Unconstrained Fingerphoto Database

Shaan Chopra*, Aakarsh Malhotra*, Mayank Vatsa, Richa Singh
IIIT Delhi, India
{shaan15090, aakarshm, mayank, rsingh}@iiitd.ac.in

Abstract

Biometrics based user authentication for mobile devices is now popular with face and fingerprints being the primary modalities. Fingerphoto, an image of a person's finger captured using inbuilt smartphone camera, based user authentication is an attractive and cost-effective alternative. Existing research focuses on constrained or semi-constrained environment; whereas, challenges such as user cooperation, number of fingers, background, orientation, and deformation are important to address before fingerphoto authentication becomes usable. This paper presents the first publicly available unconstrained fingerphoto database, termed as UNconstrained FIngerphoTo (UNFIT) database, which contains fingerphoto images acquired in unconstrained environments. We also present baseline results with deep learning based segmentation as well as CompCode and ResNet50 representation based matching approaches. We assert that the availability of the proposed database can encourage research in this important domain.

1. Introduction

Smartphones store a lot of personal and confidential information which, if compromised can lead to identity theft and loss of critical information. Various authentication methods such as passwords and PINs are used to prevent unauthorized access to the smartphones. Alternatively, there is an increasing trend in the usage of biometric modalities for mobile authentication in the last few years. Particularly, fingerprint and face are being used for mobile-based authentication. Another approach which is currently being explored is fingerphoto authentication [26]. Fingerphoto, as illustrated in Figure 1, is the image of the frontal region of fingers. Using smartphone's camera, a picture of the person's finger is captured and utilized for recognition. The 4F technology uses the rear camera and flash of the smartphone to take multiple images of the finger and utilize it for matching [22]. Existing research in fingerphoto authentication focuses on constrained

(a) (b)

Figure 1: (a) Shows the fingerphoto acquiring mechanism using a smartphone camera, and (b) sample captured fingerphoto image.

environment, and generally, these algorithms have not been evaluated for the unconstrained scenarios. Law-enforcement agencies around the world have shown their interest in similar applications [7] [10] which showcases the need for this technology.

A major reason for limited research in this problem domain is unavailability of fingerphoto databases. Table 1 lists the datasets used in literature for benchmarking fingerphoto recognition algorithms. Out of these datasets, only two of them are publicly available for the research community:

- HKPU Low Resolution Fingerprint Database [14]: The database consists of 1566 low-resolution fingerphoto images from 156 subjects. Fingerphotos are acquired over two different sessions using a webcam. However, the database incorporates only low-resolution variations and can be termed as a semi-constrained database.
- IIITD Smartphone Fingerphoto Database [26]: The database consists of 4096 fingerphoto images from 64 subjects. The database is acquired using a smartphone camera, with fingerphotos spanning challenges of varying background and varying illumination. Similar to HKPU Low-Resolution Fingerprint Database, this database also falls under the category of semi-

*Equal contribution by student authors.

Table 1: Literature review of existing work on fingerphoto databases.

Research	Device	Subjects	# Samples	Challenges	Public	Nature
Lee et al.[15]	Phone	-	1240	Background	✗	Constrained
Piuri & Scotti[24]	Webcam	15	150	Background	✗	Semi-constrained
Kumar & Zhou[14]	Webcam	156	1566	Resolution	✓	Semi-constrained
Li et al.[17]	Phone	25	1800	Background, illumination	✗	Semi-constrained
Raghavendra et al.[25]	Phone	25	1800	Illumination	✗	Constrained
Tiwari & Gupta[29]	Phone	50	150	Illumination	✗	Constrained
Sankaran et al.[26]	Phone	64	4096	Background, illumination	✓	Semi-constrained
Proposed	**Phone**	**115**	**3450**	**Background, multiple fingers, blur, illumination, resolution, affine variations, deformations**	✓	**Unconstrained**

constrained fingerphoto database.

While these two public databases are good to initiate research on fingerphoto recognition, they do not cover the challenges present in an unconstrained acquisition, as shown in Figure 2. Other in-house databases proposed in the literature [15, 16, 17, 18, 20, 28] are also constrained or semi-constrained databases. However, there is an immense scope for improvement in unconstrained touchless fingerprint recognition. As highlighted in Figure 2, the challenges due to unconstrained environment make the task of finger detection and recognition difficult. For promoting detection and authentication/recognition of fingerphoto in challenging scenarios, this paper presents UNFIT: an unconstrained fingerphoto database. The key contributions of this research are:

- a publicly available unconstrained fingerphoto database to study and analyze the variations in environmental parameters affecting fingerphoto matching. The database contains 3450 images pertaining to 115 subjects along with an annotation of finger location for every fingerphoto, and
- an experimental protocol for the database along with a segmentation algorithm for fingerphotos captured in an unconstrained environment. Classification networks such as VGG SegNet [27] and FCN 8 [19] are used to perform fingerphoto segmentation and the results are documented. Further, baseline results of fingerphoto authentication using CompCode [13] and ResNet [11] are also presented.

