0.9571 (0.9619)
Dinty-NetV3 (pruning)	1.156M	33.81M	0.8914 **(0.8933)**	0.8142 (0.7931)	0.7112 (0.6923)	0.6788 (0.6451)	0.4349 (0.4375)	0.9156 (0.9165)	0.5535 (0.5613)	0.9576 (0.9601)

TABLE 5. Proposes network and other lightweight network indicators on the MMCBNU_6000 dataset.

Model	Params	Mult-adds	AUC	Time (s) /a picture				ACC	PRE	SPE
				Nano	Tx2	NX	AGX			
Squeeze_Unet	2.893M	287.61M	0.7829	1.2051	0.6799	0.6991	0.3578	0.9146	0.5442	0.9799
Mobile_Unet	3.932M	481.35M	0.7521	1.3588	**0.5237**	0.7375	**0.2622**	0.8962	0.3775	0.9585
Ghost_Unet	6.783M	128.97M	0.8432	0.6978	0.6126	0.7349	0.3469	0.8721	0.3848	0.8860
Dinty-NetV1 (pruning)	5.29M	171.34M	0.8225 (0.7744)	0.7001 **(0.6818)**	0.6478 (0.6405)	0.5958 **(0.5849)**	0.3501 (0.3456)	**0.9180** (0.9134)	0.5931 (0.5709)	0.9835 **(0.9924)**
Dinty-NetV2 (pruning)	3.24M	141.94M	0.7916 (0.7949)	0.7727 (0.7575)	0.6927 (0.6833)	0.6392 (0.6335)	0.3748 (0.3770)	0.9141 (0.9151)	0.5825 **(0.5949)**	0.9916 (0.9905)
Dinty-NetV3 (pruning)	**1.156M**	**33.81M**	0.7730 **(0.8469)**	0.7975 (0.7794)	0.7083 (0.6953)	0.6651 (0.6462)	0.4263 (0.4235)	0.9090 (0.9125)	0.4610 (0.5166)	0.9847 (0.9763)

indicators include AUC, we takes AUC as the most important index and the following mainly analyzes AUC. In the overall comparison of the AUC of SDU-FV, it can be seen that in the pruning process of Dinty-NetV1, Squeeze_Unet and Mobile_Unet should be surpassed. Dinty-NetV2 is compressing V1 further, while AUC surpasses Dinty-NetV1. Dinty-NetV3 and its pruned network reached the highest AUC. Figure 10 shows the segmentation results of all the SDU-FV networks. Normal network includes U-Net, contrast lightweight networks include Squeeze_UNet, Mobile_UNet, Ghost_UNet and Dinty-Net includes V1, V2 and V3 (pruning). It can be seen from the red box in the figure10 that only Dinty-NetV3 (pruning) has performed a complete segmentation of the blood vessel, which is verified with the highest AUC.

On MMCBNU_6000, it can be seen that the compression of the model has a greater impact on the reduction of network learning power than SDU-FV. The model has achieved better results in Dinty-NetV1, but in the subsequent compression learning, due to the quality of the database, the difference network learned more redundant information, however in the final Dinty-NetV3 (pruning), it still achieved close to the highest results and the AUC was slightly weaker than Ghost_Unet. Figure 11 shows the segmentation results of all the MMCBNU_6000 networks as Figure 10. It can also be seen from the red box in the figure that only Dinty-NetV3 (pruning) has performed a complete segmentation of the blood vessel. Although the Ghost_Unet here is not completely segmented out, on the whole, Ghost_Unet's resolution of segmented images is equivalent to Dinty-NetV3 (pruning). Figure 13,14 show the ROC curves of different networks on two datasets, which (a) is comparative experiment including U-Net, Squeeze_Unet, Mobile_Unet, Ghost_Unet and Dinty-NetV3 (pruning), (b) is Dinty-NetV1 V2 V3 and there's pruning results.

