A Camera-Based Fingerprint Registration and Verification Method

Sajid Khan[†], Ahmad Waqas[†], Muhammad Asif Khan[†] and Arbab Waheed Ahmad[††]

[†]Sukkur Institute of Business Administration, Sukkur, Sindh, Pakistan
[††]Hanyang University, ERICA Campus, Ansan, Gyonggi-do, South Korea

Abstract
Many researchers worked on detection and verification of fingerprints that involves sensor or fingerprint detection device. However, due to lack of availability of sensors and such devices in remote areas leads to alternative and cost-effective solutions for such purpose. Using of ordinary camera or webcam for such purpose provides cost-effective solution as nowadays, most of the smartphones comes with digital cameras of reasonable image quality. This paper proposes a fingerprint registration and verification technique that involves pre-processing of digital image in order to convert it into binary image. The binary image have similar characteristics when compared to the output of scanner. Therefore, preprocessing is followed by the techniques of feature detection such as ridges and bifurcation. Those features are either stored in database in case of registration process whereas are compared with features of stored fingerprints in case of verification. Proposed method is a cost-effective solution for fingerprint registration and verification at the cost only few additional steps of image preprocessing.

Key words:
Digital camera, fingerprint, verification, registration, biometric.

1. Introduction

With the advancement of technology, security of devices, buildings and organizations become an issue. Whether it is allowing an authorized person to enter a specific building or allowing a person to use a service or facility, registration of that person is needed in order to track and minimize suspicious activities. For such purpose, biometric registration and verification methods that involves image processing techniques are used.

Automatic biometric recognition is a widely addressed area of image processing. Biometric recognition involves face detection, retina and iris detection and fingerprint detection [1-5].

Face verification [6-7] involves storing a face in a database during registration process whereas comparing a face with hundreds or thousands of images in database. Since face verification process involves storing and comparing of so many images, it is computationally expensive. Also, factors such as illumination, image scale, posture, eyes and mouth opening and closing affect the performance of an algorithm and demands complex computation.

IRIS recognition and verification [8-10] involves segmentation of IRIS radius followed by feature extraction and comparison. It is widely used at airports and other high-risk locations along with face detection and fingerprint verification. Although IRIS verification devices embedded with efficient algorithms provides good performance, however, the devices are far more expensive to be used extensively.

Fingerprint verification algorithms [11-15] on the other hand, involves extraction of features such as bifurcation and ridges of a fingerprint that is obtained using scanner. The magnitude and orientation of these features are then stored in database during registration whereas compared with existing features in case of verification. Fingerprint scanners are the most cost-effective [16] options currently and usually 3-10 times cost-effective compared to IRIS scanners.

Many researchers [11-21] worked on proposal of algorithms and applications of fingerprint verification methods. Trend is to rely on fingerprint scanners to obtain input fingerprint. For example, Mahajan et al. [17] and Priya et al. [18] proposed secure banking systems that use fingerprint verification along with username and password since using username and password is not secure. But such system have a drawback that a user cannot access his/her account without the use of a scanner and hence reducing accessibility.

Some of the researchers worked on making fingerprint sensors more secure. Such as Baldissera et al. [19] proposed to use odor sensor that use odor analysis to detect a fake fingerprint. Such method, although improve accuracy, however, an additional sensor increases the cost further. Nikam and Agarwal [20] proposed use of ridgelet transform to perform point-to-point or point-to-line mapping in order to detect and discard fake fingerprints. However, ridge transforms perform better for straight lines, however, fingerprint contains edges that are curvy instead of straight lines. Choi et al. [21] proposed fake fingerprint detection by assuming some properties of fake fingerprint such as Broken ridges, presence of noise and thick edges. However, broken ridges might be present in original fingerprints due to scanning issues whereas presence of noise might be because of faulty scanners. Also, thickness of edges sometime depends on amount of pressure applied on sensor

Manuscript received November 5, 2018
Manuscript revised November 20, 2018

while providing fingerprints. Therefore, assuming such properties while trying to discard fake fingerprints will often leads to detection of original fingerprints as fake.