2. UNconstrained FIngerphoTo (UNFIT) Database

One of the missing components in existing works is the lack of a publicly available fingerphoto database acquired in an unconstrained environment. To fill this gap, we present a novel fingerphoto database acquired in an unconstrained environment. The database incorporates several variations pertaining to unconstrained environments. The details of the database along with the variations are described below.

Table 2: A summary of the sets of the proposed unconstrained fingerphoto (UNFIT) database.

	Fingers	Classes	Images
Set-I	Index	115	1150
	Middle	115	1150
	Subtotal:	230	2300
Set-II	Multiple Fingers	115	1150
	Total:		3450

2.1. Database Details

A novel fingerphoto database consisting of images from 115 subjects is collected over a time span of three months. The database is termed as Unconstrained FInger phoTo (UNFIT) database[1]. In total, the database contains 3450 fingerphotos from 230 classes. Table 2 provides a summary of the proposed database and Figure 2 shows sample images. From 115 subjects, two sets are collected as follows:

- **Set I: Single Finger** - Fingerphoto images corresponding to index and middle fingers are acquired. As per user convenience, fingerphotos are captured either from left or right hand. However, the same hand for both index and middle fingers is used. No other constraints are enforced during acquisition with respect to position, focus, illumination, or alignment of the finger. Sample images from this set are illustrated in Figure 3(a) and Figure 3(b). For each finger, ten instances are acquired. Hence, the set has a total of: 115 subjects $\times$ 2 fingers $\times$ 10 instances = 2300 images.
- **Set II: Multiple Fingers** - In a real-world scenario, a user might intentionally or unintentionally present multiple fingers during acquisition. Instead of discarding

[1]The UNFIT database can be found at: http://iab-rubric.org/resources/UNFIT.html

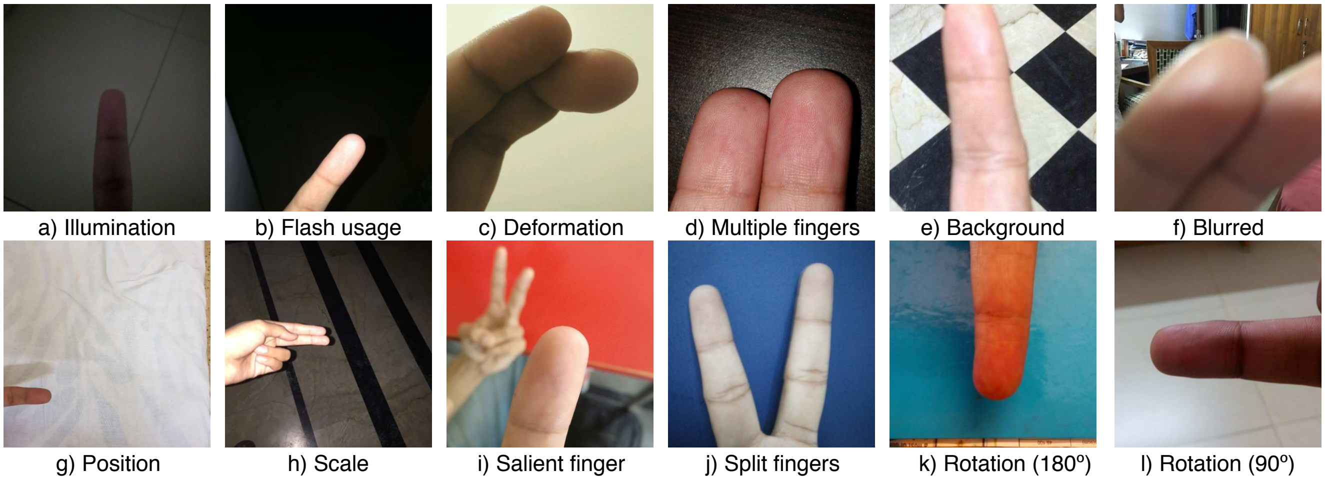


Figure 2: Sample fingerphoto images from the proposed UNFIT database. The database incorporates numerous challenges and is acquired in an unconstrained environment. The images are captured using multiple smartphones with different resolutions.

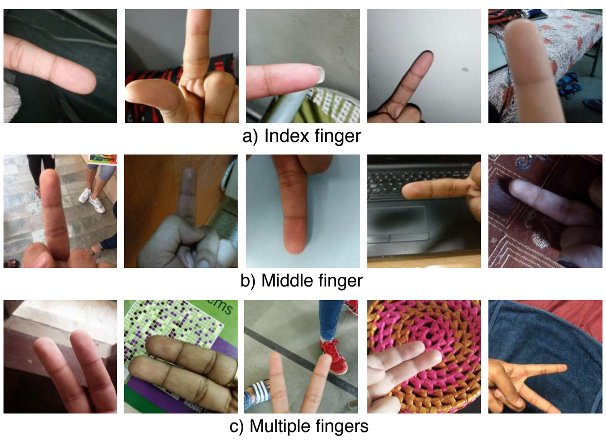


Figure 3: Sample fingerphoto images illustrating two sets of the proposed UNFIT database.

the extra fingers, information from subsequent fingers can be extracted and used towards enhancing recognition performance. To show the effect of multiple fingers towards recognition, a set containing images of both index and middle finger together is collected, as shown in Figure 3(c). Similar to the previous set, both the fingers are from the same hand. For each subject, ten fingerphoto images are acquired, resulting in 1150 images (= 115 subjects $\times$ 10 instances).