The three-stage network proposed in this paper and its pruned result output are sown in the red, yellow and blue boxes of Figures 12 and 13. The pruned network can distinguish more than the previous network. The details also correspond to that the pruning AUC of the three-stage network in the table is greater than the network before pruning.

In terms of time, it can be seen from all network evaluations that even if the network architecture and acceleration method are not considered, the performance is AGX>NX>TX2>NANO. Comparing the networks, it is found that because other networks have added other module structures to the depth of Mobile_Unet's deep separable convolution, these modules are often more time-consuming than ordinary convolutions, resulting in Mobile_Unet being able to be embedded in most of the embedded convolutions. The terminal needs the shortest running time, however, the gap displayed by all networks on embedded terminals, especially the most powerful AGX, is quite small. Pruning can effectively reduce the network time, but compared with the comparison model, the shortcomings in the time of our proposed model are also exposed. The model in this article is too complicated in operation. At the Dinty-NetV3 stage,

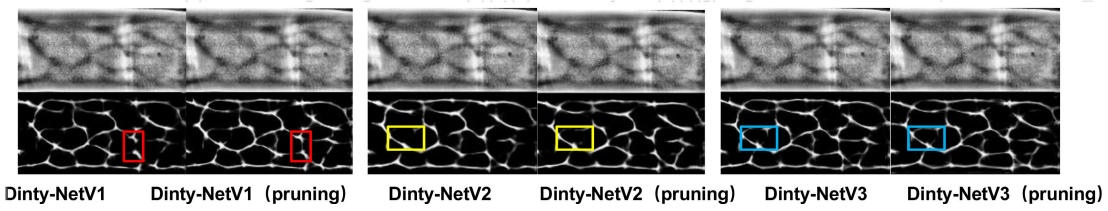


FIGURE 12. Dinty-NetV1 V2 V3 and there's pruning results are compared on SDU-FV.

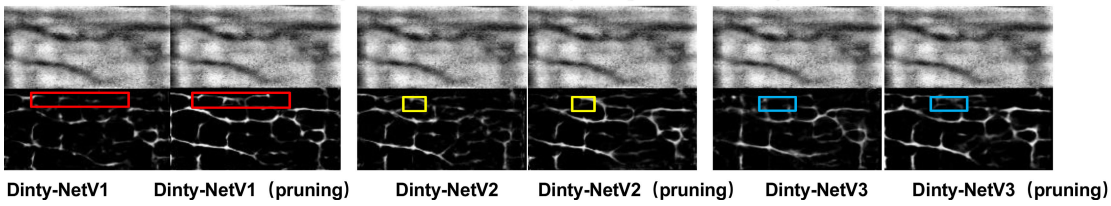


FIGURE 13. Dinty-NetV1 V2 V3 and there's pruning results are compared on MMCBNU_6000.

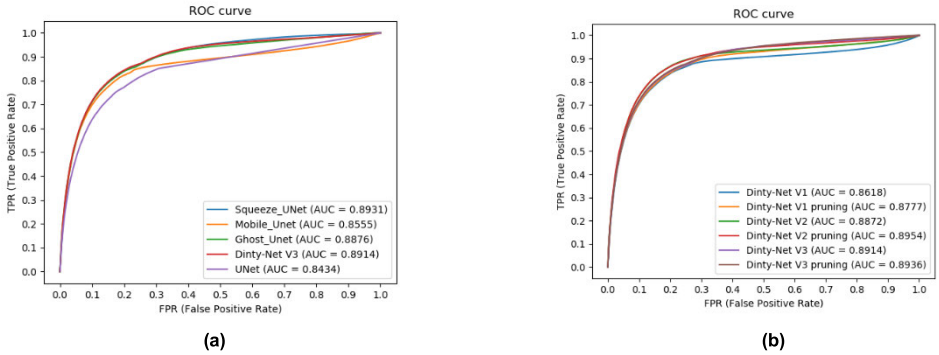


FIGURE 14. ROC curve of on SDU-FV.

