IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received May 8, 2019, accepted May 30, 2019, date of publication June 3, 2019, date of current version June 12, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2920391

Deep Residual Network-Based Recognition of Finger Wrinkles Using Smartphone Camera

CHAN SIK KIM, NAM SUN CHO, AND KANG RYOUNG PARK
Division of Electronics and Electrical Engineering, Dongguk University, Seoul 04620, South Korea

Corresponding author: Kang Ryoung Park (parkgr@dongguk.edu)

This work was supported in part by the Basic Science Research Program of the National Research Foundation of Korea (NRF) funded by the Ministry of Education under Grant NRF-2018R1D1A1B07041921, in part by the Bio and Medical Technology Development Program of the NRF funded by the Korean Government, Ministry of Science and ICT (MSIT), under Grant NRF-2016M3A9E1915855, and in part by the National Research Foundation of Korea (NRF) Grant funded by the Korea Government, MSIT, under Grant NRF-2017R1C1B5074062.

ABSTRACT Iris, fingerprint, and three-dimensional face recognition technologies used in mobile devices face obstacles owing to price and size restrictions by additional cameras, lighting, and sensors. As an alternative, two-dimensional face recognition based on the built-in visible-light camera of mobile devices has been widely used. However, face recognition performance is greatly influenced by the factors, such as facial expression, illumination, and pose changes. Considering these limitations, researchers have studied palmprint, touchless fingerprint, and finger-knuckle-print recognition using the built-in visible light camera. However, these techniques reduce user convenience because of the difficulty in positioning a palm or fingers on the camera. To consider these issues, we propose a biometric system based on a finger-wrinkle image acquired by the visible-light camera of a smartphone. A deep residual network is used to address the degradation of recognition performance caused by misalignment and illumination variation occurring during image acquisition. Owing to the unavailability of the finger-wrinkle open database obtained by smartphone camera, we built the Dongguk finger-wrinkle database, including the images from 33 people. The results show that the recognition performance by our method exceeds in those of conventional methods.

INDEX TERMS Biometrics, finger-wrinkle recognition, smartphone camera, deep residual network.

I. INTRODUCTION

Our modern lives have become intertwined with mobile devices and the internet. Most of our personal information is stored on internet devices, used for various purposes, from accessing bank accounts to printing government-issued documents. Unfortunately, using a password to log into services exposes our assets and information to a higher risk of theft than security systems based on biometric technology. To avoid forgetting passwords, users often use autocomplete or write down the password in another location, thereby weakening security. Biometrics completely avoids such problems, because the information is always encoded in the user's body. Biometrics has gained substantial popularity in industrial and academic research, because it ensures both user convenience and stability. Recent mobile devices have implemented biometric features, such as face, fingerprint, iris, vein, and voice recognition. However, these methods generally increase both the size and price of devices because of additional illumination, sensors, special cameras, etc. Methods, such as face, touchless fingerprint, finger-knuckle, and palmprint recognition using the device's built-in camera, have been introduced to address these problems. However, these methods not only lower user convenience, they also suffer from lower recognition performance because of factors such as pose changes and misalignments. Recently released smartphones have implemented fingerprint recognition [2] or palm-vein recognition [1], both employing separate near-infrared light illumination and a camera with ultrasonic sensors embedded in the display. Because of the cost, size, and inconvenience of extant biometric smartphone technologies, this study proposes a finger-wrinkle recognition method based on a convolutional neural network (CNN) that accounts for user convenience and addresses the limitations of biometric methods used in mobile devices.

This paper is organized as follows. Section II compares and analyzes the advantages and disadvantages of various existing hand-texture-based recognition methods.

The associate editor coordinating the review of this manuscript and approving it for publication was Lefei Zhang.

2169-3536 © 2019 IEEE. Translations and content mining are permitted for academic research only.
Personal use is also permitted, but republication/redistribution requires IEEE permission.
See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

Section III describes the contributions of this study, and Section IV details the proposed method. Section V describes the experimental results and analysis, and Section VI concludes the study.

II. RELATED WORKS

Conventional hand texture-based biometrics have been used for fingerprint recognition, finger-knuckle-print recognition using the inherent skin patterns of the outer surface around the phalangeal joints, palmprint recognition using the details of raised areas (ridges) and branches (bifurcations) on a human palm, and finger-wrinkle recognition. The most studied and commercially used method is fingerprint recognition [3], [4]. AlShehri et al. analyzed the characteristics of fingerprints acquired by different sensors, where cross-sensor matching is a challenging problem [5]. Engelsma et al. proposed the design and manufacturing of high-fidelity universal 3 dimensional fingerprint targets, which can be imaged by various fingerprint sensing methods, such as capacitive, contact optical, and contactless optical sensing [6].

Zhang *et al.* [8] conducted a study on finger-knuckle-print recognition. The researchers used directional information extracted from finger-knuckle images via Gabor filtering as local features. Additionally, the Fourier transform coefficients were used as global features. The recognition accuracy was improved using the obtained local and global information. A study by [9] proposed a method to encode local patterns using the Riesz transform to extract local-image features. The study proposed two coding schema based on the Riesz transform, including RCode2, which required much less time during the feature extraction stage, lending itself to be used in real-time systems. Researchers have also studied several finger-knuckle-print matching methods to address problems experienced by users with difficult skin conditions [10]–[13]. A study by [10] confirmed the plausibility of finger-knuckle-print recognition based on band-limited phase-only correlation (BLPOC). Kusanagi et al. used a BLPOC-based local block-matching method for the finger-knuckle-print pattern of metacarpophalangeal joints [13]. Kumar et al. obtained contour regions from finger-knuckle images and extracted their geometrical features. They reduced finger misalignment via detection along the valley points. They applied linear discriminant analysis, principal component analysis (PCA), and independent component analysis (ICA) to the finger-knuckle region of interest (ROI). They then fused the calculated matching scores to perform recognition [14].

Researchers have also actively studied palmprint recognition [15]–[25]. Connie et al. segmented a palmprint image in the background using a preprocessing module that automatically aligned the palmprint image and extracted the central ROI of the palm for recognition. Then, they compared PCA, Fisher discriminant analysis, and ICA, three different subspace projection methods. They also used wavelet transform to analyze the palmprint images in a multi-resolution multi-frequency representation [19]. Jain et al. used a fixed-length minutia descriptor, MinutiaCode, to capture the unique information around each minutia and used an alignment-based minutia-matching algorithm to perform palmprint recognition [20]. Priya et al. proposed a high resolution palmprint authenticating system based on level 3 pore feature [21]. A study by [22] proposed the palmprint recognition method based on binary wavelet transform and local binary pattern (LBP). Gumaei et al. proposed a method of palmprint recognition using visible and multispectral sensor images based on histogram of oriented gradients (HOG) with a steerable Gaussian filter (SGF) [23]. Kumar et al. proposed an approach for matching contactless palmprint images using accurate deformation alignment and matching [24]. Cheng et al. proposed a robust L2 sparse representation with tensor-based extreme learning machine (RL2SR-TELM) algorithm by using an adaptive image level fusion strategy to accomplish the multispectral palmprint recognition [25].

However, to apply the above-described fingerprint recognition methods on a mobile device, a separate fingerprint image acquisition sensor based on capacitance or ultrasonics must be attached. This increases the size and price of the mobile device. Otherwise, we could consider the above-described finger-knuckle-print or palmprint recognition methods for use with the built-in visible light camera of a mobile device. However, the difficulty of positioning the palm or the outside of the finger on the frontal viewing camera of the mobile device reduces user convenience. For the palm, it is inconvenient to continuously raise a hand to the frontal viewing camera to acquire a palm image. Likewise, for acquiring a finger-knuckle image, it is inconvenient to turn one's hand to make the outer surface visible to the frontal viewing camera. When acquiring an image of the user's face, the frontal viewing camera is mainly used for image acquisition, because the user cannot not see the screen when positioning his or her palm or hand. Therefore, researchers have investigated finger-wrinkle (i.e., inner knuckle-print) recognition, which boasts higher user convenience and can acquire an image of the user's finger even if the hand is not turned but only slightly raised to the frontal viewing camera. Previously, Liu et al. combined Gabor filtering and the derivative line detection method to extract line features. They then used a cross-correlation-based method to match binary line images [7]. However, their study did not use images acquired from a mobile device, and there was almost no misalignment between the acquired images, because they used finger guide bars, which are difficult to apply with mobile devices. Moreover, accuracy improvement is limited when using handcrafted features.

Although it does not belong to the category of hand texture-based biometrics, in previous research [44], authors proposed a novel descriptor estimator-based incipient fault estimation method designed for Lipschitz nonlinear descriptor systems with process disturbances and measurement output noises. By using this estimator, incipient sensor faults, abrupt actuator faults, and measurement noises could be estimated asymptotically. Although it does not belong to the category of

TABLE 1. Comparisons of proposed and previous studies on hand texture-based biometrics.