2.2. Data Acquisition

The fingerphoto images are captured using 45 different smartphones belonging to the subjects. The usage of different smartphones adds variations pertaining to resolution and camera sensor in the proposed database. In the database, 48% of the photos are captured using an iPhone or a OnePlus

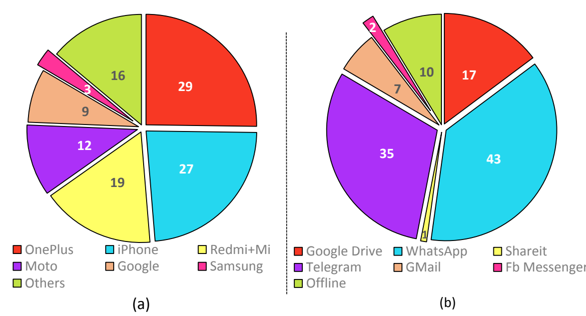


Figure 4: a) A summary of the mobile devices used to acquire the UNFIT database, and b) Online and offline mediums of data collection.

device. Other phones include Redmi devices, Mi 4, Samsung Galaxy, Google Nexus, Le 1s, Moto G, Moto C, Moto M, HTC devices, Lenovo K3 Note, Lenovo K4, and Micromax Canvas. Figure 4(a) shows the distribution of different mobile devices used for collecting the database. The resolutions of the smartphone cameras varied in the range of 8MP to 16MP.

To include the effect of image compression due to transmission, the database is collected via both online and offline procedures. The online procedure included data collection via applications such as WhatsApp [6], Telegram [5], Facebook messenger [1], Gmail [3], and Google Drive [2]. These applications add to the diversity in the database with their different compression rates for images. In the offline procedure, the database is collected using different phone devices and transfered using a Pen-drive. Figure 4(b) shows the different modes of data collection, online and offline. The optional usage of auto-focus and flash while acquiring fingerphotos of participants introduced illumination, intensity, and blur variations in the database. Other affine variations such as a scale, rotation, translation, along with background variations are also present in the database.

Due to the challenges posed in the proposed database, the first step is to locate and segment the finger(s). In the next section, we present a deep learning method to segment the foreground fingerphoto and perform its comparison with existing skin-color based segmentation techniques.

2.3. Ground-truth Annotation

The proposed database poses various challenges such as translation, rotation, scale, orientation, resolution, background, and illumination variations. Hence, the position and visual appearance of fingers vary diversely. To determine the exact location of the fingers, it is essential to provide the ground-truth annotation for finger locations. The images are manually annotated using a GUI based segmentation tool developed in MATLAB [4] using Piotr Dollar's toolbox [9]. The segmentation tool utilizes rectangular-rotating bounding boxes to locate and annotate the finger regions. Along with the database, the ground truth annotations will also be made publicly available. They are represented in form of a mask, with the same image name in a separate folder.

2.4. Potential Usage of UNFIT database

Various studies have proposed modules for pre-processing [14, 15, 16, 17, 20, 23, 24, 25, 28, 29], feature extraction [14, 20, 21, 26, 29], and feature matching [8, 14, 18, 20, 26, 29] of fingerphotos. Owing to the challenging variations and its ground truth annotation, the proposed UNFIT database can be used in the following research directions:

- **Touchless Fingerprint Detection:** The UNFIT database contains the manual annotation of the fingerphotos. These annotations allow the researchers to use the database towards evaluating the performance of fingerphoto detection and segmentation algorithms in an unconstrained environment.

- **Fingerphoto Verification and Identification:** The dataset can be used for evaluating the fingerphoto recognition algorithms under verification and identification scenarios.

- **Fusion approaches:** The dataset contains images when multiple fingers (index and middle) are acquired together. It can be potentially used for comparing fusion based approaches for fingerphoto recognition.

3. Experimental Protocol and Segmentation Benchmarking

In this paper, we perform benchmarking for fingerphoto segmentation and authentication/verification. We first prepare a protocol for the training-testing split. This would assist researchers to perform fingerphoto pre-processing, segmentation, and matching. Using the proposed protocol, we benchmark the performance of multiple fingerphoto segmentation and feature extraction, matching algorithms.

3.1. Experimental Protocol

The UNFIT database contains a total of 3450 images from 115 subjects. The dataset is divided into train and test split in a 50:50 ratio. The split is performed in a subject disjoint manner, where 58 subjects correspond to training and the images pertaining to remaining 57 subjects are used as testing data. Both index and middle fingers are treated as separate classes. Hence, the training data has 116 unique classes, whereas, the testing data has 114 unique classes. From each subject in test data, first five fingerphoto images are treated as the gallery, while the remaining samples are treated as probe (query) images. Note that, index-index, middle-middle, and multiple-multiple finger matching from the same person are considered for obtaining the genuine scores during matching, while scores generated from all other matches are considered as impostor scores.