Researchers worked on identification of problems and their solutions to propose efficient fingerprint detection and registration techniques, however, usage of sensor for getting input fingerprints decreases the factor of availability as fingerprint scanners are not widely available. Therefore, this paper proposes a technique that use image of finger captured using any digital camera, apply pre-processing techniques such as image enhancement, segmentation, morphological filtering and binarization in order to convert image captured using digital camera into equivalent to the output of a scanner. Finally detection of features and comparison is done using conventional extraction and comparison techniques. Simulation results show that proposed method provide accuracy closer to the state-of-the art methods whereas unlike other methods, it doesn't require a fingerprint sensor.

The rest of this paper is organized as follows: Section 2 describe some of the conventional methods used for detection of fingerprints through sensor, Section 3 discusses the proposed methodology and Section 4 is the conclusion.

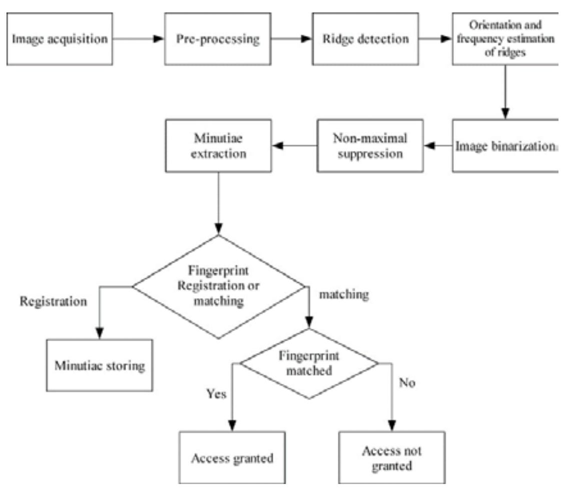

Fig. 1 Flowchart of the proposed algorithm

2. Methodology

Since the objective of proposed method is to provide a cost-effective alternative to fingerprint scanner, therefore, some pre-processing steps before extraction of features. Fig. 1 show the flowchart of the proposed method. Input image is subjected to application of some pre-processing steps. Pre-processing is followed by ridge detection. Orientation and frequency estimation of detected ridges is performed that is followed by image binarization, morphological thinning and feature extraction. Finally, input fingerprint is either stored or compared according to the requirement. All the steps shown in the flowchart are explained in details in upcoming subsections.

2.1 Image Acquisition

Instead of using scanners, any kind of digital camera can be used for acquiring images for using them as input. For this paper, images were acquired using ordinary webcam.

2.2 Pre-Processing

Input fingerprint image is of RGB format that do have three layers. Each layer have different intensity values, however, structure of edges and other details is same in all three layer. Instead of using RGB color image, input image is converted into grayscale image in order to improve performance efficiency. In grayscale image, only a single layer needs to be processed unlike three layers of RGB image. An image can be converted using (Eq. 1) as follows

$$I_{gray} = I_R * w_R + I_G * w_G + I_B * w_B \qquad (1)$$

Where $w_R + w_G + w_B = 1$. Since bayer pattern have two block representation of green layer in each 2x2 block [22-23] whereas red and blue layer have single block representation, therefore, w_G should be assigned maximum weight compared to w_B and w_R. Also, since humans are more sensitive to red color, therefore, w_R should optionally have more weight compared to w_B.

After converting image to grayscale, edge based image enhancement is applied in order to get a clear image that contains fine and enhanced details. Since sometime the fingerprint doesn't have visible fine details, therefore, edge based enhancement is done in two steps. Firstly, image enhancement method named unsharp masking [24] is applied to the image that involves in subtracting an image with a blurred version of itself and enhancing those regions that provide good responses after subtraction. For calculation of blurred image, Gaussian filter is convolved with original grayscale image using (Eq. 2) as

$$I_{gray} * G_\sigma \qquad (2)$$

Where

$$G_\sigma = \frac{1}{\sqrt{\pi}\,\sigma} e^{\frac{-x^2}{2\sigma^2}} \qquad (3)$$

