IET Journals
The Institution of Engineering and Technology

Deep Finger Texture Learning for Verifying People

ISSN 1751-8644
doi: 0000000000
www.ietdl.org

R. R. Omar[1], *Tingting Han*[1], *S. A. M. Al-Sumaidaee*[2], *Taolue Chen*[1]*

[1] *Department of Computer Science and Information Systems, Birkbeck, University of London, London, UK*
[2] *College of Engineering, Newcastle University, Newcastle upon Tyne, UK*
* *E-mail: taolue@dcs.bbk.ac.uk*

Abstract: Finger Texture (FT) is currently attracting significant attentions in the area of human recognition. Finger texture covers the area between the lower knuckle of the finger and the upper phalanx before the fingerprint. It involves rich features which can be efficiently used as a biometric characteristic. In this paper, we contribute to this growing area by proposing a new verification approach, i.e., Deep Finger Texture Learning (DFTL). To the best of our knowledge, this is the first time that deep learning is employed for recognizing people by using the FT characteristic. Four databases have been used to evaluate the proposed method: the Hong Kong Polytechnic University Contact-free 3D/2D (PolyU2D), Indian Institute of Technology Delhi (IITD), CASIA Blue spectral (CASIA-BLU) corresponding to spectral 460nm and CASIA White spectral (CASIA-WHT) from the CASIA Multi-Spectral images database. The obtained results have shown superior performance compared with recent literature. The Verification Accuracies (VAs) have attained 100%, 98.65%, 100% and 98% for the four databases of PolyU2D, IITD, CASIA-BLU and CASIA-WHT, respectively.

1 Introduction

In recent years, different finger patterns have been investigated in the area of person recognition. Examples include finger geometry [1], finger vein [2], finger outer knuckle [3] and finger texture (FT) [4][5]. Using FTs of four or five fingers in terms of individual verification has been confirmed to be valuable [6][7][8].

Fundamentally, the FT is located between the upper phalanx (directly under the fingerprint) and the lower knuckle (the base of the finger). It comprises three phalanxes and three knuckles on the inner surface of a little, ring, middle or index finger, and two phalanxes and two knuckles on the inner surface of a thumb. Thus, it involves different qualified patterns. The main FT positions in a hand image are given in Fig. 1 whereas the main FT parts in a middle finger and thumb are given in Figs. 2 and 3.

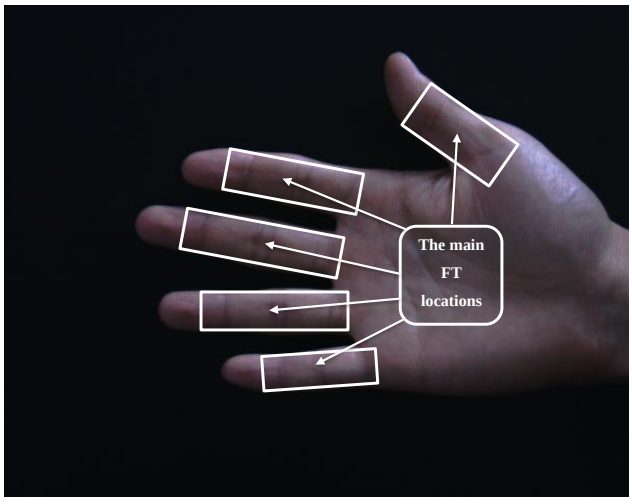

Fig. 1: The main FT locations in the five fingers (little, ring, middle, index and thumb)

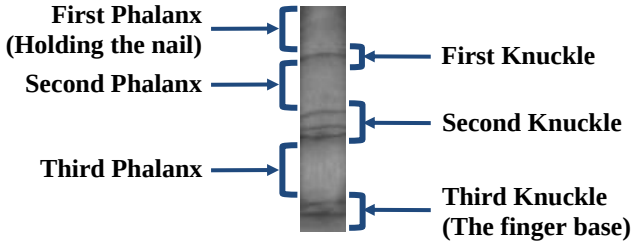

Fig. 2: Main FT parts in a middle finger (two phalanxes and two knuckles)

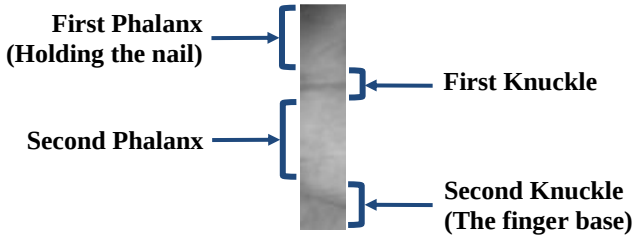

Fig. 3: Main FT parts in a thumb (three phalanxes and three knuckles)

The FT patterns are unique and reliable between people (even between identical twins). It generally includes visible principal lines and wrinkles [9]. A low resolution or an inexpensive acquisition device can be used to collect FT images [10]. All FT patterns have natural protection as they are positioned inside the fist [8]. It has advantage over the fingerprint as it contains more features.

FT is recently employed for person recognition. It was used to identify individuals as in [11][12] and to verify individuals as in [13][8].

The aim of this paper is to enhance the human verification accuracy and performance by introducing a novel approach termed the

Deep Finger Texture Learning (DFTL). The main contribution of this paper is to apply the deep learning technique with the biometric finger texture for verifying people faster and more accurately. This work overcomes main FT verification drawbacks observed in recent studies. These drawbacks include applying multiple processing stages, employing big neural network architectures, using large memory sizes and exploiting all the fingers to achieve the best performance.

The rest of this paper is organized as follows: Section 2 presents the related prior work. Section 3 explains the methodology of the work. Section 4 discusses the results. Finally, Section 5 concludes the paper.