3.2. Fingerphoto Segmentation Framework

The discriminative information in a finger lies in the ridge-valley pattern, which contributes to the uniqueness of the fingerprint. Thus, the aim of the segmentation framework is to discard background information, and keep only the foreground finger information. To achieve this task, the framework for fingerphoto segmentation utilizes VGG SegNet [27]. Pre-trained VGG SegNet is fine-tuned to perform the task of fingerphoto segmentation. However, as illustrated in the predicted mask in Figure 5, the VGG SegNet architecture provides a tight bound on the fingerphoto. Hence, the VGG Segnet architecture is followed by a layer of smoothening to increase the number of foreground pixels. Figure 5 shows the full segmentation pipeline using VGG SegNet architecture followed by 32×32 block-wise smoothening. As seen from Figure 5, VGG SegNet [27] has encoder and decoder networks followed by a Softmax classification layer that performs classification. The Softmax layer predicts whether a test pixel is a foreground pixel or not. The algorithm is summarized in Algorithm 1.

The VGG SegNet based algorithm is also compared with VGG FCN 8 [19], where the pre-trained Fully Convolutional Network (FCN) is also adapted followed by 32×32 block-wise smoothening. VGG FCN 8 [19] also trains a fully convolutional network. It uses Adadelta optimizer and

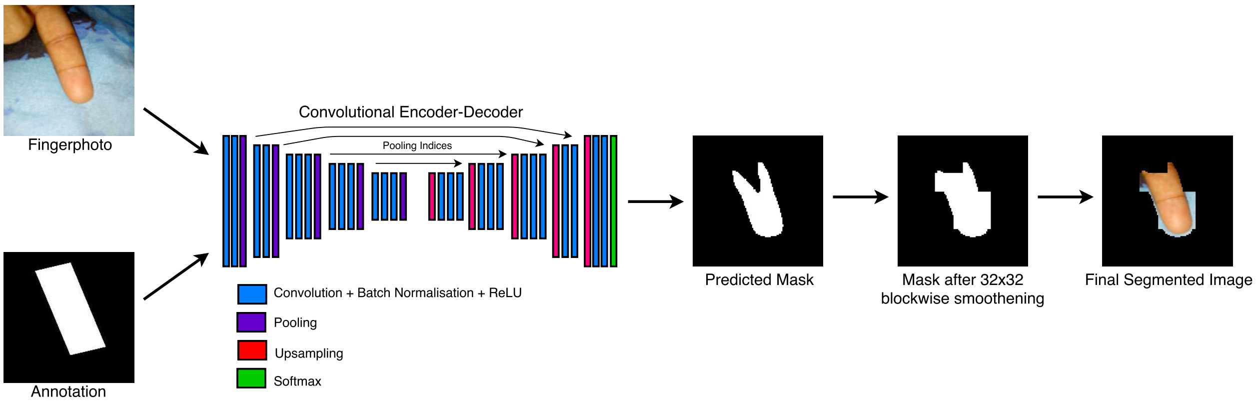


Figure 5: Stepwise illustration of the segmentation framework using VGG SegNet followed by 32×32 block-wise smoothening.

categorical cross-entropy loss.

Result: Final segmented image mask
Feed training images and annotations into VGG SegNet Architecture;
Obtain and binarize predicted images;
pred = Number of predicted images;
fp = Number of foreground (finger) pixels;
bp = Number of background (non-finger) pixels;
block = Number of 32×32 pixels non-overlapping blocks in image;
while *pred* $\neq$ *0* **do**
    Divide image into 32×32 blocks;
    while *block* **do**
        if *fp* $\geq$ *bp* **then**
            set all pixels of block as foreground;
        else
            leave block as it is;
        end
        block = block - 1;
    end
    pred = pred - 1;
end

Algorithm 1: Fingerphoto segmentation algorithm using VGG SegNet architecture followed by 32×32 block-wise smoothening.

3.2.1 Implementation Details

The encoder network of VGG SegNet is provided with an image of size 224×224×3. The output of the encoder network is a multi-channel image of size 14×14×512. The output of encoder network is then given as input to the decoder network. The final decoder output of size 112×112×2 is provided to the Softmax layer which performs binary classification on each image pixel. The prediction is a binary mask with white pixel representing the location of finger or and black representing non-finger. Similarly, the FCN is also given 224×224×3 images. Both the networks are also provided with the corresponding annotation masks of size 224×224×3, where a 0 value represents background and 1 value represents foreground. To fine-tune the deep architectures, the training dataset is first augmented and then used for fine-tuning. Image augmentation is performed by rotation (90, 180, and 270 degrees), mirror flipped, blurred, and intensity changed images in the training set. The corresponding annotated images (masks) are also updated according to the augmentation operation and added in the set accordingly. The architecture is trained for 100 epochs on the augmented training set.