After performing unsharp masking to get image $I_{unsharp}$, contextual block-wise fourier domain [25] is applied by dividing image into 16x16 overlapping block followed by filtering these blocks in the fourier domain by a frequency and orientation selective filter whose parameters are based on the estimated local ridge orientation and frequency. The

outcome of fouried domain based enhancement is another image named $I_{enhanced}$.
The last step of image pre-processing is image normalization. The enhanced image is normalized using (Eq. 4) as

$$I_{norm} = \frac{I_{max} - I_{enhanced}}{I_{max} - I_{min}} \quad (4)$$

Results of pre-process are shown in Fig. 2. Input fingerprint image is captured using ordinary camera is shown in Fig. 2(a) which have visible fingerprint lines but not with much clarity. Fig. 2(b) show result of conversion of image into grayscale. Fig. 2(c) show application of unsharp masking results in clear visibility of fingerprint lines whereas Fig. 2(d) show that application of fourier based image enhancement further clears the unclear edges and other details. Fig. 2(e) show result of application of normalization of image to convert image into scale of 0-1.

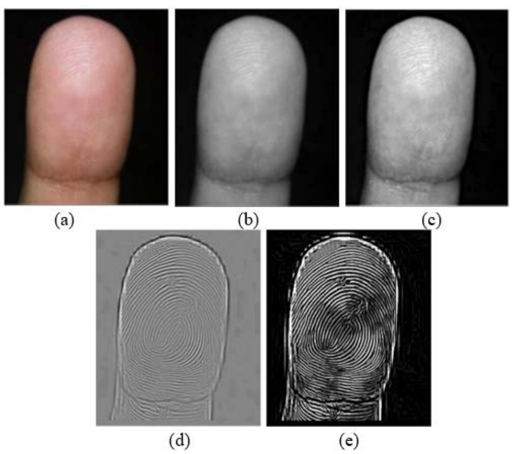

Fig. 2 Result of applying pre-processing on fingerprint image: (a) original input image, (b) grayscale conversion, (c) unsharp masking, (d) block-wise fourier based enhancement, (e) Normalization of enhanced image.

2.3 Ridge Detection

Pre-processing of input fingerprint results in clear information about required details of fingerprint. Ridge detection step is followed after pre-processing detection that actually is based on calculation of response of each pixels for belonging to a ridge or not. For this purpose, gradients of normalized image are calculated by applying sobel operator [26] to get G

$$G_x = I_{norm} * S_x \quad (5)$$

$$G_y = I_{norm} * S_y \quad (6)$$

2.4 Orientation and Frequency Estimation of Ridges

After convolving normalized image to obtain gradients along x and y directions, magnitude and orientation of ridge is calculated using (Eq. 7) and (Eq. 8) respectively

$$M = \sqrt{G_x^2 + G_y^2} \quad (7)$$

$$\theta = \tan^{-1}\left(\frac{G_y}{G_x}\right) \quad (8)$$

In order to estimate the ridge frequency that I an important property, ridge frequency can be estimated using method provided by Babatunde et al. [27] as it is immune to noise, low contrast and different resolution.
Babutunde et al. [27] divides image into QxQ blocks and estimate local orientation of the center pixel of each block using gradients along x and y direction. It then calculates uniformity level of each orientation field. It ensure the calculation of uniformity level above a certain threshold F_C and keep calculating uniformity level by using low resolution block in case it is below F_C. It then apply two adaptive filters to the image and if result of both adaptive filters for a certain pixel is above a threshold F_{Ridge}, it is marked as ridge pixel.

2.5 Image Binarization

Image binarization involves applying a certain threshold to magnitude of gradients of an image. Since an image have different variances and gradients along edges, it is not a good option to apply same threshold to all images. Therefore, some efficient threshold calculation criteria should be used in order to provide good automatic threshold calculation. Rosin [28] provided an efficient histogram based threshold calculation method that is simple as well. Idea of calculation of threshold using Rosin method can be easily explained using Fig. 3.

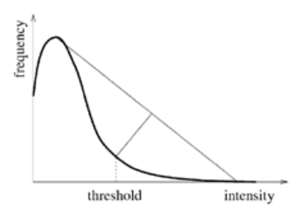

Fig. 3 Rosin method to determine threshold from gradient magnitude histogram.