Category	Methods	Advantage	Drawback
Fingerprint	Minutiae triplets [4], electrostatic fingerprint imaging [3], cross-sensor matching [5], and 3 dimensional fingerprint recognition [6]	Sensor size is small and price is low	- Affected by skin state, such as moisture and injuries - User friendliness is lower than other methods because of direct contact - Separate sensor is attached for use in mobile devices.
Finger-knuckle-print	Local and global feature [8], Riesz transform [9], and BLPOC [8,10,13,26] and subspace [14]	- Less affected than finger print recognition - Can acquire images using built-in camera of mobile device without a separate sensor	- More susceptible to skin deformities and finger movements - Poor user convenience when acquiring images with the mobile-device camera
Palmprint	3 dimensional palmprint [15], LBP [16], ridge-based image quality measurement [17], subspace projection [19], MinutiaCode [20], pore feature [21], binary wavelet transform and LBP [22], competitive coding scheme [23], contactless palmprint recognition [24], RL2SR-TELM [25], and fusion of 2 dimensional and 3 dimensional palmprint recognitions [18]	- Because the recognition region is large, compared to fingerprint recognition, more distinctive features can be obtained, which is more suitable for the recognition system. - Can acquire images using the built-in camera of the mobile device without a separate sensor	- System becomes large and expensive because a larger region must be captured compared to fingerprint scanners. - Poor user convenience when acquiring images using the mobile-device camera
Finger-wrinkle (inner knuckle-print)	Handcrafted features by Gabor filtering and derivative line detection [7]	- High user convenience when acquiring images using the mobile-device camera - Can acquire images using the built-in camera of a mobile device without a separate sensor	- Does not use images acquired from the mobile device - Almost no misalignment between the acquired images using a finger guide bar, which is difficult to apply to mobile devices - Accuracy improvement is limited when using handcrafted features
	Deep features by CNN (**proposed method**)	- High user convenience when acquiring images using the mobile-device camera - Can acquire images using the built-in camera of the mobile device without a separate sensor - High performance via deep-learning-based features for high misalignment, illumination variation, blur, and rotation	Requires intensive training for the CNN using many databases

hand texture-based biometrics, either, Kim et al. proposed multimodal biometric method based on CNN by combining both the information of finger-vein and finger shape [45].

Considering the problems of previous hand texture-based biometrics, we propose a biometric system based on the finger-wrinkle image of a user acquired with a smartphone's built-in visible-light camera. A deep residual network (ResNet) is used to address degradation in recognition performance caused by misalignment and illumination variation occurring during image acquisition. Table 1 summarizes the advantages and disadvantages of previous and proposed methods.

III. CONTRIBUTIONS

Our research is novel in the following four ways:

- This study is the first to perform finger-wrinkle recognition using the smartphone's built-in visible-light camera.
- To solve the misalignment issue between the enrolled image and the recognized image, a guide window is used in the smartphone display, which displays the image obtained from the camera. The guide window was designed to both maximize user convenience and minimize misalignment. We also compared the user convenience of the fingerprint, finger-knuckle-print, palmprint, and finger-wrinkle recognition methods with the method proposed in this study, and we experimentally compared the degrees of misalignment in the images acquired with these four methods.
- A deep ResNet-based recognition method addresses misalignment, illumination variation, blur, and degradation of recognition performance caused by rotation between the enrolled and recognized images, despite the use of a guide window. Traditional ResNet is modified by replacing its last AVG pooling layer with an additional convolutional layer.
- We made the self-collected Dongguk mobile finger-wrinkle database (DMFW-DB1) and trained ResNet model available to other researchers through [28] for fair comparisons.

IV. PROPOSED METHOD

A. OVERVIEW OF THE PROPOSED METHOD

The flowchart of the proposed method is shown in Figure 1. As shown in Figure 2, the user's four-finger image is acquired using the guide window displayed on the smartphone display and the smartphone frontal viewing camera

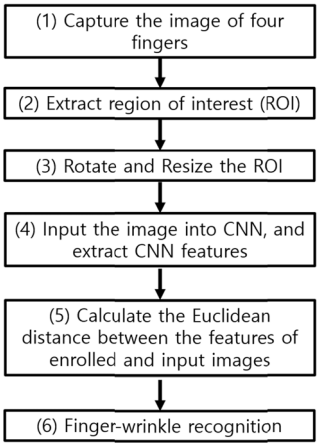

FIGURE 1. Flowchart of the proposed method.

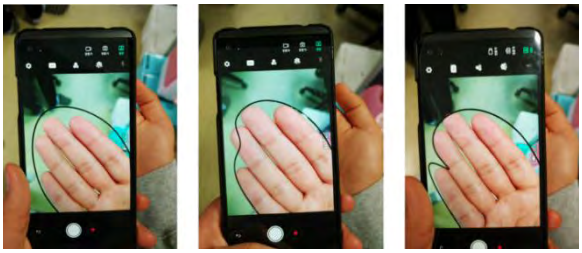

FIGURE 2. Three examples of a guide window for image acquisition.

(step (1) of Figure 1). Next, to extract the finger-wrinkle region, the designated hexagon ROI is extracted (step (1) of Figure 1). The extracted ROI is rotated and resized at a specific angle, based on the ROI center, to obtain the final ROI for the CNN input (step (3) of Figure 1). Next, the CNN uses this ROI to find the features of the finger-wrinkle (step (4) of Figure 1). Next, the Euclidean distance between the features extracted from the input image and the pre-registered image features is calculated (step (5) of Figure 1). Finally, recognition is achieved based on this distance to determine whether the input image is genuine or an imposter (step (6) of Figure 1). Here, genuine matching indicates that the input image and the pre-enrolled image are of the same class, and imposter matching indicates that the input image and the pre-enrolled image are of different classes.

B. IMAGE ACQUISITION AND PREPROCESSING

First, the user's four-finger image is captured with the smartphone's frontal viewing camera to acquire finger-wrinkle images. When capturing using a face camera, hand covers the display, making it difficult to confirm whether the screen is well focused or if the wrinkle part to be recognized is properly aligned. Moreover, it is difficult to maintain a suitable distance between the camera and the hand. Therefore, the frontal viewing camera is used. To reduce the hand-pose variation of each shot, the screen displays a rough hand-guide window, as shown in Figure 2. Based on the images

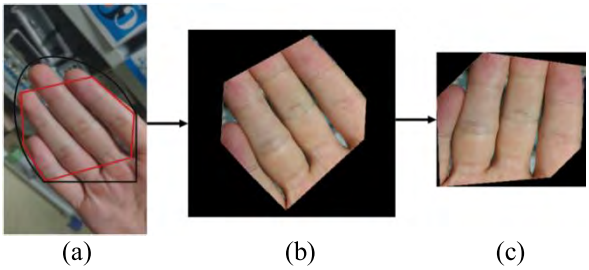

FIGURE 3. Preprocessing of extracted finger-wrinkle ROI: (a) captured finger-wrinkle image, (b) finger-wrinkle ROI, and (c) normalization by in-plane rotation.

acquired using guide windows of various shapes, we found that the guide window in the shape of (a) in Figure 2 enabled images to be captured easiest. Because the angle of the finger when acquiring the image is also important, we empirically determined the most natural and comfortable angle when the user positions a hand. To extract the finger-wrinkle region from the captured image in Figure 3(a), the ROI of the hexagon is cropped, as shown in Figure 3(b), with the six points previously given in the guide window. To insert a square image, as shown in Figure 3(c), as the CNN input (see Equation (1)), pixels $(\boldsymbol{x_{org}}, \boldsymbol{y_{org}})$, belonging to the ROI, are in-plane rotated with respect to $(\boldsymbol{C_x}, \boldsymbol{C_y})$.The empty pixels of image $(\boldsymbol{x_{new}}, \boldsymbol{y_{new}})$, subject to this rotation, are filled via bi-linear interpolation.

Here, the background region is reduced and the rotation angle ($\theta = 40°$) is used to include as much of the ROI as possible in the CNN input image. To use the image as input to the CNN, it is resized to 224 × 224 pixels via bi-linear interpolation.

$$\begin{pmatrix} x_{new} \\ y_{new} \end{pmatrix} = \begin{pmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{pmatrix} \begin{pmatrix} x_{org} - C_x \\ y_{org} - C_y \end{pmatrix} + \begin{pmatrix} C_x \\ C_y \end{pmatrix}$$
$$(x_{org}, y_{org}) \in R \text{ (Region of interest)} \quad (1)$$

C. PREPROCESSING USING RETINEX FILTERING FOR ILLUMINATION COMPENSATION

In this study, to test the robustness of the DMFW-DB1 database for illuminance variation in various environments, photographs are captured five times at different indoor illuminances for each class. This illumination variation can decrease recognition accuracy. This study, therefore, uses a Retinex filtering algorithm to reduce the illumination variation of the input images. The red-green-blue (RGB) ROI image obtained in Figure 3(c) is converted into a hue-saturation-value (HSV) color space. Among the three channels, only the value channel (illumination) is subject to Retinex filtering, as shown in Equations (2) to (5). In Equation (2), the intensity of captured image, $(L(x, y))$, is modeled by multiplying the illumination, $I_c(x, y)$, and the ratio of reflection, $(r(x, y))$ [29]. Equation (2) can be obtained from Equation (1).

$$L(x, y) = I_c(x, y) \times r(x, y) \quad (2)$$
$$logr(x, y) = logL(x, y) - logI_c(x, y) \quad (3)$$

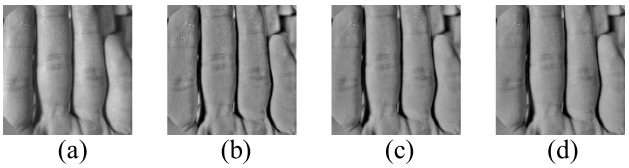

FIGURE 4. Examples of Retinex images obtained with various sigma values of Gaussian filter: (a) original image; Retinex images obtained using sigma values of (b) 10, (c) 15, and (d) 20.

The illumination ($I_c(x, y)$) is assumed to be the convolution of the Gaussian filtering ($F(x, y)$) and the image, ($L(x, y)$), as shown in Equations (3) and (4) [30].

$$logr(x, y) = logL(x, y) - log[F(x, y) * L(x, y)] \quad (4)$$

$$F(x, y) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}} \quad (5)$$

where $F(x, y)$ and $log\, r(x, y)$ respectively indicate the Gaussian filter using various sigma values ($\sigma = 10$, 15, and 20) and the result image by Retinex filtering. The finger-wrinkle ROI images subject to Retinex filtering are normalized, as shown in Figure 4.