2 Related Work

The idea of exploiting the FT as a biometric characteristic started in 2005, where Ribaric and Fratric fused the FTs of five fingers with the palmprint for individual identification [11]. In 2007, Ferrer *et al.* [14] proposed a low cost fusion system between the palmprint, hand geometry and FTs of four fingers (little, ring, middle and index). In 2009, Pavesic *et al.* combined the FTs of the four fingers with their fingerprints to establish a comparison study of different feature extraction methods [12]. In 2010, Michael *et al.* [10, 15] described a recognition system by fusing the FTs of fingers with the palm print in the case of verification. Five fingers were used in [10], whilst, the thumb was excluded in [15]. In 2011, Kanhangad *et al.* also utilized the FTs as a part of their study, where the FTs were fused with different hand characteristics to enhance the recognition [13]. In 2014, Bhaskar and Veluchamy described a multiple biometric system using a combination of palmprint features and FTs features [9]. This study did not mention the number of employed fingers.

Table 1 A summary of related FT characteristic studies

Reference	Year	Employed Objects	No. of Utilized Fingers
Ribaric and Fratric [11]	2005	FTs and palmprint	5
Ferrer *et al.* [14]	2007	FTs, palmprint and hand geometry	4
Pavesic *et al.* [12]	2009	FTs and fingerprints	4
Michael *et al.* [10]	2010	FTs and palmprint	5
Michael *et al.* [15]	2010	FTs and palmprint	4
Kanhangad *et al.* [13]	2011	FTs, 2D palmprint, 2D hand geometry, 3D palmprint and 3D hand geometry	4
Bhaskar and Veluchamy [9]	2014	FTs and palmprint	—
Al-Nima *et al.* [8]	2015	Four FTs	4
Al-Nima *et al.* [6]	2017	Five FTs	5
Al-Nima *et al.* [4]	2017	Five FTs	5

In 2015, Al-Nima *et al.* investigated employing the full FT regions of four fingers to enhance the verification performance. Concatenations between extracted FT features with Probabilistic Neural Networks (PNNs) were applied to confirm the outcome [8]. The main regions of the FTs for the four fingers were specified in that work. More recently, Al-Nima *et al.* expanded the FT study by providing a robust finger segmentation approach, enhancing a feature extraction method and introducing a novel approach to salvage the missing FT features [5]. They also proposed another finger segmentation approach to collect the finger images from different hand postures and consequently extracted the Region of Interests (ROIs) of the five FTs [6]. In 2017, Al-Nima *et al.* explained a new FT feature extraction method called the Multi-scale Sobel Angles Local Binary Pattern (MSALBP) and described an innovation classifier named the Finger Contribution Fusion Neural Network (FCFNN) in terms of person verification [4]. A summary of related FT characteristic studies are given in Table 1.

It can be investigated from the prior FT studies that the deep learning has not been applied to this phenomenon. In addition, previous FT characteristic publications had multiple processes for person recognition. In particular, best FT feature extraction methods usually consist of different stages of first analysing the vertical and horizontal patterns, and then extracting the features from the resulted values as in [5][4][6]. Data arrangements are always performed to justify the variances between different sample information, for example, applying data windowing and statistical computations [8][7]. The determined classifier is commonly selected under the requirements of managing and fusing various data, for instance, combining the FT features of four or five fingers as in [12][13][5][4]. Therefore, adopted classifiers usually have big architectures.

This study can reduce the number of operations and enhance the FT performance human verification.

3 Methodology

It is usually the case that five times as many FT images can be obtained from hand images. This would mostly result in a large number of FT images, which needs a powerful technique that can process huge databases or datasets. Deep learning is one of such techniques and we will show how it works in our setting as the Deep Finger Texture Learning (DFTL).

As an overview of the process, the main architecture of DFTL is given in Fig. 4. Unlike the convolutional neural network [16][17][18], the layers do not need to be repeated. For instance, there should be only one convolution layer and one ReLU layer, etc. This is because the FT patterns have simple descriptions of horizontal and vertical patterns. As a result, it greatly simplifies the algorithm and boost the performance.

In the following, we will explain each layer in details.

3.1 Finger Segmentation and FT Region Extraction

The DFTL structure starts from the finger segmentation. This has been achieved by applying the finger segmentation method in [5]. After that, extracting the ROI of the FT has been performed. A useful method is utilized based on the Adaptive Inner Rectangle (AIR). This method has been reported in [13], where an adaptive inner rectangle was applied to extract the ROI for just the four fingers (index, middle, ring and little). However, in this publication the lower or third inner knuckle was not considered and thus important features of the FTs are avoided. This knuckle contains important textures according to [8]. A further modified model for the adaptive ROI rectangle is applied here, where more features can be collected for each finger from the ROI areas. All the five fingers including the thumb are considered. After specifying the ROIs, a fixed resize has been applied to each finger image in order to normalize them into fixed sized vectors. The normalization resize is considered equal to 30×150 following [4–7]. Each FT image is resized to 30×150 pixels as this size can provide fair comparisons with prior studies. In addition, this size was examined and found to be appropriate in [5].