The deep learning segmentation framework is also compared with the skin-color based segmentation algorithm [12]. The skin-color based segmentation is performed by converting the original RGB image to YCbCr and HSV color space. The Cb, Cr, and Hue channels are used to find the probable skin-colored region in the image. The comparison across algorithms are performed using the metrics presented in the following sections.

3.2.2 Segmentation Performance Metrics

To test the performance of fingerphoto segmentation, the following metrics are used:

- Segmentation Accuracy (SA) is defined as

$$SA = \frac{CCB}{TB} \tag{1}$$

  where, CCB is the number of Correctly Classified Blocks and TB is the total number of blocks.

- Foreground Segmentation Accuracy (FSA) is defined as

$$FSA = \frac{CCFB}{TFB} \tag{2}$$

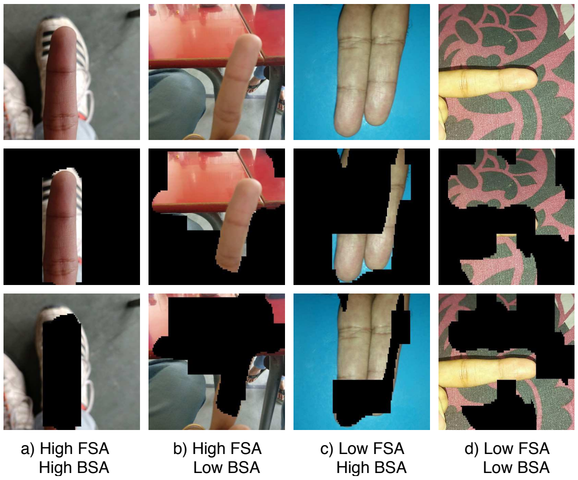


Figure 6: A visual interpretation of FSA and BSA with respect to the fingerphoto segmentation.

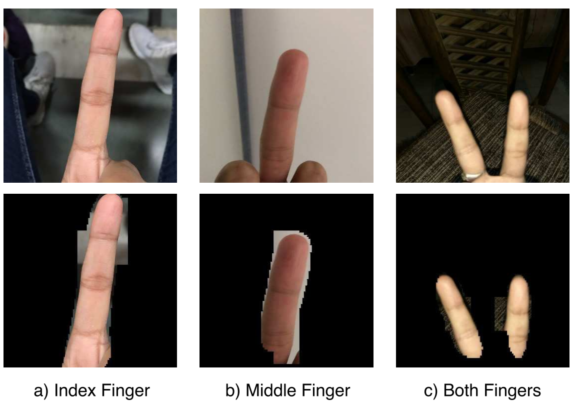


Figure 7: Demonstration of the cases where the VGG SegNet + 32×32 block-wise smoothening framework successfully segmented.

where, CCFB is the number of Correctly Classified Foreground Blocks and TFB is the total number of foreground blocks in the ground truth image.

- Background Segmentation Accuracy (BSA):

$$BSA = \frac{CCBB}{TBB} \tag{3}$$

where, CCBB is the number of correctly classified background blocks and TBB is the total number of background blocks in the ground truth images.

As shown in Figure 6, a visual interpretation of FSA and BSA can be observed using deep learning based algorithm. In a real world scenario, we expect the segmentation algorithm to yield high FSA and high BSA and hence, high overall segmentation accuracy.

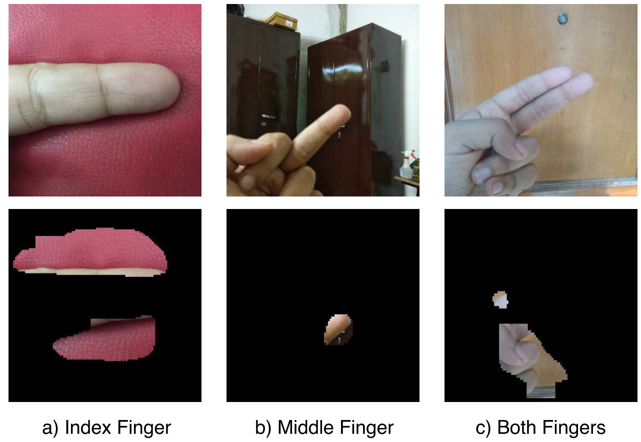


Figure 8: Demonstration of the cases where the VGG SegNet+32×32 block-wise smoothening framework failed.

3.3. Fingerphoto Feature Extraction and Matching

As shown in the literature [17, 26], minutiae-based techniques are likely to fail for fingerphoto recognition. Hence, in our experiments, two algorithms are used: Competitive Coding (CompCode) [13] and ResNet50 [11]. CompCode features are non-minutiae based feature representation for fingerprints recognition. It utilizes Gabor filters with J different orientations, each varying by $\frac{\pi}{J}$. The CompCode features are extracted by convolving the real part of the Gabor filter G_r with the image I. These features are then matched using Hamming distance to obtain a distance score. In the experiments, all the segmented image are first resized to a standard size of 400×400, followed by extracting their CompCode features. A comparison with representation obtained by a deep learning model is also performed. We utilized pre-trained ResNet50 architecture to obtain feature representation, which are matched using cosine similarity. To showcase the verification results, Receiver Operating Characteristic (ROC) curve is used.