D. CNN-BASED FINGER-WRINKLE RECOGNITION

This study is based on the pre-trained model of ResNet-101 [27], with the exception of the output node and replacement of last average (AVG) pooling layer by additional convolutional layer. This model is pre-trained using the ImageNet database, comprising millions of images, used in the ImageNet Large-scale Visual Recognition Competition (ILSVRC) [51]. Thus, image preprocessing includes resizing the training and testing images to 224 × 224 pixels, the size of ImageNet data input. This pre-trained model was used to fine-tune the training data of DMFW-DB1 built into this study. To perform finger-wrinkle recognition, the number of classes included in the training data is set to the number of CNN output nodes. Figure 5 and Table 2 show the structure of the deep ResNet model used in this study. The size of the feature map is typically calculated based on (output width (or height) = (input width (or height) – filter width (or height) + (# of padding) × 2) / (# of stride) + 1) [31]. For example, in Table 2, the feature-map width (or height) output in the first convolution layer (Conv1) is 112 ((224 – 7 + 3 × 2) / 2 + 1). Batch normalization [32] and rectified linear units (ReLU) are used to optimize the model. During batch normalization after each convolution layer, rather than obtaining the mean and variance of all training data, the values are obtained in mini-batch units and normalized. Batch normalization can address the problems of a vanishing/exploding gradient or dropping to the local minimum occurring when the learning rate is increased in the deep CNN. It can also reduce the internal covariance shift that makes the training process unstable and stabilize the overall system to accelerate training speed. The activation function is important to the training process. Although a sigmoid function [33] is commonly used, ReLU is used to alleviate the

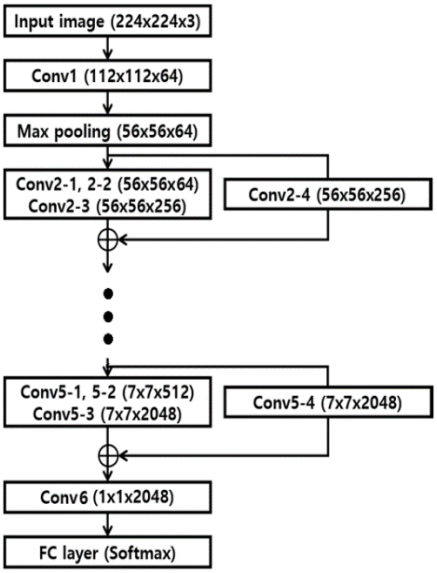

FIGURE 5. CNN structure used in our research.

vanishing-gradient problem [34] in backpropagation during training. Additionally, the learning convergence rate becomes faster than the non-linear activation function when using the ReLU function. ReLU can be expressed as follows [33], [35].

$$y = max(0, x) \quad (6)$$

where x and y are the input and output values of the ReLU function, respectively.

As shown in Table 2, a bottleneck structure is used in Conv3–Conv5 of ResNet [27]. A convolution layer comprising filters of 1×1, 3×3, and 1×1 is referred to as a bottleneck design. The first 1 × 1 convolution is intended to reduce the dimension, after which the 3 × 3 convolution is performed, and then the last 1 × 1 convolution serves to expand the dimension again. This bottleneck structure can reduce computational complexity compared to structures in which two 3 × 3 convolution layers are directly connected. The shortcut of ResNet addresses the problem of poor training caused by deepening layers, and it alleviates signal attenuation by transferring the input to the output in the residual block. After Equation (6) is processed, the backpropagation process is performed to modify the weight using the stochastic gradient descent (SGD) method, which compares the obtained output with the ground-truth value to optimize the weight of the learning model. The first calculation is the loss between the current result and the ground-truth. Based on how precisely it is calculated, it is possible to determine if they converge so that the training can complete. The basic multinomial logistic loss method is used for calculation here, and softmax is used to calculate the predicted output. Thus, better numerical stability can be maintained in the slope calculation. The following is the equation of the softmax function [35].

$$\sigma(z)_j = \frac{e^{zj}}{\sum_{k=1}^{K} e^{zk}} \quad for\ j = 1 \ldots, K \quad (7)$$

TABLE 2. Architecture of CNN used in our research. Note: N* indicates that N pixels are included as padding in the left, right, up, and down positions of an input image or feature map. 2/1** means 2 at the 1st iteration and 1 from the 2nd iteration.

Layer name		Size of feature map (height × width × channel)	# of filters	Size of filters	# of strides	# of padding	# of iterations
Image input layer		224×224×3					
Conv1		112×112×64	64	7×7×3	2	3*	1
Max pool		56×56×64	1	3×3×1	2	0	1
Conv2	Conv2-1	56×56×64	64	1×1×64	1	0	3
	Conv2-2	56×56×64	64	3×3×64	1	1*	
	Conv2-3	56×56×256	256	1×1×64	1	0	
	Conv2-4 (Shortcut)	56×56×256	256	1×1×64	1	0	
Conv3	Conv3-1	28×28×128	128	1×1×256	2/1**	0	4
	Conv3-2 (Bottleneck)	28×28×128	128	3×3×128	1	1*	
	Conv3-3	28×28×512	512	1×1×128	1	0	
	Conv3-4 (Shortcut)	28×28×512	512	1×1×256	2	0	
Conv4	Conv4-1	14×14×256	256	1×1×512	2/1**	0	23
	Conv4-2 (Bottleneck)	14×14×256	256	3×3×256	1	1*	
	Conv4-3	14×14×1024	1024	1×1×256	1	0	
	Conv4-4 (Shortcut)	14×14×1024	1024	1×1×512	2	0	
Conv5	Conv5-1	7×7×512	512	1×1×1024	2/1**	0	3
	Conv5-2 (Bottleneck)	7×7×512	512	3×3×512	1	1*	
	Conv5-3	7×7×2048	2048	1×1×512	1	0	
	Conv5-4 (Shortcut)	7×7×2048	2048	1×1×1024	2	0	
Conv6		1×1×2048	2048	7×7×2048	1	0	1
Fully connected (FC) layer		# of class					1
Softmax		# of class					1

In case that the array of output neurons is set to z, the probability of the neurons belonging to the j^{th} class is obtained by dividing the value of the j^{th} class by the sum of the values of all classes. Using these results as input, the multinomial logistic loss [52] is calculated (Equation (8)). The following is the equation for calculating loss.

$$E = -\frac{1}{N}\sum_{n=1}^{N} log(\hat{p}_n, l_n) \tag{8}$$

$\hat{P}_n$: *predicted probability*
l_n : *label* $l_n \in [0, 1, 2, \ldots, K-1]$

In Equation (8), K is the number of classes. As explained at the beginning of this section, we modified the traditional ResNet model [27] by replacing its last 7 × 7 AVG pooling layer with an additional convolutional layer of 7 × 7 × 2048 (Conv6 in Table 2 and Figure 5) and performed the fine-tuning of this revised ResNet model with our finger-wrinkle

training database. The reasons why we used $7 \times 7 \times 2048$ convolutional layer are as follows. Compared to the conventional biometrics such as fingerprint, iris, and vein recognition, the classification of our finger-wrinkle images has the problems of high inter-class similarity due to low image quality caused by illumination variation, misalignment, blur, and in-plane rotation. Therefore, more features without loss should be extracted from the CNN for the classification. The traditional ResNet [27] obtains the feature map of $1 \times 1 \times 2048$ from the previous feature map of $7 \times 7 \times 2048$ by using AVG pooling layer including one filter of 7×7, and this can lose useful features. To address this issue, the additional convolution layer (Conv6 in Table 2 and Figure 5) including 2048 filters of $7 \times 7 \times 2048$ in our revised ResNet model generates the feature map of $1 \times 1 \times 2048$ from the previous feature map of $7 \times 7 \times 2048$, and this can maintain the useful features without loss. Moreover, the filter coefficients of AVG pooling layer in traditional ResNet model are fixed ones. However, the optimal filter coefficients of the Conv6 in our revised ResNet model can be obtained by training.

V. EXPERIMENTAL RESULTS

A. EXPERIMENTAL DATABASE AND ENVIRONMENT

Because there is no smartphone-acquired finger-wrinkle open database, we built DMFW-DB1 to support the experiments of this study. The images of this database were captured using the frontal-viewing camera (8 mega-pixels ($2,160 \times 3,840$ pixels), 30 fps, auto-mode) of an LG V20 smartphone based on the guide window shown in Figure 2(a) [36]. To make the database robust at several illuminances, images of both hands of 33 people (male and female) were captured in five different indoor environments, each for $2 \sim 3$ s. A total of 33,359 images were extracted from the captured videos by frame, creating 66 classes with approximately 500 images per class. To use 2-fold cross validation (a testing-error measurement method) in supervised learning, the classes of this database were divided into two groups for training and testing. Thus, in the 1st-fold validation, 34 classes (class 1–class 34) of images (subset 1 in Table 3) were used in training, and the remaining 2nd-fold 32 classes (class 35–class 66) of images (subset 2 in Table 3) were used in testing. In contrast, in the 2nd-fold validation, 32 classes (class 35–class 66) of images (subset 2 in Table 3) were used in training, and 34 classes (class 1–class 34) of images (subset 1 in Table 3) were used in testing. Figure 6 shows an example of DMFW-DB1. Figures 6(a) and (b) show five trial images taken at different illuminances of the same person's left and right hands, and Figure 6(c) shows five trial images taken at different illuminances of another person's left hand.

This study used the rough guide window shown in Figure 2(a), which is not strict. As shown in Figure 6, there are many misalignments caused by translation and in-plane rotation in the input images, even when the same hand is captured several times. There are also various illumination variations. Therefore, we acquired and tested data, including various environmental change factors, to measure recognition performance in an unconstrained real environment rather than the constrained environment of a laboratory.