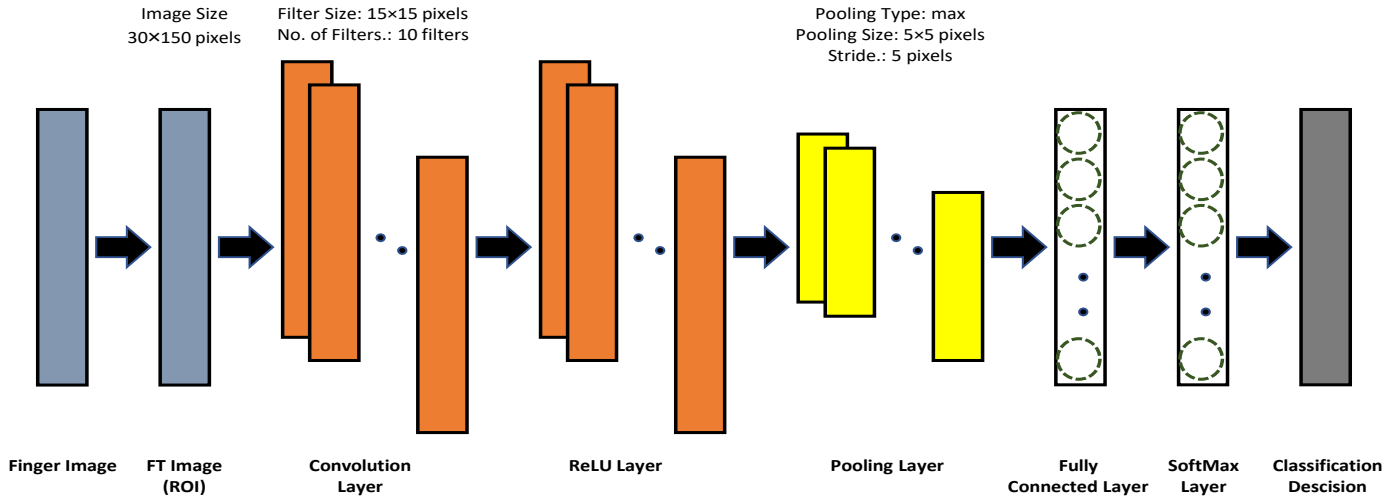

Fig. 4: The proposed architecture of the DFTL, it only exploits main deep learning layers

3.2 Convolution Layer

One convolution layer is used. Empirically, this has been found to be sufficient, because the FTs consist of simple patterns. In this layer, the FT input image $\mathbf{X}$ will be transformed into feature maps. The feature map is denoted as a convolution with a kernel weights of a channel number C, which is here equal to 1 as we are using one channel of grayscale FT images. In general, this kernel has a size of $k_h \times k_w \times C$ followed by the addition of bias. Note that k_h and k_w represent the height and width of the kernel weights, respectively. Let $W^{c^l}_{i,j,c^{l-1}}$ be the components of the kernel weights, B_{c^l} represents the channel bias of the convolution layer, $l-1$ represents the previous layer (input layer here) and l represents the current layer. The value z_{u,v,c^l} of an assigned pixel at (u,v) in channel c^l of the layer l is given by:

$$z_{u,v,c^l} = B_{c^l} + \sum_{i=-k_h^l}^{k_h^l} \sum_{j=-k_w^l}^{k_w^l} \sum_{c^{l-1}=1}^{C^{l-1}} W^{c^l}_{i+k_h^l, j+k_w^l, c^{l-1}} z_{u+i,v+j,c^{l-1}} \tag{1}$$

where z_{u,v,c^l} represents the outcome of a convolution layer node, $k_h^l = (k_h - 1)/2$ and $k_w^l = (k_w - 1)/2$ [19].

3.3 ReLU Layer

Subsequently, a Rectified Linear Unit (ReLU) activation function is applied in this layer. This layer has an advantage of providing non-linear computations to the DFTL. It removes the negative values and preserves the positive values from the feature maps. Equation (2) is utilized to perform the ReLU activation function:

$$o_{u,v,c^l} = f(z_{u,v,c^l}) = \max(0, z_{u,v,c^l}) \tag{2}$$

where o_{u,v,c^l} represents the outcome of a ReLU layer node and $\max$ represents the maximum operation [20].

3.4 Pooling Layer

Hereafter, a pooling layer is employed. The pooling layer played an essential role in reducing the feature maps of the FT features. It collected the maximum values of the feature maps from the previous layer, which is the ReLU layer. In other words, it reduced the extracted features sizes of the FT and preserved the maximum values of these features. In general, the pooling layer can be implemented according to the following equation:

$$q_{a^l,b^l,c} = \max_{0 \le a < p_h, 0 \le b < p_w} o_{a^l \times p_h + a,\ b^l \times p_w + b,\ c} \tag{3}$$

where $q_{a^l,b^l,c}$ represents the outcome of a pooling layer node, $0 \le a^l < p_h^l$, p_h^l represents the height of the pooled feature map, $0 \le b^l < p_w^l$, p_w^l represents the width of the pooled feature map, $0 \le c < C^l = C^{l-1}$, p_h and p_w respectively represent the height and width of the feature map subregions that need to be pooled [21].

3.5 Fully Connected Layer

Now, fully connected layer is applied to map between the pooling layer and the required number of recognizing subjects. Equation (4) simulates the operation of the fully connected layer. The produced feature maps in layer $l-1$ are interpreted as $m_1^{l-1} \times m_2^{l-1} \times m_3^{l-1}$ dimensional vectors as follows:

$$g_r = \sum_{a=1}^{m_1^{l-1}} \sum_{b=1}^{m_2^{l-1}} \sum_{c=1}^{m_3^{l-1}} W^l_{a,b,c,r} (\boldsymbol{Q}_c)_{a,b}\ , \qquad \forall 1 \le r \le m^l \tag{4}$$

where g_r represents the outcome of a fully connected layer node, m_1^{l-1} and m_2^{l-1} represent the height and width of a feature map in the previous layer respectively, m_3^{l-1} represents the number of generated feature maps in the previous layer, $W^l_{a,b,c,r}$ represents the connection weights between the pooling layer and the fully connected layer, $\boldsymbol{Q}_c$ represents vector(s) of pooling layer outcomes, and m^l represents the number of required recognizing subjects [22].

3.6 Softmax Layer

Consequently, softmax activation function can be used to compute posterior classification probabilities. It utilizes the following equation:

$$y_r = \frac{\exp(g_r)}{\sum_{s=1}^{m^{l-1}} \exp(g_s)} \tag{5}$$

where y_r represents the outcome of a softmax layer node [22].

3.7 Classification Layer

Finally, to achieve the final decision, a classification layer is used. This layer follows a competitive rule called the winner-takes-all rule.

The operation in this layer can be described in Equation (6):

$$D_r = \begin{cases} 1 & \text{if } y_r = max \\ 0 & \text{otherwise} \end{cases}, \quad r = 1, 2, ..., class \tag{6}$$

where D_r represents the output decision of a classification layer node and max represents the extracted maximum y_r value [23].

Due to the fact that a multi-class classifier can be utilized for verifying subjects [8][4], we have adapted the DFTL for the human verification by using its multi-class outputs of the classification layer.

4 Results and Discussions

4.1 Descriptions of Employed Databases

Four databases with a large number of contact-less hand images have been employed in this paper.

1) The Hong Kong Polytechnic University Contact-free 3D/2D (PolyU2D) Hand Images Database (Version 1.0) [24].