4. Experimental Results

The segmentation results are reported in terms of FSA, BSA, and SA. The proposed deep learning technique is also compared with state-of-the-art method deployed for fingerphoto segmentation [26] (Exp. 8). It is observed that VGG SegNet along with 32×32 block-wise smoothening yields the best FSA and performs well in terms of BSA and SA. Table 3 and Table 4 shows the FSA, BSA, and SA obtained using various segmentation techniques. The instances where the deep learning segmentation algorithm performed well is shown in Figure 7, whereas Figure 8 shows the samples where the deep learning algorithm failed. The major conclusions that can be drawn are as follows:

- On comparison of FSA with BSA in Table 3, we ob-

Table 3: Segmentation accuracies using different deep learning algorithms with and without block-wise smoothening layer.

Exp. #	Algorithm		All Together	Index Finger	Middle Finger	Multiple Fingers
Exp. 1	VGG FCN 8	FSA	61.46%	60.11%	63.66%	60.62%
		BSA	93.92%	94.22%	94.09%	93.45%
		SA	88.55%	89.45%	90.19%	86.01%
Exp. 2	VGG FCN 8 + 32×32 block-wise smoothening	FSA	65.81%	64.19%	67.97%	65.26%
		BSA	92.04%	92.41%	92.43%	91.27%
		SA	87.56%	88.37%	89.16%	85.16%
Exp. 3	VGG SegNet	FSA	66.75%	65.98%	70.16%	64.10%
		BSA	**94.69%**	**95.04%**	**94.89%**	**94.15%**
		SA	**90.08%**	**91.01%**	**91.77%**	**87.45%**
Exp. 4	VGG SegNet + 32×32 block-wise smoothening (Proposed)	FSA	**71.22%**	**70.28%**	**74.49%**	**68.90%**
		BSA	92.71%	93.16%	93.06%	91.91%
		SA	89.04%	89.89%	90.62%	86.61%

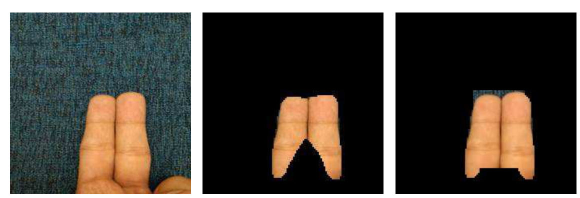
Table 4: Segmentation accuracies obtained using skin-color based techniques and combining it with deep learning algorithms.

Exp. #	Algorithm		All Together	Index Finger	Middle Finger	Multiple Fingers
Exp. 5	Skin-color based segmentation	FSA	58.63%	58.13%	57.52%	60.22%
		BSA	88.95%	89.35%	88.85%	88.65%
		SA	84.40%	85.25%	85.22%	82.73%
Exp. 6	Skin-color based segmentation + VGG FCN 8 + 32×32 block-wise smoothening	FSA	50.70%	46.77%	50.50%	54.83%
		BSA	77.09%	78.54%	78.77%	73.96%
		SA	73.16%	74.29%	75.36%	69.81%
Exp. 7	Skin-color based segmentation + VGG SegNet + 32×32 block-wise smoothening	FSA	32.32%	32.51%	32.65%	31.79%
		BSA	89.78%	90.69%	90.66%	87.98%
		SA	81.37%	83.20%	84.15%	76.75%
Exp. 8	Skin-color based segmentation by Sankaran et al. [26]	FSA	6.48%	6.94%	6.63%	5.87%
		BSA	98.83%	98.84%	98.79%	98.84%
		SA	85.97%	87.75%	89.09%	81.05%

serve that BSA outperforms FSA in all the cases and for all the cases. It can be attributed to the fact that both VGG SegNet and FCN provide a very tight bound for the located finger. This results in some foreground pixels (finger regions) termed as background, whereas, most background pixels are predicted as background. Hence, BSA is high due to correct classification of background, FSA remains lower due to the incorrect classification of foreground pixels.

- While BSA is higher than FSA in all the experiment, the overall segmentation accuracy (SA) is closer to BSA. Overall, in the dataset, foreground pixels constitute 13.79%, compared to 86.21% background pixels. Hence, with higher correctly classified background pixels, overall segmentation accuracy (SA) is closer to BSA compared to FSA.

- In Exp. 1 and Exp. 3 (Table 3), it is observed that FSA is lower due to a tight bound. However, if the bound provided by FCN and SegNet can be made loose, FSA would increase. In the proposed architecture (Exp. 4), a 32×32 block-wise smoothing operation is performed. This makes the predicted masks looser, hence increasing the FSA significantly to 71.22% from 66.75%. However, in this process, BSA decreases by 1.98% and SA is decreased by 1.04%. A sample output from Exp. 3 and Exp. 4 is illustrated in Figure 9.



a) Original image b) Output: Exp. 3 b) Output: Exp. 4

Figure 9: Sample showing the significance of 32×32 block-wise smoothening post VGG SegNet.