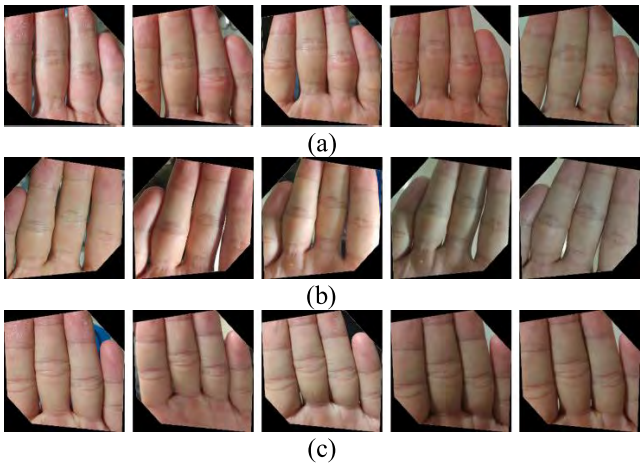

FIGURE 6. Examples of DMFW-DB1 images. Different trials of the same person: (a) image of left hand; (b) image of right hand; and (c) image of another person's left hand.

TABLE 3. Description of DMFW-DB1 for 2-fold cross validation.

DMFW-DB1	Subset 1	Subset 2
# of images	17,076	16,283
# of people	17	16
# of hands	34	32
# of classes	34	32

As described above, a mean accuracy in this study was measured using two-fold cross validation, which exchanges the images used in training and testing. Because different classes are used during training and testing, the output node values of ResNet cannot be used for recognition. Therefore, we performed distance matching for the 2,048 features of the input and enrolled images extracted from the AVG pool of Table 2 rather than the final output value of the CNN. Then, we measured recognition performance. Table 3 shows a description of the database used in this research.

In this research, the CNN training and testing was performed using a system with Intel® Core™ i7-7700 central processing unit (CPU) @ 3.60 GHz (4 cores) (Intel, Santa Clara, CA, U.S.) with 24-GB memory and an NVIDIA GeForce GTX 1070 (1920 CUDA cores, NVIDIA Corporation, Santa Clara, CA, U.S.) graphics card having a memory of 8 GB [37]. The training and testing algorithms were implemented using Windows Caffe (version 1) [38] with Microsoft Visual Studio 2013 [39], and was compute on a unified device architecture (CUDA) (version 8.0) [40] with the CUDA deep neural network library (version 5.1) [41].

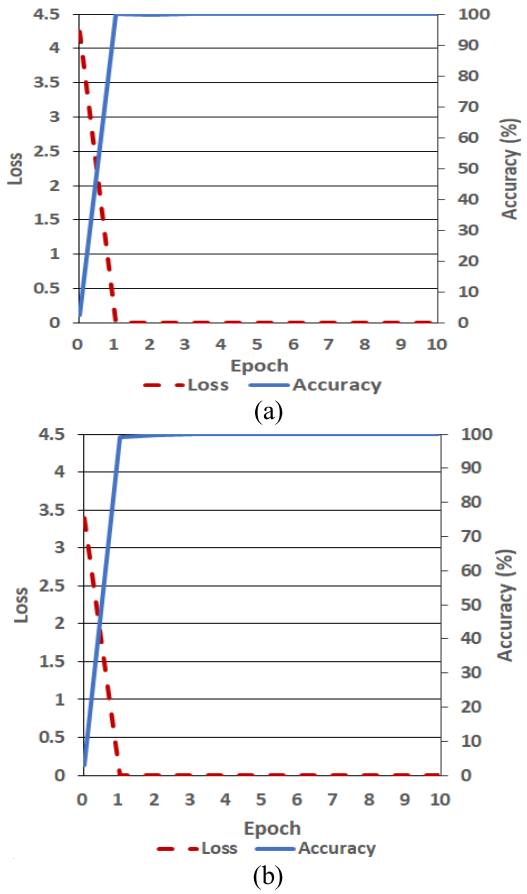

FIGURE 7. Examples of loss and accuracy curves with training data of 2-fold cross validation: (a) is from the 1st fold validation; (b) is from the 2nd fold validation.

B. TRAINING OF CNN

An SGD optimizer was used to optimize CNN training [42]. In the SGD optimizer, optimization is performed using a step policy that multiplies the gamma value for each fixed iteration so that the training accuracy and loss converge quickly. Training, a function of SGD, was performed in mini-batch size units. The number of iterations is calculated as "number of training data / mini-batch size," defined as 1 epoch. In this experiment, learning rate [32] is 0.0005, momentum is 0.9, and gamma is 0.1, whereas the mini-batch size [32] in ResNet-50 is 16, 6 in ResNet-101, 3 in ResNet-152, and 20 in visual geometry group (VGG)-16, with a maximum epoch of 10. Because fine-tuning was performed using the existing pre-trained weights, we used small learning-rate values. After seven epochs, the learning rate is reduced. One epoch indicates that training is performed as many times as the total number of iterations. Therefore, the total number of trainings is equal to the number of iterations × the number of epochs. To calculate the training loss, the softmax [35] function was used to calculate multinomial logistic loss, as in Equation (8).

Figure 7 shows the training accuracy and loss of the ResNet-101 model proposed in this study. The x-axis represents the number of epochs, and the y-axis shows the training accuracy and loss in each epoch. As shown in Figure 7, when training is performed, the accuracy converges to 100 and the loss converges to 0. This shows that the training of the CNN model used in this study was successful. We made the self-collected DMFW-DB1 and trained ResNet model available to other researchers through [28] for fair comparisons.

TABLE 4. Comparison of error rates of finger-wrinkle recognition according to the input of gray and color spaces.

Method	EER (%)
H + S + grayscale + ResNet-101	22.73
HSV + ResNet-101	16.20
Grayscale + ResNet-101	8.67
RGB color + ResNet-101 **(Proposed method)**	**5.42**

C. TESTING OF PROPOSED CNN-BASED RECOGNITION

1) COMPARISON OF FINGER-WRINKLE RECOGNITION PERFORMANCE ACCORDING TO COLOR SPACE

For the first experiment, we compared the recognition performance of input images of gray and various color spaces. As shown in Table 4, we compared performances when using the input images expressed in RGB, grayscale obtained from RGB, HSV, and H + S + grayscale obtained from RGB. For grayscale, the same image was copied to the three channels of the CNN. As shown in Table 4, recognition error is lowest when the RGB color image is used. In this study, we measured the recognition error rate using the equal error rate (EER). EER is the error rate at the point at which the false acceptance rate, the error rate of incorrectly accepting imposter data as genuine, and false rejection rate (FRR), the error rate of incorrectly rejecting genuine data as imposter, become the same.

Figure 8 shows the accuracy measured in Table 4 in more detail with the receiver operating characteristic (ROC) curve. The genuine acceptance rate is calculated as 1 – FRR. Each graph is a mean graph of the two graphs in 2-fold validations. Figure 8 shows that the best recognition performance is obtained when using the RGB color image.

2) COMPARISON OF PERFORMANCE BETWEEN RETINEX FILTERED IMAGE AND ORIGINAL IMAGE

In the following experiment, we compared the recognition performance of Retinex filtering and the original image. The σ value of Retinex filtering determines the shape of Gabor filter as shown in Equation (5), and the performance is compared with the Retinex filtered images according to various σ values, as shown in Table 5. Table 5 also shows that the recognition error is lowest when the original RGB color image is used without Retinex filtering. This occurs, because, as shown in Figure 4, applying Retinex filtering reduces not only illumination variation, but it also increases the distinctiveness of finger-wrinkle texture. Moreover, the

TABLE 5. Comparison of error rates using Retinex filtered image and original image according to various σ values.

Method	EER (%)
H + S + Retinex filtered V with σ of 10 + ResNet-101	21.61
Retinex filtered V with σ of 20 + ResNet-101	10.82
Retinex filtered V with σ of 15 + ResNet-101	10.04
Retinex filtered V with σ of 10 + ResNet-101	8.76
Retinex filtered RGB with σ of 150 + ResNet-101	7.58
RGB color without Retinex filtering + ResNet-101 **(Proposed method)**	**5.42**

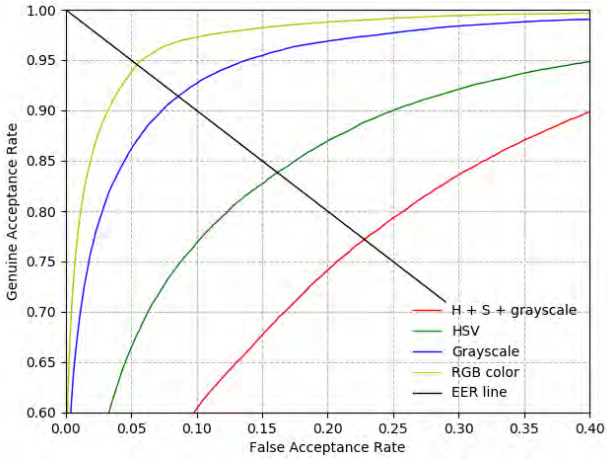

FIGURE 8. Comparative ROC curves of finger-wrinkle recognition according to the input of gray and color spaces.

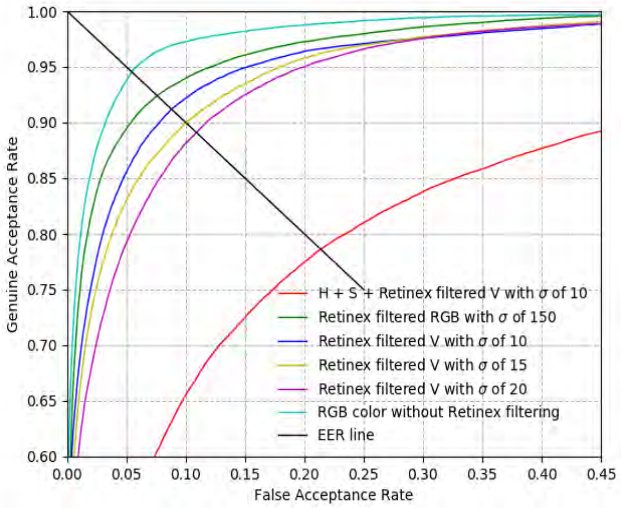

FIGURE 9. Comparative ROC curves of finger-wrinkle recognition according to the image with or without Retinex filtering.

deep ResNet used in this study can obtain sufficiently robust features to address the illumination variation.