2) The Indian Institute of Technology Delhi (IITD) Palmprint Image Database (Version 1.0) [25][26].

3) A specific spectral wavelength of 460nm (or blue light) from the CASIA Multi-Spectral Palmprint image database (Version 1.0) [27]. We will refer to this database by CASIA-BLU.

4) A white spectral (or white light) also from the CASIA Multi-Spectral Palmprint image database (Version 1.0) [27]. We will refer to this database by CASIA-WHT.

The first database consists of 8,850 fingers from 1770 very low resolution hand images. For the second database, a total of 4,440 finger images involved in 888 high resolution hand images have been used in this study. The participants of this database were asked to provide high degrees of hand movements during the capturing step [25][26]. The third database contains FT features as mentioned in [28][29]. It has 3,000 fingers included in 600 low resolution hand images. It is valuable to work with this type of database as it has different FT features than that acquired under the normal light. The fourth database contains standard FT features as it was collected under a white light. It has the same number of samples as in CASIA-BLU. From all the four databases, 2D right hand images are only used in this work. A summary of the specifications of employed samples for the four databases is demonstrated in Table 2.

Table 2 Summary of the specifications of employed samples for the four databases

Database	Number of People	Number of Right Hand Images	Number of Fingers (or FTs)	Database Resolution According to [5]
PolyU2D	177	1770	8,850	Very low resolution
IITD	148	888	4,440	High resolution
CASIA-BLU	100	600	3,000	Low resolution
CASIA-WHT	100	600	3,000	Low resolution

4.2 Practical Experiments

DFTL Inputs & Outputs: The following process has been applied to all the employed databases: 25 finger images for each person or subject have been utilized in the training phase following [5][6][4]; the remaining samples have been used in the testing phase. All the five finger images have been considered in this paper. However, any finger can confirm the successful verification in the DFTL.

The ROIs of the five-finger images have been extracted according to [5][4], where the image resizing is equally normalized to 30×150 pixels. By employing similar FT normalizations, fair comparisons can be established.

The output of the DFTL is to generate the verification decision, where each neuron in the last layer refers to a person. The number of the output values is equal to the number of the output neurons and this is equal to the number of utilized subjects. This means that 177, 148, 100 and 100 output neurons have been assigned for the PolyU2D, IITD, CASIA-BLU and CASIA-WHT databases, respectively.

Training DFTL: The suggested DFTL network has been trained for each set of FTs in each database. The following parameters were used during the training phase: Stochastic Gradient Descent with Momentum (SGDM) optimizer, momentum value of 0.9, fixed learn rate value equal of 0.03, weight decay equal to 0.0001 and mini batch size equal to 128. Fig. 5 indicates the success of the training phases for the four databases. It illustrates the relationships between the iterations and the Mini-batches losses, where a mini-batch is defined in [30] as a training data sub-group and it is used to adjust the weights after assessing the gradient of the loss function. The loss function is denoted as the cross entropy loss function, which is the probability that the network assigns the class r for the eth input $P(t_r = 1|x_e)$ [30]. The differences between the curves are due to the differences between the features of the FTs, where each database has different specifications as the various number of training samples.

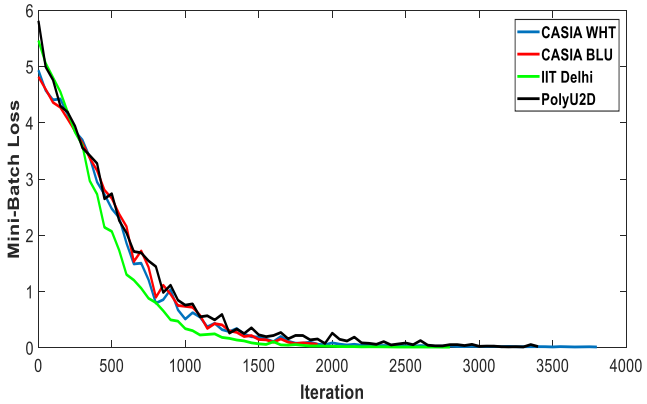

Fig. 5: The relationships between the iterations and the mini-batches losses for the four employed databases during the training phases

DFTL Parameters: Many experiments are applied to select the appropriate DFTL parameters. Examples of these experiments are:

1) Specifying the appropriate filter size for the convolution layer of the DFTL. The VAs were evaluated for various filter sizes of ten filters. In order to avoid the time consuming of learning the DFTL, 50

epochs were determined in all trainings. Table 3 shows the obtained VAs by changing the convolution layer filter size.

From this table, it can be investigated that almost all the databases have confirmed the filter size of 15×15, which could provide the best VA values. This is clearly appeared in the IITD, CASIA-BLU and CASIA-WHT databases. On the other hand, the filter size of 13×13 has provided the best VA value for the PolyU2D database, but this size cannot be considered with the other databases especially the CASIA-BLU. In order to choose the optimal DFTL which can be applied under any circumstances, the filter size of 15×15 seems to be the appropriate selection.

2) Determining the suitable number of filters for the DFTL convolution layer. The VAs were observed after selecting the filter size of 15×15. Table 4 gives the attained VAs by tuning the number of filters in the DFTL convolution layer.

From this table, it has been approved that the best number of filters to be selected is 10. This is because this number of filters achieved the best VAs in all the used databases. Other number of filters (4, 6 and 8) were almost provided values around the best benchmarked accuracies. The number of filters which is equal to 8 attained the same best recorded performances for two databases, the PolyU2D and IITD. However, this number of filters can not be considered because it failed to achieve the best values in the CASIA-BLU and CASIA-WHT databases. So, again the number of filters which is equal to 10 appears the appropriate choice for the DFTL convolution layer.

3) Assigning the parameters of the DFTL pooling layer. The pooling type was examined as given in Table 5.