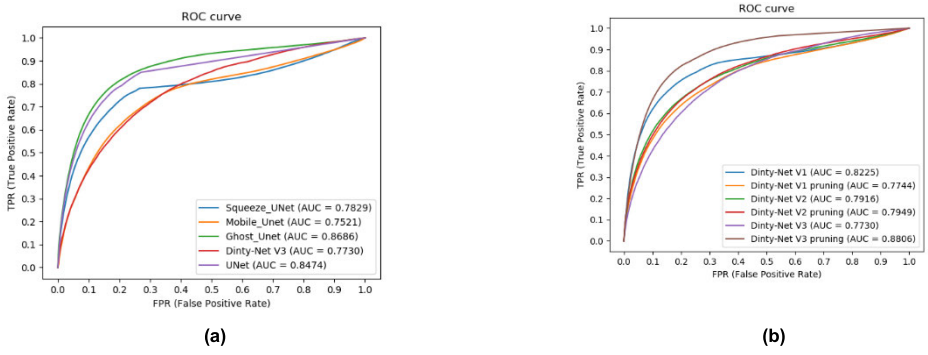


FIGURE 15. ROC curve of on MMCBNU_6000.

except for the transformation of convolution, two modules SE and channel shuffle were used after each convolution. Even if the model has been compressed very small, these operations have increased the network time to a certain extent, so it is still necessary to seek a gap between the compression operation and the running time Balance. In fact, the green box in Figure 10 and11 indicates that the blood vessel failed to segment at this place, and there is a relatively complete finger vein segmentation result in the previous comparison figure. The reason may be the violent selection of pruning rate. The shallow 64 feature map adopts 20% rate and the 1024 feature map adopts 20% pruning rate too. This is obviously easier to lose feature information in the shallow layer, so we maybe use adaptive pruning later to try to solve this problem. In this paper, the model size, segmentation index and running time shown in Tables 4 and 5 are modeled through the Analytic hierarchy process [35] and the consistency ratio is 0.045 less than 0.1. It is considered that there is satisfactory consistency.

TABLE 6. Comprehensive evaluation result of analytic hierarchy process.

Model	SDU-FV	MMCBNU_6000
Squeeze_Unet	0.1053	0.1054
Mobile_Unet	0.1043	0.1033
Ghost_Unet	0.0889	0.0962
Dinty-NetV1	0.0917	0.0944
(pruning)	0.0932	0.0909
Dinty-NetV2	0.1035	0.1021
(pruning)	0.1043	0.1022
Dinty-NetV3	0.1541	0.1498
(pruning)	**0.1546**	**0.1556**

Table 6 shows the indicators of Dinty-NetV1, V2, V3 and other lightweight networks obtained from the analysis we performed. From the Dinty-NetV1, V2, V3 and their pruning networks on the two databases, it can be seen that the analysis results are in line with our expectations. Overall, we can also conclude that the pruning network of Dinty-NetV3 proposed in this paper achieves the best overall model size, segmentation index and running time.

V. CONCLUSION

Aiming at the current research status of finger vein embedded terminals, this paper proposes a lightweight real-time finger vein segmentation method based on embedded terminals. First, we used deep separable convolution to significantly reduce the U-Net parameters of the basic network, followed by introduction of SE module. Reorder features to improve network performance. Secondly, the Ghost module is added to the deep separable convolution, and the feature map of the network part is obtained through the cheap operation alone. After adding channel shuffle, all the characteristic channels were evenly shuffled and reorganized to obtain Dinty-NetV3. Finally, our study on filter norm showed the distribution characteristics of the finger vein picture features. By using the median pruning method, the network models proposed in each stage of this paper achieved better segmentation performance and shorter after pruning Split time. Further, nearly a hundred sets of experiments were conducted on two-finger vein databases on NVIDIA's full range of embedded platforms and classic network compression models in the past 5 years, it is a systematic summary of the previous finger vein segmentation and their lightweight method, and then have a rich a priori effect on the realization of subsequent finger vein recognition and lightweight work in embedded terminals. Because the lightweight methods in this article are highly portable, they can be used as a reference for other algorithms besides image segmentation.