Figure 9 shows the accuracy measured in Table 5 in more detail through ROC curves. Each graph is a mean graph of two graphs in the 2-fold validations. Figure 9 shows that the best recognition performance is obtained when the original RGB image is used without Retinex filtering.

3) PERFORMANCE COMPARISON ACCORDING TO BACKGROUND PROCESSING METHOD

In the next experiment, we measured the recognition accuracy according to the processing method of the background region, excluding ROI in Figure 3(c). Because the 224 × 224-pixel images, including ROI and background, are used as input images of the CNN, they can affect the deep features extracted from the CNN according to the background pixel values. We compared the recognition accuracy when the background is filled with values of 0 (Figure 3(c)) and when the background is filled with the mean pixel value of the ROI (Figure 10).

As shown in Table 6, the former case (EER of 5.42%) showed a higher recognition rate than the latter (EER of 8.35%). In the former case, no valid features were

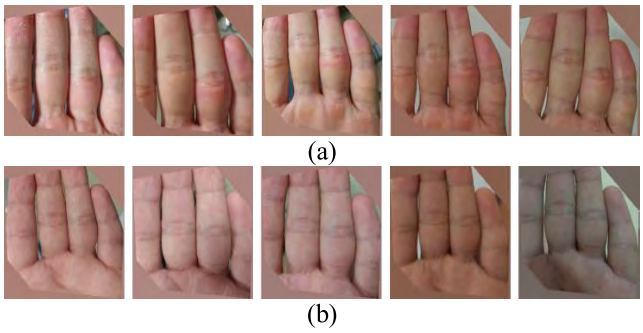

FIGURE 10. Example of other trial images from the same hand of one person, in which the background outside the ROI was filled with the mean pixel value of the ROI: (a) Trial 1 and (b) Trial 2.

automatically extracted by CNN training for the background comprising values of 0. Whereas, in the latter case, the pixel values of the background changed according to the illumination variation of the ROI, as shown in Figure 10. This is because unnecessary features adversely affecting recognition were extracted from the background.

Figure 11 shows the ROC curve for the accuracy measured in Table 6. Each graph is a mean graph of the two graphs in

TABLE 6. Comparison of error rates according to background processing method.

Method	EER (%)
Background is filled with the mean pixel value of ROI (image + ResNet-101)	8.35
Background filled with values of 0 (image + ResNet-101) **(Proposed method)**	5.42

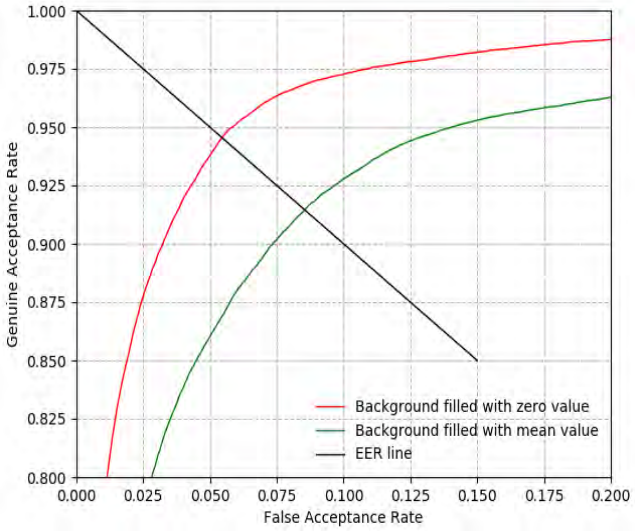

FIGURE 11. ROC curves of finger-wrinkle recognition according to the background processing method.

TABLE 7. Comparative accuracies of finger-wrinkle recognition according to various CNN networks.

Model	Features for recognition	EER (%)
VGG Net-16 [43]	Fc6	18.1
VGG Net-16 [43]	Fc7	13.29
ResNet-50	Pool5	9.584
ResNet-152	Pool5	8.16
ResNet-101 **(proposed method)**	Pool5	**5.42**

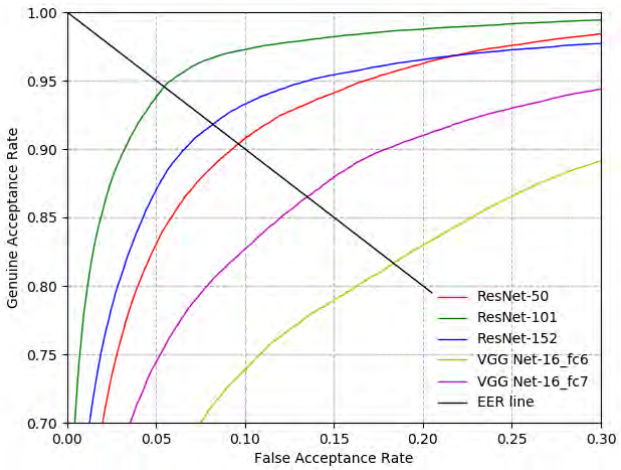

FIGURE 12. ROC curves of recognition by ResNet-50, 101, 152, and VGG-16.

the 2-fold validations. Figure 11 shows that the background filled with values of 0 shows better recognition performance.

4) PERFORMANCE COMPARISON ACCORDING TO CNN MODEL

For the next experiment, we compared the recognition performance of various CNN models to ResNet-101. The models for comparison are VGG Net-16 [43], ResNet-50, and ResNet-152 [27]. VGG Net-16 performs distance matching using features extracted from fully connected layer 6 (Fc6) and fully connected layer 7 (Fc7), which are typically used for deep features. Additionally, ResNet-50 and ResNet-152 performed distance matching using features extracted from the AVG pool (Pool 5) in Table 2 for fair evaluation.

Table 7 shows the performance according to the features extracted from different layers of the CNN model and comparison models. ResNet, a network deeper than VGG using residual connection, showed high recognition performance. ResNet-101 showed the highest recognition performance. Figure 12 shows the accuracy measured in Table 7 in more detail using ROC curves. Each graph is a mean graph of the two graphs in the 2-fold validations. Figure 12 shows that ResNet-101 has the highest recognition performance compared to the other CNN models.

TABLE 8. Comparison of finger-wrinkle recognition accuracy.

Category	Method	EER (%)
Handcrafted features	Subspace [19]	22.65
	LBP [16,22]	21.46
Deep features	ResNet-101 **(Proposed method)**	5.42

5) PERFORMANCE COMPARISON BETWEEN THE EXISTING METHODS AND PROPOSED METHOD

In the next experiment, we compared the performance of the proposed method with that of the existing method. The existing methods are based on handcrafted features. Table 8 shows that the deep feature-based method proposed in this study has higher recognition accuracy than the existing methods.

Figure 13 shows the ROC curve for the accuracy measured in Table 8. Each graph is a mean graph of the two graphs in the 2-fold validations. Figure 13 shows that recognition performance improves when utilizing the deep features of this study rather than handcrafted features.

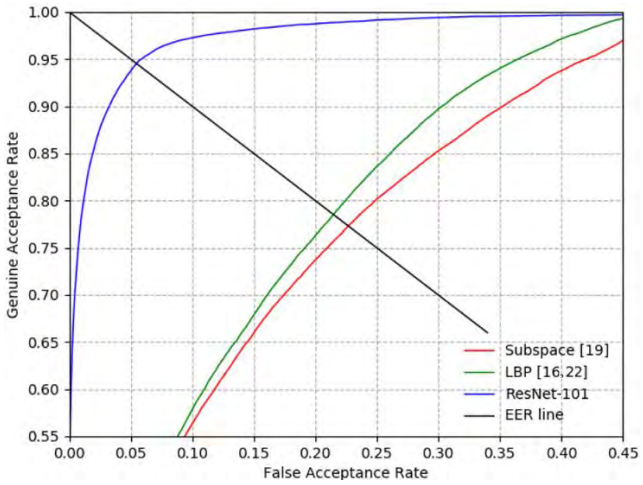

FIGURE 13. Comparative ROC curves of finger-wrinkle recognition via previous and proposed methods.

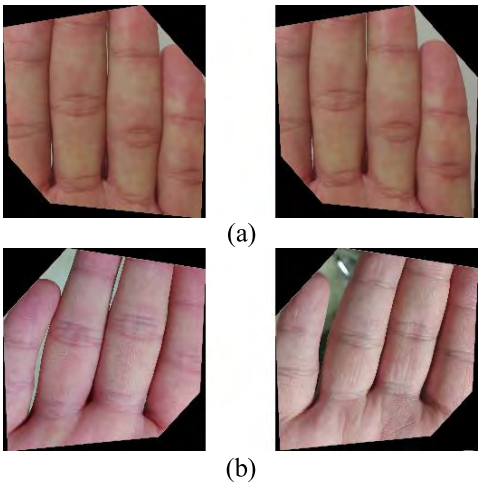

FIGURE 14. Examples of correct recognition cases: (a) authentic matching and (b) imposter matching. Left and right images in (a) and (b) show the enrolled and input images, respectively. In (a) and (b), the images are the resized ROI image (224 × 224).

Figure 14 shows an example of correct recognition (correct acceptance and correct rejection) by the proposed method. As shown in Figure 14(a), even with a variation in misalignment between the enrolled image and the input image, the proposed method returns the correct acceptance result. In Figure 14(b), even with a similar finger appearance between two images, our method produces the correct rejection result. Figure 15 shows an example of incorrect recognition (false rejection and false acceptance) by the proposed method. As shown in Figure 15, when misalignment and appearance similarity between the enrolled image and input image are large, the proposed method shows an incorrect recognition result.