From this table, the verification performances of the maximum pooling type is obviously better than the performances of the average pooling type. So, the pooling type has been assigned to the maximum as this type can maintain the extracted features from the previous layer. Whereas, pooling average numbers may waste the high values of FT features by combining them with the low values according to the average computations. The pooling filter size with the stride have been determined to 5×5 pixels and 5 pixels, respectively. This is because that the pooling size filters and the stride parameters are respectively equivalent to the parameters of the windowing size and non-overlapped partitions in [8][5][6][4]. Therefore, the same parameter settings have been determined, where they have been confirmed as the best selected parameters in [23].

So, the filter size with the number of filters in the convolution layer have been found to be 15×15 pixels and 10 filters, respectively. In the pooling layer, the pooling type has been assigned to maximum numbers, where these numbers reserve the extracted features by the ReLU function. Here, the pooling size is equal to 5×5 pixels, this means that this layer returned the maximum values in the regions of height 5 pixels and width 5 pixels. The stride here is equal to 5 steps of pixels. So, there is no overlap between the pooled regions.

VA Values: The number of the successfully verified subjects has been counted and recorded for various verification methods based on the FT characteristic. Consequently, the successful rates have been calculated and benchmarked as VA values. Table 6 highlights a comparison between the outcomes of the different recognition methods. It can be observed that each of the prior FT studies used a feature extraction approach, non-overlapped windowing of 5×5 pixels, efficient statistical analysis computations called Coefficient of Variance (COV) and an effective classifier. On the other hand, our approach has used only one efficient network that can obtain better performances over other work.

As it can be seen from Table 6 that the VA has been computed for the verification method in [8] and it attained 98.53%. In that paper, large ROI sizes of the FTs and only four fingers in each hand image were applied for the PolyU2D database. Furthermore, a feature extraction called the IFE was exploited and a multiple classifier known as the PNN was utilized. The main drawback in this work is that the suggested IFE still weak in extracting the FT features as it focused on increasing the contrast between only the top-hat and bottom-hat values of the FT. This seems not sufficient to analyse the full FT features.

As suggested in [6], the recognition method has achieved best VA values of 99.32%, 97.97% and 95% for the PolyU2D, IITD and CASIA-BLU databases, respectively. The employed feature extraction was the one that was proposed in [31], which is termed Local Line Binary Pattern (LLBP). This feature extraction method is based on analysing the vertical and horizontal patterns, then combining the resulted values by applying the amplitude equation. The best line lengths were reported for N=13,15,17 or 19 pixels after examining the different lengths. The recognition method was employed for three databases as in [6] and it was cited in the same study that the most challenging database in terms of finger segmentation is the IITD as it involves various finger views. The best performances were obtained by the lengths N=13(and 17), N=13, and N=19 for the PolyU2D, IIT Delhi and CASIA-BLU databases, respectively. It appears that the confusion of selecting the appropriate length in the LLBP is one of the main drawbacks here, because the experimental results were repeated multiple times to cover all the possible analysing length. Whereas, in our work we focused on establishing one optimal network that can be applied to different FT databases (or to FT images under various circumstances).

The verification method in [5] has recorded VA values of 99.66%, 98.65% and 97% for the PolyU2D, IITD and CASIA-BLU databases, respectively. Originally, the Enhanced Local Line Binary Pattern (ELLBP) has been applied as a feature extraction for the length of (N=17) pixels in vertical and horizontal directions. It can be observed that the VA values have been noticeably increased by using the ELLBP feature extraction compared with the VA values which have been reported by utilizing the LLBP feature extraction. Incorporating the FTs of five fingers altogether for one subject in this study caused increasing the architecture of the applied neural network and this is the same problem in [6].

According to [4] a new feature extraction method termed the MSALBP was proposed. Furthermore, an innovation classifier named the FCFNN was approached. In the same publication, these two approaches were used for two databases: the PolyU2D and the CASIA-BLU. The VAs have been benchmarked for two cases following [4]: exploiting the MSALBP with the PNN and employing the MSALBP with the FCFNN. After applying the first case, the VAs have achieved 99.32% and 98% for the PolyU2D and the CASIA-BLU databases, respectively. After applying the second case, the VAs have obtained 99.77% and 98% for the same two databases, respectively. The suggested MSALBP with the PNN/FCFNN are significantly faster than other previous methods as it will be explained in the time comparison subsection. The main drawbacks in this work are that the MSALBP use fusions between multiple operations, the proposed FCFNN has a very large architecture, and only two databases were exploited.

The achieved DFTL results show surpassing and superior performance compared with above aforementioned publications. The results have been confirmed by using four types of databases. These are the PolyU2D, IITD, CASIA-BLU and CASIA-WHT. It is clear that our DFTL approach has recorded such interesting VA values compared with the previous calculated VAs. To clarify, the percentages of the VA have attained 100% for both the PolyU2D and CASIA-BLU databases. This can be considered as remarkable outcomes in the case of personal verification based on the FTs. Expectedly, the VA has achieved 98.65% for the IITD database as this database includes challenging FT images. To illustrate, some extracted FT images have been affected by the different hand postures and the existed finger rings [6]. Nevertheless, this is the best VA benchmarked value for this database. Additional database which has been employed in this paper is the CASIA-WHT database. The VA of this database has also reported a valuable result of 98%. To summarize, the proposed DFTL approach provides superior VA values over state-of-the-art work. Furthermore, the number of exploited FT databases are more than what has been employed before and the

Table 3 The obtained VAs by changing the convolution layer filter size of the DFTL (50 epochs were determined in all trainings)

PolyU2D database

Filter size	VA(%)	Mini-batch Accuracy(%) (From last iteration)
5×5	91.53	83.59
7×7	93.90	77.34
9×9	98.31	92.97
11×11	98.87	94.53
13×13	99.66	99.22
15×15	98.19	96.88
17×17	97.97	97.66
19×19	98.42	97.66

IITD database

Filter size	VA(%)	Mini-batch Accuracy(%) (From last iteration)
5×5	93.24	87.50
7×7	93.24	93.75
9×9	97.30	92.19
11×11	97.97	98.44
13×13	97.30	96.88
15×15	98.65	97.66
17×17	97.30	98.44
19×19	95.95	100.00