In the future, we plan to explore the multi-objective optimization problem of segmentation performance, network size and the running time of the embedded end with different computing performance. Further, we plan to increase the innovation of the network and without affecting the segmentation performance while compressing the model. This would be a part of our contributions to better feature extraction and recognition in the follow-up. This work contributes to the advancement of embedded terminal series knowledge base. In addition to the NVIDIA series, the embedded terminals of Huawei and Google are added for further networking on terminals with different hardware characteristics in the study.

REFERENCES

[1] A. Jain, L. Hong, and R. Bolle, "On-line fingerprint verification," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 19, no. 4, pp. 302–314, Apr. 1997, doi: 10.1109/34.587996.

[2] D. Zhang, W.-K. Kong, J. You, and M. Wong, "Online palmprint identification," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 25, no. 9, pp. 1041–1050, Sep. 2003, doi: 10.1109/TPAMI.2003.1227981.

[3] A. Kumar and Y. Zhou, "Human identification using finger images," *IEEE Trans. Image Process.*, vol. 21, no. 4, pp. 2228–2244, Apr. 2012, doi: 10.1109/TIP.2011.2171697.

[4] L. Yang, G. Yang, X. Xi, K. Su, Q. Chen, and Y. Yin, "Finger vein code: From indexing to matching," *IEEE Trans. Inf. Forensics Security*, vol. 14, no. 5, pp. 1210–1223, May 2019, doi: 10.1109/TIFS.2018.2871778.

[5] A. Kumar and K. V. Prathyusha, "Personal authentication using hand vein triangulation and knuckle shape," *IEEE Trans. Image Process.*, vol. 18, no. 9, pp. 2127–2136, Sep. 2009, doi: 10.1109/TIP.2009.2023153.

[6] Y. Zhou and A. Kumar, "Human Identification Using Palm-Vein Images," in *IEEE Trans. Inf. Forensics Security*, vol. 6, no. 4, pp. 1259–1274, Dec. 2011, doi: 10.1109/TIFS.2011.2158423.

[7] M. A. Turk and A. P. Pentland, "Face recognition using eigenfaces," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, Maui, HI, USA, Dec. 1991, pp. 586–591, doi: 10.1109/CVPR.1991.139758.

[8] J. Daugman, "How iris recognition works," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 14, no. 1, pp. 21–30, Jan. 2004, doi: 10.1109/TCSVT.2003.818350.

[9] J. Ramirez, J. C. Segura, J. M. Gorriz, and L. Garcia, "Improved voice activity detection using contextual multiple hypothesis testing for robust speech recognition," *IEEE Trans. Audio, Speech Lang. Process.*, vol. 15, no. 8, pp. 2177–2189, Nov. 2007, doi: 10.1109/TASL.2007.903937.

[10] N. Adelson, "Analyzing and recognizing walking figures in XYT," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Seattle, WA, USA, 1994, pp. 469–474, doi: 10.1109/CVPR.1994.323868.

[11] M. A. El-Yacoubi, M. Gilloux, and J.-M. Bertille, "A statistical approach for phrase location and recognition within a text line: An application to street name recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 24, no. 2, pp. 172–188, Feb. 2002, doi: 10.1109/34.982898.

[12] J. Yang and Y. Shi, "Finger–vein ROI localization and vein ridge enhancement," *Pattern Recognit. Lett.*, vol. 33, no. 12, pp. 1569–1579, Sep. 2012, doi: 10.1016/j.patrec.2012.04.018.

[13] E. Shelhamer, J. Long, and T. Darrell, "Fully convolutional networks for semantic segmentation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 4, pp. 640–651, Apr. 2017, doi: 10.1109/CVPR.2015.7298965.