- Motivated by existing literature [15, 24, 25, 26], we

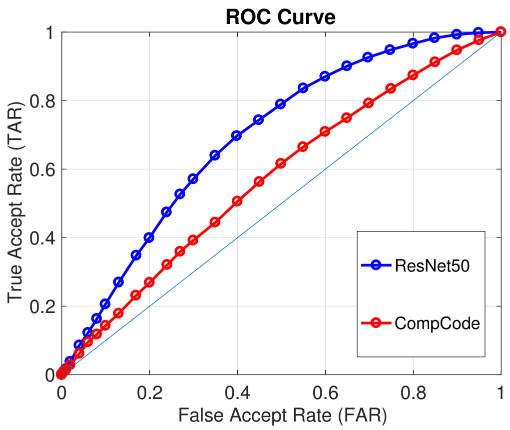


Figure 10: ROC curve for the proposed segmentation with ResNet50 and CompCode features on the UNFIT database.

report results with skin-color based segmentation on the proposed dataset in Exp. 5 (in Table 4). It is observed that though skin-color based segmentation yields around 58% FSA and it does not outperform the deep learning approach. This is due to the presence of illumination variations in fingerphoto images, because of which the skin regions becomes too bright or too dull in certain cases. Also, variations in illumination are also induced by the camera flash; in some cases, users utilized the camera flash, and in some, they did not.

- To combine skin-color segmentation with deep learning approach, we first find salient region using skin-color based segmentation. This region is cropped and given as input to VGG SegNet, followed by a 32×32 smoothing. However, the overall performance and FSA is reduced. The results are shown in Table 4 as Exp. 6 and Exp. 7. These results suggest that skin-color based segmentation is likely to fail on the proposed IIITD database.

- The segmentation for IIITD Smartphone Fingerphoto Database [26] is also performed using VGG SegNet [27]. Since the ground truth annotations are not available, it is difficult to report FSA, BSA, and SA for IIITD Smartphone Fingerphoto Database. However, it can be visually seen in Figure 11 that the deep learning algorithm worked well for the semi-constrained database.

The verification accuracy on the testing set of 57 subjects is computed using CompCode [13] Features followed by Hamming distance. The ROC curve in Figure 10 presents the baseline results. Despite the effectiveness of CompCode for fingerprint recognition, an Equal Error Rate (EER) of 44.35% is observed for fingerphoto matching. Similarly, an EER of 35.48% is observed when representation obtained

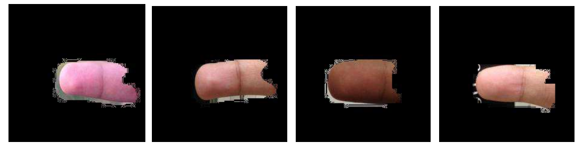


Figure 11: Sample output on the IIITD Smartphone Fingerphoto Database [26] using the deep learning based segmentation framework.

from ResNet50 [11] model is matched using cosine similarity for verification. Such a performance highlights the challenging nature of the proposed dataset. With the proposed dataset, the research community would be able to address the variations in fingerphotos and possibly propose improved quality assessment, segmentation, feature extraction, and matching algorithms.

5. Conclusion and Future Work

This paper presents an unconstrained fingerphoto database of 3450 images pertaining to 230 classes. The proposed database incorporates variations in terms of translation, rotation, scale, orientation, resolution, background, and illumination. The proposed database includes an experimental protocol, using which benchmarking is performed for segmentation and matching. For the proposed UNFIT database, a segmentation framework using VGGSegNet is presented which outperforms the algorithm proposed in [26] and the skin-color based segmentation algorithm. CompCode and ResNet50 based approaches show the challenging nature of the proposed database. Future work could potentially be (i) to include a quality assessment module to detect poor quality fingerphotos, and (ii) explore other popular features used in fingerprint and palmprint recognition such as minutiae features, Fast Compcode [30], and Fast-RLOC [30].

6. Acknowledgements

The authors acknowledge the active participation of the volunteers in the database collection. A. Malhotra is partially supported through Visvesvaraya PhD fellowship. M. Vatsa and R. Singh are partly supported by the Infosys Center of Artificial Intelligence, IIIT Delhi, India.

References

[1] Facebook Messenger. https://www.messenger.com/. Accessed: 2018-02-22. 3

[2] Google Drive. https://drive.google.com/. Accessed: 2018-02-22. 3

[3] Google Mail. https://mail.google.com/mail/. Accessed: 2018-02-22. 3

[4] MATLAB. https://www.mathworks.com/products/matlab.html. Accessed: 2018-02-22. 4

[5] Telegram. https://web.telegram.org. Accessed: 2018-02-22. 3

[6] WhatsApp. https://www.whatsapp.com/. Accessed: 2018-02-22. 3

[7] WhatsApp photo drug dealer caught by 'groundbreaking' work. http://www.bbc.com/news/uk-wales-43711477. Accessed: 2018-04-16. 1