CASIA-BLU database

Filter size	VA(%)	Mini-batch Accuracy(%) (From last iteration)
5×5	79	66.41
7×7	74	55.47
9×9	77	60.94
11×11	93	83.59
13×13	85	69.53
15×15	95	85.16
17×17	88	71.88
19×19	91	71.88

CASIA-WHT Database

Filter size	VA(%)	Mini-batch Accuracy(%) (From last iteration)
5×5	84	63.28
7×7	77	49.22
9×9	81	62.50
11×11	93	84.38
13×13	95	80.47
15×15	97	86.72
17×17	91	82.81
19×19	93	82.81

Table 4 The achieved VAs by tuning the number of filters in the DFTL convolution layer (a filter size of 15×15 was determined in all trainings)

PolyU2D database

Filter size	VA(%)
4	99.44
6	99.10
8	100
10	100

IITD database

Filter size	VA(%)
4	97.97
6	97.97
8	98.65
10	98.65

CASIA-BLU database

Filter size	VA(%)
4	93
6	97
8	96
10	100

CASIA-WHT Database

Filter size	VA(%)
4	96
6	98
8	97
10	98

Table 5 Examining the pooling type of the DFTL pooling layer

Databases	VAs(%) of Each Pooling Layer Type	
	Average	Maximum
PolyU2D database	99.89	100
IITD database	98.65	98.65
CASIA-BLU database	95	100
CASIA-WHT database	97	98

Table 6 The VAs of different human verification methods by using the FT biometric characteristic

Reference	FT Recognition Process				VA of Each Database			
	Feature Extraction	*Windowing (Pixels)*	*Statistical Calculations*	*Classifier*	*PolyU2D*	*IITD*	*CASIA -BLU*	*CASIA -WHT*
[8]	IFE	5×5	COV	PNN	98.53%	—	—	—
[6]	LLBP [31]	5×5	COV	PNN	99.32%	97.97%	95%	—
[5]	ELLBP (N=17)	5×5	COV	PNN	99.66%	98.65%	97%	—
[4]	MSALBP (P=8,R=2)	5×5	COV	PNN	99.32%	—	98%	—
				FCFNN	99.77%	—	98%	—
Our Approach		**DFTL**			**100%**	**98.65%**	**100%**	**98%**

results have been improved by almost all the databases. This means that the DFTL can be utilized for FT images under different circumstances. It seems that the embedded processes within the DFTL such as the feature extraction, reduction operation and classification can produce valuable FT analysis procedure. Moreover, the DFTL has an acceptable architecture and it does not use any fusion operation.

Recognition Performances: Traditionally, the Receiver Operating Characteristic (ROC) and Detection Error Tradeoff (DET) curves have widely been used to report the recognition performances. Two main problems arise in depicting these curves here. Firstly, the output values of the DFTL in the output layer, which is the classification decision layer, are always logical (zeros and one) because this layer uses the winner-takes-all rule. Secondly, a DFTL is considered as a multi-class classifier, because it has more than one output class. Therefore, generating the ROC and DET graphs from the DFTL are not a straightforward task. To solve these problems a novel method that has been proposed in [7] and exploited in [4] is adopted.

The key idea of this method is to extract the score values from each single class of the DFTL. These values can be found in the SoftMax layer, where the nodes of this layer hold the actual output values. However, these values have been remapped or recalculated according to their relationships with the class targets. This method seems more efficient than using the logical output values from the classification decision layer. Hence, the ROC and the DET parameters are collected for each single class, which are the False Acceptance Rate (FAR); False Rejection Rate (FRR) and True Positive Rate (TPR). Each parameter was fused or combined to all single classes by using the average operation. Afterwards, beneficial demonstrated ROC and the DET curves have been produced. Figs. 6 and 7 show the ROC and the DET graphs.

Complexity: The complexity of the DFTL is concentrated in main three layers: the convolution layer, pooling layer and fully connected layer. The operations in the remaining layers have direct calculations. That is the operations in the ReLU, Softmax and classification layers can be directly applied to their previous main layers according to Equations (2), (5) and (6), respectively. This can further reduce the complexity of the DFTL. Hence, the complexity of the convolution and pooling layers are represented by $O(IJK2F)$, where I is the first dimension of the FT input image J is the second dimension of the FT input image, $K \times K$ is the kernel size of each filter and F is the number of filters. The complexity of the classification layer of the DFTL is represented by $O(NM)$, where N is the number of input nodes (it is equal to the number of last generated feature maps multiplied by the size of the feature map) and M is the number of output nodes.

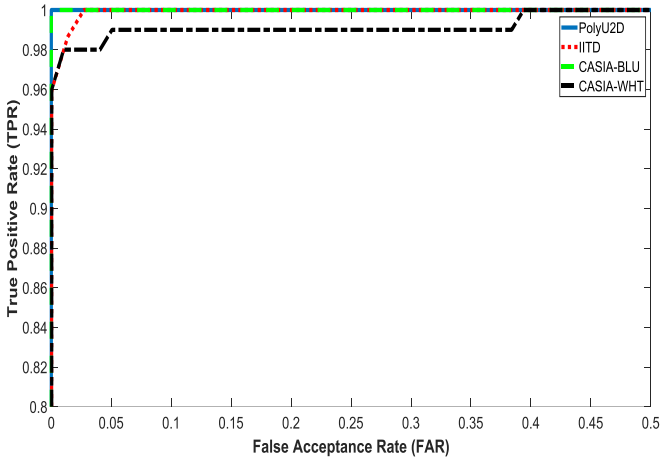

Fig. 6: The ROC curves of the DFTL for the different utilized databases

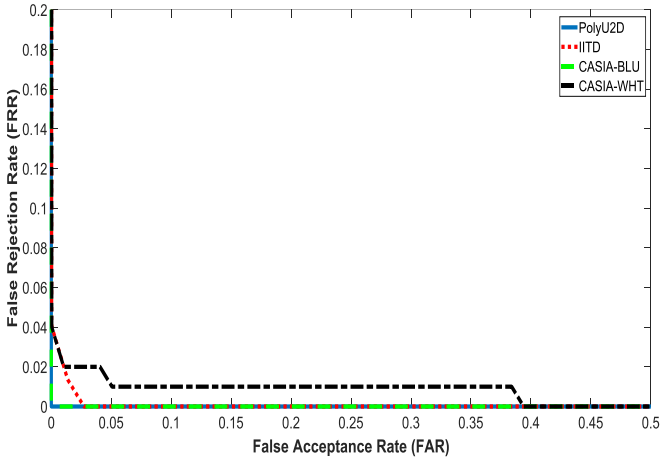

Fig. 7: The DET curves of the DFTL for the various employed databases

Table 7 Time comparison table between different human verification methods based on the FT characteristic