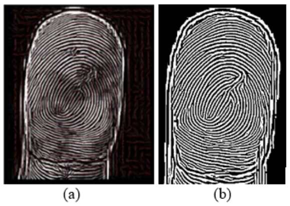

Fig. 4 Results after applying different steps on normalized image: (a) Ridge orientation, (b) Image binarization.

Rosin draw a straight line that connect peak of histogram of gradient magnitude with the first zero bin at the tail of histogram. The bin that provides maximum perpendicular distance to the line is selected as the threshold. Fig. 4(a-b) show results of application of ridge orientation calculation and image binarization.

2.6 Non-Maximal Suppression Based Edge Thinning

It is clear from Fig. 4(b) that image binarization provide information of most of the important ridges, however, in order to extract features such as minutiae, binarized image with thick edges cannot be used as thick edges doesn't provide good clues about their directions, endings or junctions. Therefore, there is need of conversion of thick edges into edges of one pixel thickness(thin edges). Canny [29] use an edge thinning method called non-maximal suppression that uses directions of gradients and suppresses all those edge pixels that don't have maximum value along that direction by assigning them 0 values while retaining the maximum one. Therefore, for a thick edge, it retains the one that have maximum gradient values while suppressing others. Fig. 5(a) show the result of application of non-maximal suppression based edge thinning. It can be seen from the image that edges that represent ridges are having thickness of one pixel and are ready to be used for minutiae extraction.

Fig. 5 Results after applying different steps on binary image: (a) Non-maximal Suppression, (b) Minutiae Extraction.

2.7 Minutiae Extraction

Minutiae are the features that are used to either store the information of a fingerprint or are used for comparison. Minutiae can be categorized into many types, however, ridge ending or ridge bifurcation are considered in this work. Ridge bifurcation is the point where a single ridge branches out into two or more ridges whereas ridge ending is the ending of a ridge.
For extracting minutiae, a 3x3 window is moved across each pixel to calculate its cross number using (Eq. 9) as

$$CN(p) = \frac{1}{2}\sum i = 1\ to\ 8 |\ val(p_i\ mod\ 8) - val(p_{i-1}) \quad (9)$$

Where p_i denotes the pixel around the centering pixel p. Spatial coordinates (x, y), orientation $\propto$ and cross number CN for each minutiae are also extracted for comparison or storing purpose. Fig. 5(b) show labeling of bifurcation and ridge ending. Blue squares represent ridge bifurcation whereas red squares represent ridge ending.

2.8 Fingerprint Registration and Matching

Extraction of minutiae is followed by fingerprint registration or matching according to our need. For registration purpose, all the relevant information extracted during of extraction of minutiae is stored in a database. Other information, such as person's id, name or image can be also stored in the database with certain label or hash key. Fingerprint matching, on the other hand is not that simple process. For fingerprint matching, the minutiae of input fingerprint is matched with minutiae of all the stored fingerprints in the database. Suppose that minutiae of input fingerprints are stored in matrix X_1 whereas it is currently compared with minutiae X_i in the database. Firstly, N_{min} is determined using (Eq. 10) as

$$N_{min} = \min(N_1, N_i) \quad (10)$$

Where N_1 and N_i are number of extracted minutiae for X_1 and X_i respectively. If at least $N_{min} \times t_{match}$ minutiae of both fingerprint matches, then both the fingerprints are said to be of same person. Otherwise, they are said to be unmatched. Where t_{match} is a threshold that preferably may have value in the range of 0.8-0.9.
It is suggested to align fingerprint in vertical direction while capturing its picture, however, input fingerprint sometimes isn't perfectly aligned along vertical axis. Therefore, it is suggested to rotate the input fingerprint with rotation angles of $-2°, -1°, 0, 1°, 2°$ and compare it with the fingerprints in the database.

3. Results and Experiments

For experiment purpose, Matlab 2016a was used. Values of $w_G = 0.59, w_B = 0.18$ and $w_R = 0.23$ was used in (1) whereas $\sigma = 3$ where Gaussian of window size $(2\sigma + 1 \mathrm{x} 2\sigma + 1)$ was used in unsharp masking.
For experiment purpose, total of 30 persons of age groups from 19-40 years were registered in databse whereas 100 persons including 30 registered persons along with 70

unregistered persons images of fingerprint with some clarity were taken along vertical axis with possible rotation of $-2°$ to $2°$. The storage process was done successfully as there was no comparison involved, however, fingerprint matching process needed some performance evaluation criteria.