6) SUBJECTIVE TEST

Because user convenience is an important and integral part of biometrics, we conducted an experiment to subjectively

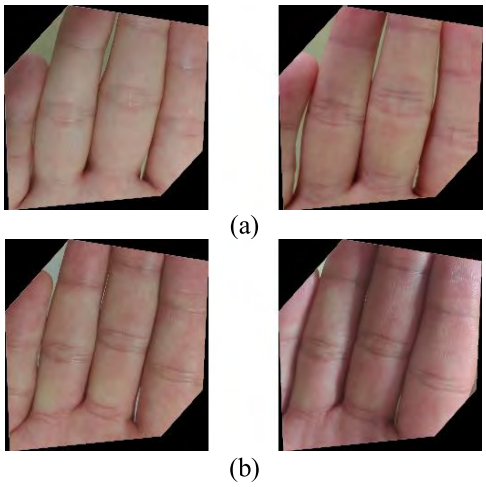

FIGURE 15. Examples of incorrect recognition cases: (a) false rejection case and (b) false acceptance case. Left and right images in (a) and (b) show the enrolled and input images, respectively. In (a) and (b), the images are the resized ROI image (224 × 224).

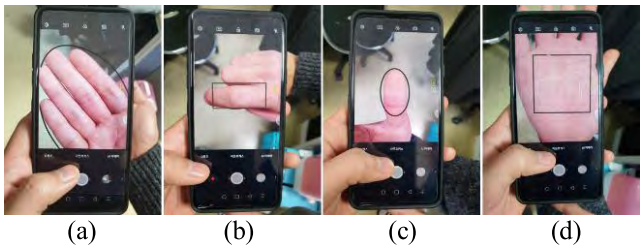

FIGURE 16. Subjective test of user convenience with (a) proposed method, (b) finger-knuckle-print recognition, (c) touchless fingerprint recognition, and (d) palmprint recognition.

evaluate the user convenience of the proposed method and the touchless fingerprint, palmprint, and finger-knuckle recognition methods (Figure 16). For fair evaluation, we used a guide window on the screen display in each method, as shown in Figure 16. The convenience score comprised five levels (5: very satisfied, 4: satisfied, 3: normal, 2: dissatisfied, and 1: very dissatisfied) and was conducted for 20 participants 10 times each.

The average user convenience scores in Figure 17 show that the proposed method received higher scores than the conventional hand-texture-based recognition methods. The touchless fingerprint received the lowest user convenience score because of the difficulty of positioning one finger in the guide window, as shown in Figure 16. In the case of the palmprint, whereas it is easy to position the palm in the guide window, the user convenience score is the second lowest because of the inconvenience of frequently raising the hand when acquiring the image.

We then performed a t-test [46] to show the significance of the user convenience of the proposed method. In Figure 17, the significance of the score difference between the proposed method and the finger-knuckle-print method, which has the second-highest user convenience score, was measured. The experimental results in Figure 18 show that the p-value of

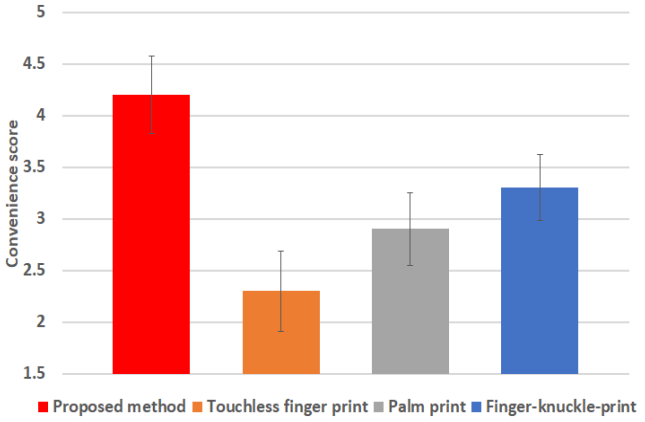

FIGURE 17. Comparisons of user convenience with our method and other hand-texture-based methods.

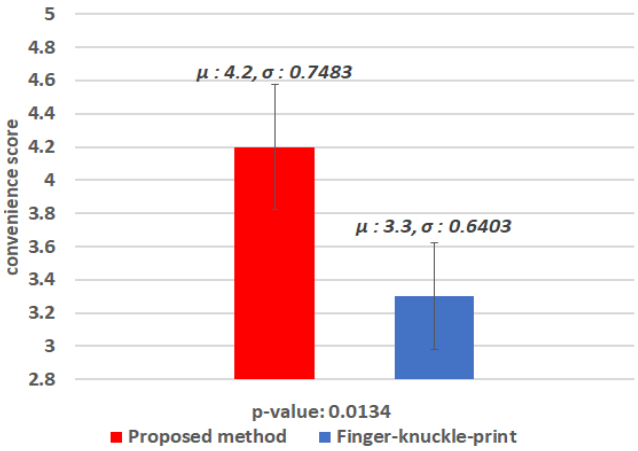

FIGURE 18. T-test with user convenience score by proposed method and finger-knuckle-print method.

the user convenience is 0.0134 (less than 0.05). These results show that the null hypothesis for user convenience is rejected at a 95% confidence level, showing that there is a significant difference between the user convenience of our method and that of the finger-knuckle-print method.

We also performed Cohen's d [47] analysis, by which the size of the difference between the two groups are shown using the effect size [48]. Cohen's d analysis has been widely used for analyzing the difference between two measured values. Generally, Cohen's d is classified as small at about 0.2–0.3, as medium at about 0.5, and as large at greater than or equal to 0.8. For example, if the calculated Cohen's d is closer to 0.2–0.3 than 0.5 and 0.8, we can say that the difference between measured values has a small effect size. If the calculated Cohen's d is closer to 0.8 than 0.2–0.3 and 0.5, we can say that the difference between measured values has a large effect size. The calculated Cohen's d about our user convenience was about 1.71 (closer to 0.8), from which we can conclude that the difference in user convenience between our method and the finger-knuckle-print method has a large effect. From the t-test and Cohen's d analysis, we can conclude that there is a significant difference in user convenience between our method and the finger-knuckle-print method.

In the next experiment, we examined the misalignment of the acquired images of finger-knuckle-print, fingerprint, and palmprint methods with the proposed method. Generally, the larger the misalignment, the larger the intra-class variation, thereby increasing FRR.

We used an LG G7 ThinQ [49] instead of an LG V20, which captured the images for the DMFW-DB1 database, to evaluate performance in various mobile devices. Using the same guide window and conditions as when we captured the images for the DMFW-DB1 database using an LG V20, we used the LG G7 ThinQ's frontal viewing camera (8 megapixels (2, 160 × 3, 840), 30 fps, auto-mode) to capture images of 20 participants, as shown in Figure 16. We sampled the acquired images in sets of five and calculated the motion deviation of the x and y coordinates between the acquired images. We then calculated the degree of misalignment

TABLE 9. Comparison of motion deviation within images obtained by the proposed method and other hand-texture-based methods (unit: Pixels).

Method	Motion deviation	
	(x, y)	Root mean square
Fingerprint	14.92, 13.32	20
Finger-knuckle-print	12.63, 11.25	16.91
Palmprint	10.59, 13.29	16.99
Finger-wrinkle **(proposed method)**	11.43, 12.7	17.09

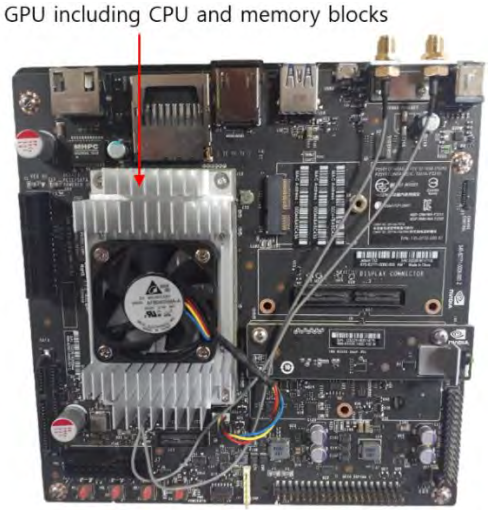

FIGURE 19. Jetson TX2 embedded system.

induced for each guide window according to recognition target. As shown in Table 9, the fingerprint method exhibited the greatest motion deviation, and the proposed method showed the second-largest deviation. However, this deviation was