Reference	FT Recognition Process				Times
	Feature Extraction	***Windowing (Pixels)***	***Statistical Calculations***	***Classifier***	**(sec. per person)**
[8]	IFE	5×5	COV	PNN	0.166
[6]	LLBP [31] (N=13)	5×5	COV	PNN	0.366
[5]	ELLBP (N=17)	5×5	COV	PNN	0.366
[4]	MSALBP (P=8, R=2)	5×5	COV	PNN	0.076
				FCFNN	0.082
Our Approach		**DFTL**			**0.019**

Additional Benefits: In addition to all of the discussed VA values, it can be also noticed that the DFTL network reduces all the previous operations of the feature extraction, windowing, statistical calculations and classifier. So, instead of using four operations, only one DFTL network can be implemented for the verification. Moreover, both the PNN and the FCFNN classifiers were used five FTs from the five fingers. On the other hand, the DFTL processes small FT image size at a time and any FT can confirm the verification claim.

Finally, the DFTL can be trained for a very large number of FT images. Whilst, the PNN and the FCFNN classifiers are restricted to the provided memory size. Essentially, the architecture of the PNN classifier expands if the training samples increase [32] and the architecture of the FCFNN classifier has the same attribute [4].

4.3 Verification Time Comparisons

The testing times of the individual verification based on the FT characteristics were examined for all of the adopted methods. The time comparisons were established on a desktop computer of the following specifications: 3.2 GHz, Intel Core i5 processor and 8 GB RAM. All implementations were carried out by using the microprocessor only. The computed times are given in Table 7 and they are recorded in (second per person). From this table, it can be seen that the human recognition was approximately taken 0.166 second to complete the FT operations according to the verification method in [8]. It can be also found that the time of the individual recognition was approximately equal to 0.366 second for the FT operations as in [5]. Similarly, the time is recorded to around 0.366 second for the FT processes as in [6]. The length of the LLBP here is selected to be N=13 pixels. Because this length has quicker implementation time than other utilized lengths [23] and this length was achieved the most successful results according to [6]. Significant improvements of verification times can be observed for the operations in [4]. This is due to the proposed MSALBP feature extraction method as it was highlighted in the same publication that the MSALBP reported a very quick execution time. Also, it can be investigated that utilizing the FCFNN classifier slightly obtains slower execution time than using the PNN classifier. Finally, it is obvious that the DFTL has recorded a remarkable time enhancement as its implementation time was equal to around 0.019 second. Again, the suggested DFTL has proved its capability as an effective and efficient human verification method.

Finally, the proposed method is superior to other methods because:

- The embedded feature extraction within the DFTL is very effective. Extracting the full FT features are basically co-operated by three layers: convolution, ReLU and pooling.
- The time of human verification based on the DFTL is significantly enhanced compared to the state-of-the-art approaches. This is because that the previous approaches used multiple FT processes and utilized large neural networks architectures.
- The deep learning technology can usually provide better performances than using combinations between different processes such as extracting features, windowing, applying computations and performing matchings [6]. Principally, increasing the processing steps may increase the drawbacks of the system, because each stage has specific limitations and specifications [4]. So, using the FT characteristic with the deep learning as a single package can surpassing other complicated methods.

5 Conclusion

In this paper, an efficient individual verification method based on a deep neural network termed the DFTL has been proposed. It has superior performance than recent studies. Fundamentally, all of the previous FT characteristic work had to apply multiple operations in order to provide perceptible FT performance. So, in prior studies complicated biometric systems were usually suggested. On the other hand, the proposed DFTL has the following facilities:

◇ Due to the fact that the FT patterns are simple and reliable, the suggested DFTL has a simple deep learning architecture.

◇ It reduces the number of biometric system recognition operations such as the feature extraction, statistical calculations and classifier.

◇ It uses smaller memory size than other utilized neural networks such as the PNN, if a big number of users participates.

◇ Furthermore, the DFTL can train and test a very large number of FT images.

◇ It has reported the shortest testing time against other recent methods.

We have examined the proposed DFTL by employing four databases. These are the PolyU2D, IITD, CASIA-BLU and CASIA-WHT where it obtained remarkable VA results of 100%, 98.65%, 100% and 98%, respectively.

Acknowledgment

- "RC grant EP/P015387/1".

- "The Hong Kong Polytechnic University Contact-free 3D/2D Hand Images Database version 1.0".

- "IIT Delhi Palmprint Image Database version 1.0".

- "Portions of the research in this paper use the CASIA-MS-PalmprintV1 collected by the Chinese Academy of Sciences' Institute of Automation (CASIA) ".

6 References

[1] Liu, F., Liu, H., Gao, L., 'Hand recognition based on finger-contour and PSO', Proceedings of International Conf. on Intell. Comput. and Internet of Things, 2015, pp. 35-39

[2] Lu, Y., Yoon, S., Park, D. S., 'Finger vein identification system using two cameras', Electron. Letters, 2014, 50, (22), pp. 1591-1593

[3] Kumar, A., 'Importance of Being Unique From Finger Dorsal Patterns: Exploring Minor Finger Knuckle Patterns in Verifying Human Identities', IEEE Tran. on Info. Forensics and Security, 2014, 9, (8), pp. 1288-1298

[4] Al-Nima, R.R.O., Abdullah, M.A.M., Al-Kaltakchi, M.T.S., et al., 'Finger texture biometric verification exploiting multi-scale sobel angles local binary pattern features and score-based fusion', Elsevier, Digit. Signal Process., 2017, 70, pp. 178-189