[14] V. Badrinarayanan, A. Kendall, and R. Cipolla, "SegNet: A deep convolutional encoder-decoder architecture for image segmentation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 12, pp. 2481–2495, Dec. 2017, doi: 10.1109/TPAMI.2016.2644615.

[15] G. Lin, A. Milan, C. Shen, and I. Reid, "RefineNet: Multi-path refinement networks for high-resolution semantic segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 1925–1934, doi: 10.1109/cvpr.2017.549.

[16] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in *Proc. 18th Int. Conf. Med. Image Comput. Comput.-Assist. Intervent.*, vol. 9351, 2015, pp. 234–241, doi: 10.1007/978-3-319-24574-4_28.

[17] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Las Vegas, NV, USA, Jun. 2016, pp. 770–778, doi: 10.1109/CVPR.2016.90.

[18] Gartner. (2019). *Gartner Says 8.4 Billion Connected 'Things' Will Bein Use in 2017, Up 31 Percent From 2016*. [Online]. Available: https://www.gartner.com/en/newsroom/press-releases/2017-02-07-gartner-says-8-billion-connected-things-will-be-in-use-in-2017-up-31-percent-from-2016

[19] S. Han, X. Liu, H. Mao, J. Pu, A. Pedram, M. Horowitz, and B. Dally, "Deep compression and EIE: Efficient inference engine on compressed deep neural network," in *Proc. IEEE Hot Chips 28 Symp. (HCS)*, Aug. 2016, p. 28, doi: 10.1109/HOTCHIPS.2016.7936226.

[20] A. G. Howard, M. Zhu, B. Chen, D. Kalenichenko, W. Wang, T. Weyand, M. Andreetto, and H. Adam, "MobileNets: Efficient convolutional neural networks for mobile vision applications," 2017, *arXiv:1704.04861*. [Online]. Available: http://arxiv.org/abs/1704.04861

[21] C. Szegedy, S. Ioffe, V. Vanhoucke, and A. Alemi, "Inception-v4, inception-ResNet and the impact of residual connections on learning," 2016, *arXiv:1602.07261*. [Online]. Available: http://arxiv.org/abs/1602.07261

[22] Y. He, G. Kang, X. Dong, Y. Fu, and Y. Yang, "Soft filter pruning for accelerating deep convolutional neural networks," in *Proc. 27th Int. Joint Conf. Artif. Intell.*, Jul. 2018, pp. 1–8, doi: 10.24963/ijcai.2018/309.

[23] H. Li, A. Kadav, I. Durdanovic, H. Samet, and H. Peter Graf, "Pruning filters for efficient ConvNets," 2016, *arXiv:1608.08710*. [Online]. Available: http://arxiv.org/abs/1608.08710

[24] J. Ye, X. Lu, Z. Lin, and J. Z. Wang, "Rethinking the smaller-norm-less-informative assumption in channel pruning of convolution layers," 2018, *arXiv:1802.00124*. [Online]. Available: http://arxiv.org/abs/1802.00124

[25] M. Zahangir Alom, M. Hasan, C. Yakopcic, T. M. Taha, and V. K. Asari, "Recurrent residual convolutional neural network based on U-Net (R2U-Net) for medical image segmentation," 2018, *arXiv:1802.06955*. [Online]. Available: http://arxiv.org/abs/1802.06955

[26] Q. Jin, Z. Meng, T. D. Pham, Q. Chen, L. Wei, and R. Su, "DUNet: A deformable network for retinal vessel segmentation," *Knowl.-Based Syst.*, vol. 178, pp. 149–162, Aug. 2019, doi: 10.1016/j.knosys.2019.04.025.

[27] K. Han, Y. Wang, Q. Tian, J. Guo, C. Xu, and C. Xu, "GhostNet: More features from cheap operations," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 1580–1589, doi: 10.1109/CVPR42600.2020.00165.