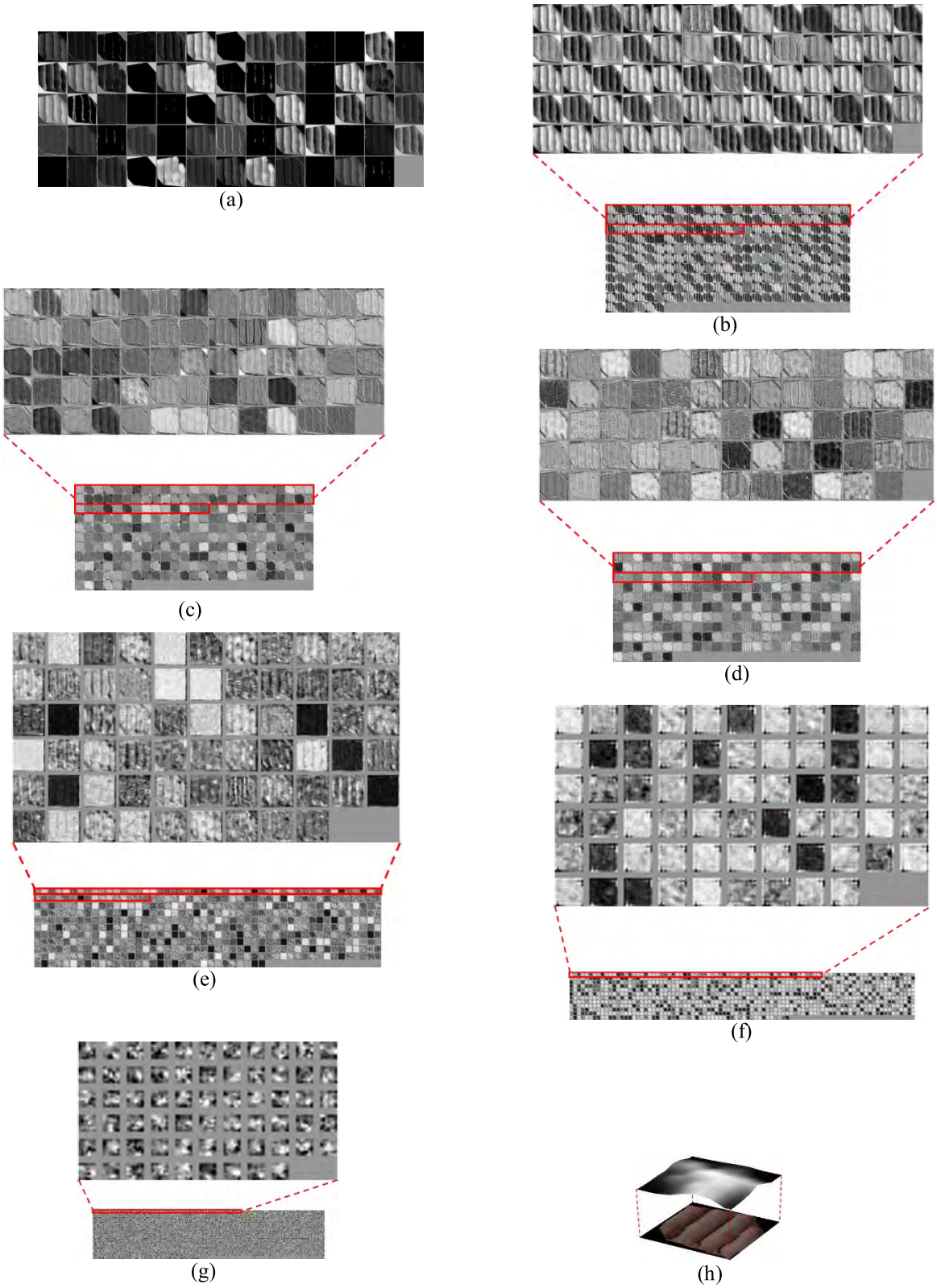

FIGURE 20. Examples of features maps extracted from each layer for the input finger-wrinkle images: Feature maps from (a) Conv1 of Table 2; (b)–(d) the first, second, and third iterations of residual blocks in Conv2 of Table 2, respectively; (e)–(g) the last residual blocks in Conv3, Conv4, and Conv5 of Table 2, respectively; (h) the 3 dimensional feature-map image obtained by averaging all feature-map values of (g).

TABLE 10. Comparisons of processing speed by the proposed method on a desktop computer and an embedded system (unit: ms).

Platform	Processing time
Desktop computer	27
Jetson TX2	72.9

not significantly different from the finger-knuckle-print and palmprint methods (maximum 0.18 pixels). The t-test results showed insignificant motion deviations between the finger-knuckle-print, palmprint, and proposed methods, suggesting that the proposed method does not generate significant misalignment between acquired images.

Based on these experimental results, the proposed method showed significantly higher user convenience than the other hand-texture-based methods with similar misalignments between acquired images.

7) PROCESSING SPEED OF PROPOSED METHOD

The processing speed of the ResNet-101-based finger-wrinkle recognition method proposed in this study was measured via the following experiment. Measurements were conducted using the desktop computer described in Section V.*A*. and the Jetson TX2 embedded system [53], which is often used in on-board deep-learning processing (Figure 19). Jetson TX2 has an NVIDIA Pascal^{TM}-family graphical processing unit (GPU) (256 CUDA cores) with 8-GB memory shared between the CPU and GPU, and 59.7 GB/s of memory bandwidth; it uses less than 7.5-W of power.

As shown in Table 10, regarding the recognition speed of the proposed method for one image, 27 ms of processing time was consumed in the desktop computer, whereas 72.9 ms was consumed in the Jetson TX2 embedded system. The Jetson TX2 embedded system has a longer processing time, because the computing resources are much more limited than with the desktop computer. However, this confirmed that the proposed method could be applied to embedded systems with limited computing resources.

8) ANALYSIS OF FEATURE MAP

In this subsection, the feature map extracted from each layer of deep ResNet for the input finger-wrinkle image is analyzed as shown in Figure 20. As described in Table 2, as the depth of the output feature map grows, the layer deepens and it becomes difficult to express in three dimensions. The feature maps for each depth are shown in order from top-left to bottom-right (Figure 20). Figure 20(a) shows the feature maps from Conv1 of Table 2, whereas Figures 20(b), (c), and (d) show the feature maps from the 1st, 2nd, and 3rd iterations of residual blocks in Conv2 of Table 2, respectively. Additionally, Figures 20(e), (f), and (g) show the feature maps from the last residual blocks in Conv3, Conv4, and Conv5 of Table 2, respectively. For example, the feature map obtained from Conv1 in Table 2 has a size of $112 \times 112 \times 64$. Therefore, Figure 20(a) shows 64 feature maps of 112×112 pixels from top-left to bottom-right.

As shown in Figure 20, the more feature maps are extracted from the deeper convolutional layers, the more abstracted the extracted features become and the wider the region of dominant features in one feature map. Thus, in Figure 20(a), features represent the shape of the original finger-wrinkle and high-frequency edge components. However, in Figure 20(g), the original finger-wrinkle shape disappears in place of a feature map with mainly abstracted low-frequency features. Figures 20(b), (c), and (d) involve more convolutional layers than Figure 20(a) and have feature maps in which the finger-wrinkle shape has vanished more. However, because they comprise a structure that maintains the original feature map through the residual block before the convolution operation, the original finger-wrinkle shape does not disappear and can be confirmed to some extent. However, in Figures 20(f) and (g), the convolution layers become deeper, and the original finger-wrinkle shape is hardly retained. Additionally, the degradation of recognition performance caused by misalignment between the enrolled and recognized images caused by the rough detection of the finger-wrinkle region proposed in this study can be compensated to some extent by the characteristics of the abstract and wide low-frequency features in the feature map of Figure 20(g).

In Figure 20(h), the 3 dimensional image obtained by averaging the feature magnitudes in the channel (depth) direction in the feature map of Figure 20(g) is presented with the original finger-wrinkle image. As shown in this figure, the magnitudes of feature map values are also large in the rough finger-wrinkle region, which can prove that important features can be extracted from whole finger-wrinkle regions.

VI. CONCLUSION

This study proposed the finger-wrinkle recognition method based on the visible light camera of smartphone. To address misalignment issues between the enrolled and the recognized images, a guide window with captured image was shown in the smartphone display. The guide window was designed to enhance user convenience with less misalignment. ResNet-based recognition method was used to alleviate the degradation of recognition performance caused by misalignment and illumination variation occurring between the enrolled and recognized images, blur, and rotation. Most recognition errors occurred because of severe misalignment between the enrolled and recognized images caused by translation, rotation, and severe illumination variation.

Furthermore, with a subjective test, we compared the user convenience of the fingerprint, finger-knuckle-print, and palmprint recognition methods with our method. We verified its reliability with a t-test and Cohen's d value. Additionally, we experimentally compared the degree of misalignment during image acquisition of the four methods. Thus, we found no significant differences between the amount of misalignment between the images acquired from the four methods.

In future work, we would research about generative adversarial network-based method [50] to convert input finger-wrinkle image similar to the enrolled image, thereby improving recognition performance. In addition, we would conduct research on improving recognition performance via multimodal biometrics, combining finger-shape or finger-print recognition.