[5] Al-Nima, R.R.O., Dlay, S.S., Al-Sumaidaee, S.A.M., et al., 'Robust feature extraction and salvage schemes for finger texture based biometrics', IET Biometrics, 2017, 6, (2), pp. 43-52

[6] Al-Nima, R.R.O., Dlay, S.S., Woo, W.L., et al., 'Efficient finger segmentation robust to hand alignment in imaging with application to human verification', 5th IEEE Int. Workshop on Biometrics and Forensics (IWBF), 2017, pp. 1-6

[7] Al-Nima, R.R.O., Dlay, S.S., Woo, W.L., et al., 'A novel biometric approach to generate ROC curve from the Probabilistic Neural Network', 24th IEEE Signal Process. and Commun. Appl. Conf. (SIU), 2016, pp. 141-144

[8] Al-Nima, R.R.O., Dlay, S.S., Woo, W.L., *et al.*, 'Human Authentication With Finger Textures Based on Image Feature Enhancement'. The Second IET Int. Conf. Intell. Signal Process. (ISP), 2015

[9] Bhaskar, B., Veluchamy, S., 'Hand based multibiometric authentication using local feature extraction'. Int. Conf. Recent Trends Inf. Technol. (ICRTIT), 2014, pp. 1-5

[10] Michael, G.K.O., Connie, T., Jin, A.T.B., 'Robust palm print and knuckle print recognition system using a contactless approach'. 5th IEEE Conf. Ind. Electron. Appl. (ICIEA), 2010, pp. 323-329

[11] Ribaric, S., Fratric, I., 'A biometric identification system based on eigenpalm and eigenfinger features', IEEE Trans. Pattern Anal. Mach. Intell., 2005, 27, (11), pp. 1698-1709

[12] Pavesic, N., Ribaric, S., Grad, B., 'Finger-based personal authentication: a comparison of feature-extraction methods based on principal component analysis, most discriminant features and regularised-direct linear discriminant analysis', IET Signal Process., 2009, 3, (4), pp. 269-281

[13] Kanhangad, V., Kumar, A., Zhang, D., 'A Unified Framework for Contactless Hand Verification', IEEE Trans. Inf. Forensics Security, 2011, 6, (3), pp. 1014-1027

[14] Ferrer, M.A., Morales, A., Travieso, C.M., Alonso, J.B., 'Low Cost Multimodal Biometric identification System Based on Hand Geometry, Palm and Finger Print Texture'. 41st Annual IEEE Int. Carnahan Conf. on Security Technol., 2007, pp. 52-58

[15] Michael, G.K.O., Connie, T., Jin, A.T.B., 'An innovative contactless palm print and knuckle print recognition system', Pattern Recognition Lett., 2010, 31, (12), pp. 1708-1719

[16] Yahyatabar, M.E., Ghasemi, J., 'Online signature verification using double-stage feature extraction modelled by dynamic feature stability experiment', IET Biometrics, 2017, 6, (6), pp. 393-401

[17] Cintas, C., Quinto-SÃąnchez, M., AcuÃśa, V, et al., 'Automatic ear detection and feature extraction using Geometric Morphometrics and convolutional neural networks', IET Biometrics, 2017, 6, (3), pp. 211-223

[18] SimÃşn, M.O., Corneanu, C., Nasrollahi, K., et al., 'Improved RGB-D-T based face recognition', IET Biometrics, 2016, 5, (4), pp. 297-303

[19] Simo-Serra, Edgar, Iizuka, Satoshi, Sasaki, Kazuma, et al., 'Learning to simplify: fully convolutional networks for rough sketch cleanup', ACM Trans. on Graphics (TOG), , 2016, 35, (4), pp. 121

[20] Krizhevsky, Alex, Sutskever, Ilya, Hinton, Geoffrey E., 'Imagenet classification with deep convolutional neural networks', Adv. in neural info. process. syst., 2012, pp. 1097-1105

[21] Wu, Jianxin, 'Introduction to convolutional neural networks', National Key Lab for Novel Softw. Technol., Nanjing University, China, 2017

[22] Stutz, David, 'Neural Codes for Image Retrieval', Proc. of the Comput. Vision-ECCV, 2014, pp. 584-599

[23] Al-Nima, R.R.O., 'Signal Processing and Machine Learning Techniques for Human Verification Based on Finger Textures', PhD thesis, School of Engineering, Newcastle University, 2017

[24] 'The Hong Kong Polytechnic University Contact-free 3D/2D Hand Images Database version 1.0', http://www.comp.polyu.edu.hk/ csajaykr/myhome/database_request/3dhand/Hand3D.htm, Online Database

[25] 'IIT Delhi Palmprint Image Database version 1.0', http://www4.comp.polyu.edu.hk/ csajaykr/IITD/Database_Palm.htm, Online Database

[26] Kumar, Ajay, 'Incorporating cohort information for reliable palmprint authentication', 6th IEEE Indian Conf. on Comput. Vision, Graphics & Image Process., ICVGIP'08., 2008, pp. 583-590

[27] 'CASIA-MS-PalmprintV1', http://biometrics.idealtest.org/, Online Database

[28] Khan, Z., Mian, A., Hu, Y., 'Contour Code: Robust and efficient multispectral palmprint encoding for human recognition'. IEEE Int. Conf. Comput. Vision (ICCV), 2011, pp. 1935-1942

[29] Khan, Zohaib, Shafait, Faisal, Hu, Yiqun, et al., 'Multispectral palmprint encoding and recognition', arXiv preprint

arXiv:1402.2941, 2014

[30] Beale, Mark H., Hagan, Martin T., Demuth, Howard B., 'Neural Network Toolbox User's Guide', The Mathworks Inc, 1992

[31] Petpon, A., Srisuk, S., 'Face Recognition with Local Line Binary Pattern', 5th Int. Conf. on Image and Graphics (ICIG), 2009, pp. 533-539

[32] Shorrock, S., Yannopoulos, A., Dlay, S.S., *et al.*, 'Biometric verification of computer users with probabilistic and cascade forward neural networks', Adv. in Physics, Electron. and Signal Process. Appl., 2000, pp. 267-272