[28] N. Ma, X. Zhang, and H. T. Zheng, "ShuffleNet V2: Practical guidelines for efficient CNN architecture design," in *Proc. ECCV*, 2018 pp. 122–138, doi: 10.1007/978-3-030-01264-9_8.

[29] Y. He, P. Liu, Z. Wang, Z. Hu, and Y. Yang, "Filter pruning via geometric median for deep convolutional neural networks acceleration," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 4340–4349, doi: 10.1109/CVPR.2019.00447.

[30] J. Hu, L. Shen, and G. Sun, "Squeeze-and-excitation networks," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 42, no. 8, pp. 2011–2023, Aug. 2017, doi: 10.1109/TPAMI.2019.2913372.

[31] Y. Yin, L. Liu, and X. Sun, "SDUMLA-HMT: A multimodal biometric database," in *Proc. Chin. Conf. Biometric Recognit.* Berlin, Germany: Springer, 2011, pp. 260–268, doi: 10.1007/978-3-642-25449-9_33.

[32] Y. Lu, S. Xie, S. Yoon, Z. Wang, and D. Park, "An available database for the research of finger vein recognition," *Proc. Int. Congr. Image Signal Process.*, Dec. 2014, pp. 410–415, doi: 10.1109/CISP.2013.6744030.

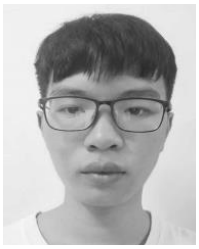
[33] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L.-C. Chen, "MobileNetV2: Inverted residuals and linear bottlenecks," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 4510–4520, doi: 10.1109/CVPR.2018.00474.

[34] H. Ali Imran, U. Mujahid, S. Wazir, U. Latif, and K. Mehmood, "Embedded development boards for edge-AI: A comprehensive report," 2020, *arXiv:2009.00803*. [Online]. Available: http://arxiv.org/abs/2009.00803

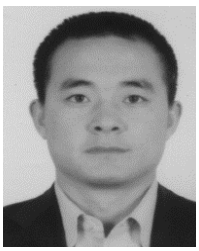
[35] T. L. Saaty. (Oct. 2001). *Analytic Hierarchy Process*. [Online]. Available: https://doi.org/10.1007/1-4020-0611-X_31

JUNYING ZENG (Member, IEEE) received the master's degree in physical electronic from Yunnan University, Yunnan, China, in 2005, and the Ph.D. degree in physical electronics from the Beijing University of Posts and Telecommunications, Beijing, China, in 2008. He is currently an Associate Professor with Wuyi University, Guangdong, China. Since June 2008, he has been working with the Department of Intelligence Manufacturing, Wuyi University, Guangdong, China. His research interests include image understanding, deep learning, and signal processing.

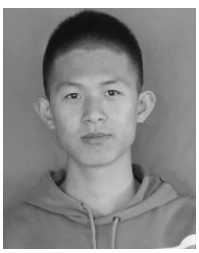
BOYUAN ZHU received the B.S. degree from Shenyang LiGong University, in 2017. He is currently pursuing the master's degree with the Department of Intelligence Manufacturing, Wuyi University. His research interests include biometric image segmentation and deep learning.



YUJIE HUANG is currently pursuing the bachelor's degree with the Department of Intelligence Manufacturing, Wuyi University. His research interests include image segmentation of medical imaging and deep learning.



CHUANBO QIN received the B.S. and M.S. degrees from WYU University, China, in 2004 and 2008, respectively, and the Ph.D. degree from the Department of Automation Science and Engineering, South China University of Technology, in 2015. He is currently a Lecturer with WYU University. His main research directions are medical image segmentation and biometric recognition.



JINGMING ZHU received the B.S. degree in mechanical engineering from Shandong Jiao Tong University, China, in 2019. He is currently pursuing the master's degree. His research is focused on computer vision.

FAN WANG received the B.S. degree from the Qilu University of Technology (Shandong Academy of Sciences), in 2016, and the M.S. degree from Wuyi University, in 2020. He is currently a Teacher with Guangdong Ocean University. His research interests include biometric identification and deep learning.