REFERENCES

[1] *LG G8 ThinQ*. Accessed: Mar. 18, 2019. [Online]. Available: https://www.lg.com/us/mobile-phones/g8-thinq
[2] *Samsung Galaxy S10*. Accessed: Mar. 18, 2019. [Online]. Available: https://www.samsung.com/us/mobile/galaxy-s10
[3] K. Tang, A. Liu, W. Wang, P. Li, and X. Chen, "A novel fingerprint sensing technology based on electrostatic imaging," *Sensors*, vol. 18, no. 9, p. 3050, 2018.
[4] M. A. Medina-Pérez, M. García-Borroto, A. E. Gutierrez-Rodríguez, and L. Altamirano-Robles, "Improving fingerprint verification using minutiae triplets," *Sensors*, vol. 12, no. 3, pp. 3418–3437, 2012.
[5] H. AlShehri, M. Hussain, H. AboAlSamh, and M. AlZuair, "A large-scale study of fingerprint matching systems for sensor interoperability problem," *Sensors*, vol. 18, no. 4, p. 1008, 2018.
[6] J. J. Engelsma, S. S. Arora, A. K. Jain, and N. G. Paulter, "Universal 3D wearable fingerprint targets: Advancing fingerprint reader evaluations," *IEEE Trans. Inf. Forensics Security*, vol. 13, no. 6, pp. 1564–1578, Jun. 2018.
[7] M. Liu, Y. Tian, and L. Li, "A new approach for inner-knuckle-print recognition," *J. Vis. Lang. Comput.*, vol. 25, no. 1, pp. 33–42, 2014.
[8] L. Zhang, L. Zhang, D. Zhang, and H. Zhu, "Ensemble of local and global information for finger–knuckle-print recognition," *Pattern Recognit.*, vol. 44, pp. 1990–1998, Sep. 2011.
[9] L. Zhang and H. Li, "Encoding local image patterns using Riesz transforms: With applications to palmprint and finger-knuckle-print recognition," *Image Vis. Comput.*, vol. 30, pp. 1043–1051, Dec. 2012.
[10] S. Aoyama, K. Ito, and T. Aoki, "Finger-knuckle-print recognition using BLPOC-based local block matching," in *Proc. 1st Asian Conf. Pattern Recognit.*, Beijing, China, Nov. 2011, pp. 525–529.
[11] L. Zhang, L. Zhang, D. Zhang, and Z. Guo, "Phase congruency induced local features for finger-knuckle-print recognition," *Pattern Recognit.*, vol. 45, pp. 2522–2531, Jul. 2012.
[12] S. Aoyama, K. Ito, and T. A. Aoki, "A finger-knuckle-print recognition algorithm using phase-based local block matching," *Inf. Sci.*, vol. 268, pp. 53–64, Jun. 2014.
[13] D. Kusanagi, S. Aoyama, K. Ito, and T. Aoki, "A practical person authentication system using second minor finger knuckles for door security," *IPSJ Trans. Comput. Vis. Appl.*, vol. 9, p. 8, Mar. 2017.
[14] A. Kumar and C. Ravikanth, "Personal authentication using finger knuckle surface," *IEEE Trans. Inf. Forensics Security*, vol. 4, no. 1, pp. 98–110, Mar. 2009.
[15] L. Fei, G. Lu, W. Jia, S. Teng, and D. Zhang, "Feature extraction methods for palmprint recognition: A survey and evaluation," *IEEE Trans. Syst. Man, Cybern. Syst.*, vol. 49, no. 2, pp. 346–363, Feb. 2019.
[16] J. Promila and V. Laxmi, "Palmprint matching using LBP," in *Proc. Int. Conf. Comput. Sci.*, Phagwara, India, Sep. 2012, pp. 110–115.
[17] F. Hao, L. Yang, G. Yang, N. Liu, and Z. Liu, "RFPIQM: Ridge-based forensic palmprint image quality measurement," *IEEE Access*, vol. 6, pp. 62076–62088, 2018.
[18] Q. Zheng, A. Kumar, and G. Pan, "Suspecting less and doing better: New insights on palmprint identification for faster and more accurate matching," *IEEE Trans. Inf. Forensics Security*, vol. 11, no. 3, pp. 633–641, Mar. 2016.
[19] T. Connie, A. B. J. Teoh, M. G. K. Ong, and D. N. C. Ling, "An automated palmprint recognition system," *Image Vis. Comput.*, vol. 23, no. 5, pp. 501–515, May 2005.
[20] A. K. Jain and J. Feng, "Latent palmprint matching," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 31, no. 6, pp. 1032–1047, Jun. 2009.
[21] S. Priya and M. Ezhilarasan, "A novel palmprint authentication system using level 3 pore feature," in *Proc. Int. Conf. Commun. Signal Process.*, Chennai, India, Apr. 2018, pp. 623–626.
[22] P. Dubey, T. Kanumuri, and R. Vyas, "Palmprint recognition using binary wavelet transform and LBP representation," in *Proc. Int. Conf. Recent Innov. Signal Process. Embedded Syst.*, Bhopal, India, Oct. 2017, pp. 201–205.
[23] A. Gumaei, R. Sammouda, A. M. Al-Salman, and A. Alsanad, "An effective palmprint recognition approach for visible and multispectral sensor images," *Sensors*, vol. 18, p. 1575, May 2018.
[24] A. Kumar, "Toward more accurate matching of contactless palmprint images under less constrained environments," *IEEE Trans. Inf. Forensics Security*, vol. 14, no. 1, pp. 34–47, Jan. 2019.
[25] D. Cheng, X. Zhang, and X. Xu, "An improved recognition approach for noisy multispectral palmprint by robust L2 sparse representation with a tensor-based extreme learning machine," *Sensors*, vol. 19, no. 2, p. 235, 2019.
[26] L. Zhang, L. Zhang, and D. Zhang, "Finger-knuckle-print verification based on band-limited phase-only correlation," in *Proc. Int. Conf. Comput. Anal. Images Patterns*, Münster, Germany, Sep. 2009, pp. 141–148.
[27] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Las Vegas, NV, USA, Jun./Jul. 2016 pp. 770–778.
[28] *Dongguk Mobile Finger-Wrinkle Database (DMFW-DB1) and CNN Models With Algorithms*. Accessed: Mar. 28, 2019. [Online]. Available: http://dm.dgu.edu/link.html
[29] Z.-U. Rahman, D. J. Jobson, and G. A. Woodell, "Retinex processing for automatic image enhancement," *Proc. SPIE*, vol. 13, pp. 100–111, Jan. 2004.
[30] G. P. Nam and K. R. Park, "New fuzzy-based Retinex method for the illumination normalization of face recognition," *Int. J. Adv. Robotic Syst.*, vol. 9, no. 3, p. 103, 2012.
[31] *CS231n Convolutional Neural Networks for Visual Recognition*. Accessed: Mar. 4, 2019. [Online]. Available: https://cs231n.github.io/convolutional-networks/#overview
[32] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," in *Proc. 32nd Int. Conf. Mach. Learn.*, Lille, France, Jul. 2015, pp. 448–456.
[33] V. Nair and G. E. Hinton, "Rectified linear units improve restricted boltzmann machines," in *Proc. 27th Int. Conf. Mach. Learn.*, Haifa, Israel, Jun. 2010, pp. 807–814.
[34] X. Glorot, A. Bordes, and Y. Bengio, "Deep sparse rectifier neural networks," in *Proc. 14th Int. Conf. Artif. Intell. Statist.*, Fort Lauderdale, FL, USA, Jun. 2011, pp. 315–323.
[35] J. Heaton, *Artificial Intelligence for Humans: Deep Learning and Neural Networks*, vol. 3. Chesterfield, MO, USA: Heaton Res. Inc., 2015.
[36] *LG V20*. Accessed: Mar. 18, 2019. [Online]. Available: https://www.lg.com/us/cell-phones/lg-US996-Silver-v20-unlocked
[37] *GeForce GTX 1070*. Accessed: Mar. 4, 2019. [Online]. Available: https://www.geforce.com/hardware/desktop-gpus/geforce-gtx-1070/specifications
[38] *Caffe*. Accessed: Mar. 4, 2019. [Online]. Available: http://caffe.berkeleyvision.org/
[39] *Visual Studio 2013*. Accessed: Mar. 4, 2019. [Online]. Available: https://www.microsoft.com/en-us/search/result.aspx?q=visual+studio+2013
[40] *CUDA*. Accessed: Mar. 4, 2019. [Online]. Available: https://en.wikipedia.org/wiki/CUDA
[41] *cuDNN*. Accessed: Mar. 4, 2019. [Online]. Available: https://developer.nvidia.com/cudnn
[42] L. Bottou, "Large-scale machine learning with stochastic gradient descent," in *Proc. 19th Int. Conf. Comput. Statist.*, Paris, France, Aug. 2010, pp. 177–186.
[43] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," in *Proc. 3rd Int. Conf. Learn. Represent.*, San Diego, CA, USA, May 2015, pp. 1–14.
[44] Y. Wu, B. Jiang, and N. Lu, "A descriptor system approach for estimation of incipient faults with application to high-speed railway traction devices," *IEEE Trans. Syst., Man, Cybern. Syst.*, to be published.
[45] W. Kim, J. M. Song, and K. R. Park, "Multimodal biometric recognition based on convolutional neural network by the fusion of finger-vein and finger shape using near-infrared (NIR) camera sensor," *Sensors*, vol. 18, p. 2296, Jul. 2018.
[46] *Student's t-Test*. Accessed: Mar. 20, 2019. [Online]. Available: https://en.wikipedia.org/wiki/Student%27s_t-test
[47] J. Cohen, "A power primer," *Psychol. Bull.*, vol. 112, pp. 155–159, Jul. 1992.

[48] S. Nakagawa and I. C. Cuthill, "Effect size, confidence interval and statistical significance: A practical guide for biologists," *Biol. Rev.*, vol. 82, pp. 591–605, Nov. 2007.
[49] *LG G7 ThinQ*. Accessed: Mar. 18, 2019. [Online]. Available: https://www.lg.com/us/cell-phones/lg-G710ULM-Unlocked-g7-thinq
[50] P. Isola, J.-Y. Zhu, T. Zhou, and A. A. Efros, "Image-to-image translation with conditional adversarial networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Honolulu, HI, USA, Jul. 2017, pp. 5967–5976.
[51] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "ImageNet: A large-scale hierarchical image database," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Miami, FL, USA, Jun. 2009, pp. 248–255.
[52] *Multinomial Logistic Loss*. Accessed: Nov. 30, 2018. [Online]. Available: http://caffe.berkeleyvision.org/doxygen/classcaffe_1_1MultinomialLogisticLossLayer.html
[53] *Jetson TX2 Module*. Accessed: Feb. 24, 2019. [Online]. Available: https://www.nvidia.com/en-us/autonomous-machines/embedded-systems-dev-kits-modules/

CHAN SIK KIM received the B.S. degree in electronics and electrical engineering from Dongguk University, Seoul, South Korea, in 2017, where he is currently pursuing the M.S. degree in electronics and electrical engineering. He designed the recognition system by finger-wrinkle-based convolutional neural network and wrote the draft of original paper. His research interests include biometrics and pattern recognition.

NAM SUN CHO received the B.S. degree in electronics and electrical engineering from Dongguk University, Seoul, South Korea, in 2018, where he is currently pursuing the M.S. degree in electronics and electrical engineering. His research interests include biometrics and pattern recognition. He helped to perform the experiments.

KANG RYOUNG PARK received the B.S. and M.S. degrees in electronic engineering and the Ph.D. degree in electrical and computer engineering from Yonsei University, Seoul, South Korea, in 1994, 1996, and 2000, respectively. He has been a Professor with the Division of Electronics and Electrical Engineering, Dongguk University, since 2013. He supervised this research and revised the original paper. His research interests include image processing and biometrics.