[8] M. O. Derawi, B. Yang, and C. Busch. Fingerprint Recognition with Embedded Cameras on Mobile Phones. In *MobiSec*, pages 136–147, 2011. 4

[9] P. Dollár. Piotr's Computer Vision Matlab Toolbox (PMT). https://github.com/pdollar/toolbox. Accessed: 2018-02-22. 4

[10] FBI. Next Generation Identification. https://www.fbi.gov/services/cjis/fingerprints-and-other-biometrics/ngi. Accessed: 2018-02-17. 1

[11] K. He, X. Zhang, S. Ren, and J. Sun. Deep Residual Learning for Image Recognition. In *IEEE International Conference on Pattern Recognition*, 2016. 2, 6, 8

[12] S. Kolkur, D. Kalbande, P. Shimpi, C. Bapat, and J. Jatakia. Human Skin Detection Using RGB, HSV and YCbCr Color Models. In *arXiv:1708.02694*, 2017. 5

[13] A.-K. Kong and D. Zhang. Competitive coding scheme for palmprint verification. In *IAPR International Conference on Pattern Recognition*, 2004. 2, 6, 8

[14] A. Kumar and Y. Zhou. Contactless Fingerprint Identification using Level Zero Features. In *IEEE Computer Vision and Pattern Recognition Workshop*, 2011. 1, 2, 4

[15] C. Lee, S. Lee, J. Kim, and S.-J. Kim. Preprocessing of a fingerprint image captured with a mobile camera. In *IAPR International Conference on Biometrics*, 2005. 2, 4, 7

[16] D. Lee, K. Choi, H. Choi, and J. Kim. Recognizable-Image Selection for Fingerprint Recognition With a Mobile-Device Camera. In *IEEE Transactions on Systems, Man, and Cybernetics*, volume 38, pages 233–243, 2008. 2, 4

[17] G. Li, B. Yang, M. A. Olsen, and C. Busch. Quality assessment for fingerprints collected by smartphone cameras. In *IEEE Computer Vision and Pattern Recognition Workshop*, 2013. 2, 4, 6

[18] G. Li, B. Yang, R. Raghavendra, and C. Busch. Testing Mobile Phone Camera Based Fingerprint Recognition under RealLife Scenarios. In *Norsk informasjonssikkerhetskonferanse*, 2012. 2, 4

[19] J. Long, E. Shelhamer, and T. Darrell. Fully Convolutional Networks for Semantic Segmentation. In *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2016. 2, 4

[20] A. Malhotra, A. Sankaran, A. Mittal, M. Vatsa, and R. Singh. Fingerphoto Authentication Using Smartphone Camera Captured Under Varying Environmental Conditions. In *Human Recognition in Unconstrained Environments,*, pages 119–144, 2017. 2, 4

[21] A. Malhotra, A. Sankaran, M. Vatsa, and R. Singh. Learning Representations for Unconstrained Fingerprint Recognition. In *Deep Learning in Biometrics*, page 197. CRC Press, 2018. 4

[22] F. Mather. 4F allows the use of smartphone finger photos as a contactless fingerprint identification system to match with legacy databases. http://www.biometricupdate.com/201601/4f-allows-the-use-of-smartphone-finger-photos-as-a-contactless-fingerprint-identification-system-to-match-with-legacy-databases. Accessed: 2018-02-17. 1

[23] R. Mueller and R. Sanchez-Reillo. An Approach to Biometric Identity Management Using Low Cost Equipment. In *IEEE Intelligent Information Hiding and Multimedia Signal Processing*, 2009. 4

[24] V. Piuri and F. Scotti. Fingerprint biometrics via low-cost sensors and webcams. In *IEEE Conference on Biometrics: Theory, Applications, and Systems*, 2008. 2, 4, 7

[25] R. Raghavendra, C. Busch, and B. Yang. Scaling-robust fingerprint verification with smartphone camera in real-life scenarios. In *IEEE Conference on Biometrics: Theory, Applications, and Systems*, 2013. 2, 4, 7

[26] A. Sankaran, A. Malhotra, A. Mittal, M. Vatsa, and R. Singh. On smartphone camera based fingerphoto authentication. In *IEEE Conference on Biometrics: Theory, Applications, and Systems*, 2015. 1, 2, 4, 6, 7, 8

[27] K. Simonyan and A. Zisserman. SegNet: A Deep Convolutional Encoder-Decoder Architecture for Image Segmentation. In *arXiv:1511.00561v3*, 2016. 2, 4, 8

[28] C. Stein, C. Nickel, and C. Busch. Fingerphoto recognition with smartphone cameras. In *IEEE International Conference of Biometrics Special Interest Group*, pages 1–12, 2012. 2, 4

[29] K. Tiwari and P. Gupta. A touch-less fingerphoto recognition system for mobile hand-held devices. In *IAPR International Conference on Biometrics*, 2015. 2, 4

[30] Q. Zheng, A. Kumar, and G. Pan. Suspecting less and doing better: New insights on palmprint identification for faster and more accurate matching. *IEEE Transactions on Information Forensics and Security*, 11(3):633–641, 2016. 8