YIKUI ZHAI (Member, IEEE) received the bachelor's degree in optical electronics information and communication engineering and the master's degree in signal and information processing from Shantou University, Guangdong, China, in 2004 and 2007, respectively, and the Ph.D. degree in signal and information processing from Beihang University, in June 2013. He is currently an Associate Professor with Wuyi University, Guangdong, China. He has been a Visiting Scholar with the Department of Computer Science, University of Milan, since 2016. Since October 2007, he has been working with the Department of Intelligence Manufacturing, Wuyi University, Guangdong, China. His research interests include image processing, deep learning, and pattern recognition.

JUNYING GAN (Member, IEEE) received the B.S., M.S., and Ph.D. degrees in electrical information engineering from Beihang University, in 1987, 1992, and 2003, respectively. She is currently a Full Professor with Wuyi University, Guangdong, China. She joined Wuyi University, in 1992. She is the Executive Director of Guangdong image graphics association. She has published more than 50 journal articles, and has received several provincial technology awards. Her research interests include biometric extraction and pattern recognition.

YUCONG CHEN received the bachelor's degree from Wuyi University, where he is currently pursuing the master's degree with the Intelligent Manufacturing Department. His research interests include image processing and deep learning.

YINGBO WANG received the B.S. degree from Dalian Minzu University, in 2020. He is currently pursuing the master's degree with the Department of Intelligence Manufacturing, Wuyi University. His research interests include biometric identification and deep learning.

RUGGERO DONIDA LABATI (Member, IEEE) received the Ph.D. degree in computer science from the Universitá degli Studi di Milano, Crema, Italy, in 2013. He has been an Assistant Professor in computer science with the Universitá degli Studi di Milano, since 2015. He has been a Visiting Researcher with Michigan State University, East Lansing, MI, USA. Original results have been published in more than 50 papers in international journals, proceedings of international conferences, books, and book chapters. His current research interests include intelligent systems, signal and image processing, machine learning, pattern analysis and recognition, theory and industrial applications of neural networks, biometrics, and industrial applications. He is an Associate Editor with *Journal of Ambient Intelligence and Humanized Computing* (springer).

VINCENZO PIURI (Fellow, IEEE) received the M.S. and Ph.D. degrees in computer engineering from Politecnico di Milano, Italy. He was an Associate Professor with Politecnico di Milano, Italy, from 1992 to 2000, a Visiting Professor with the University of Texas at Austin, USA, from summers 1996 to 1999, and a Visiting Researcher with George Mason University, USA, from summers 2012 to 2016. He founded a start-up company, Sensuresrl, in the area of intelligent systems for industrial applications (leading it from 2007 to 2010) and was active in industrial research projects with several companies. His main research and industrial application interests are intelligent systems, computational intelligence, pattern analysis and recognition, machine learning, signal and image processing, biometrics, intelligent measurement systems, industrial applications, distributed processing systems, Internet-of-Things, cloud computing, fault tolerance, application-specific digital processing architectures, and arithmetic architectures. He is an ACM Fellow and a Full Professor with the University of Milan, Italy, since 2000, where he was also a Department Chair, from 2007 to 2012.

FABIO SCOTTI (Senior Member, IEEE) received the Ph.D. degree in computer engineering from the Politecnico di Milano, Milan, Italy, in 2003. He has been an Associate Professor in computer science with the Universitá degli Studi di Milano, Crema, Italy, since 2015. Original results have been published in more than 100 papers in international journals, proceedings of international conferences, books, book chapters, and patents. His current research interests include biometric systems, machine learning and computational intelligence, signal and image processing, theory and applications of neural networks, three-dimensional reconstruction, industrial applications, intelligent measurement systems, and high-level system design. He is an Associate Editor with the IEEE TRANSACTIONS ON HUMAN-MACHINE SYSTEMS AND SOFT COMPUTING (Springer).