Table 1: Accuracy of proposed method for different values of t_{match}.

t_{match}	.7	.75	.8	.85	.9	.95	1
TP	100	100	100	100	100	.97	.9
TN	100	100	100	100	100	100	100
FP	0	0	0	0	0	0	0
FN	0	0	0	0	0	1.408	4.11

True positives, true negatives, false positives and false negatives were calculated for different values of matching thresholds t_{match} . True positives, true negatives, false positives and false negatives are given in (11-14) as

$$TP = \frac{n_{trueDetection} \times 100}{n_{trueActual}} \quad (11)$$

$$TN = \frac{n_{falseDetection} \times 100}{n_{falseActual}} \quad (12)$$

$$FP = \frac{n_{mistakenTrue} \times 100}{n_{total}} \quad (13)$$

$$FN = \frac{n_{mistakenFalse} \times 100}{n_{falseDetection}} \quad (14)$$

Where $n_{trueActual}$, $n_{falseActual}$ and n_{total} were 30, 70 and 100 respectively as registered fingerprints were 30 whereas 100 fingerprints were used for experiment purpose.
Table 1 show TP, TN, FP and FN calculation for different values of matching threshold t_{match}. It can be seen that for value of $t_{match} > 0.9$, 1-3 persons that were registered in database weren't matched to any of the fingerprint stored in the database. It is because images usually are not clear enough to extract more than 90% of bifurcation and ridges compared to the bifurcation and ridges of corresponding actual stored fingerprints. However, for values of $t_{match} \leq 0.9$, matching accuracy is up to 100% whereas no FP were detected for all values of t_{match}. Very small values of t_{match} guarantees 100% detection of TP, however, they might result in detection of FP as well and therefore, as a precaution, are not recommended.

4. Conclusions

This paper proposed a system that does not need a fingerprint or any other sensor for the registration and verification of fingerprints. The proposed system recommend using any ordinary digital camera for capturing images of fingerprint with some clarity of ridges. Input fingerprint image is not always perfect and sometime misses clear visibility along ridges, therefore, image enhancement techniques such as unsharp masking and fourier based blockwise enhancements are applied as the pre-processing steps. Techniques such as ridge detection, image binarization and edge thinning are followed by bifurcations and ridge endings detections. Those features are stored in database during fingerprint registration whereas are compared with features of stored fingerprints during fingerprint comparison process. Simulation results shows that detection accuracy of 100% can be achieved when matching threshold is in suggested range.

Acknowledgments

This research is funded by Higher Education Commission (HEC) Pakistan under the project of Startup Research Grant Program (SRGP).

References

[1] M.J. Burge, K.W. Bowyer, Handbook of Iris Recognition, Springer Publishing Company, Incorporated, 2013.
[2] Moon, D. S., Moon, K. Y., Jang-Hee, Y. O. O., Chung, Y. S., Choi, W. Y., Park, S. H., ... & Lee, Y. J. (2014). U.S. Patent No. 8,699,799. Washington, DC: U.S. Patent and Trademark Office.
[3] Guesmi, H., Trichili, H., Alimi, A. M., & Solaiman, B. (2015). Fingerprint verification system based on curvelet transform and possibility theory. Multimedia Tools and Applications, 74(9), 3253-3272.
[4] Raja, K. B., Raghavendra, R., Vemuri, V. K., & Busch, C. (2015). Smartphone based visible iris recognition using deep sparse filtering. Pattern Recognition Letters, 57, 33-42.
[5] Kaur, E. G., Kaur, E. H., & Kaur, E. M. (2017). A Survey on Face Recognition Techniques. International Journal of Advanced Research in Computer Science, 8(4).
[6] Parkhi, O. M., Vedaldi, A., & Zisserman, A. (2015, September). Deep Face Recognition. In BMVC (Vol. 1, No. 3, p. 6).
[7] Taigman, Y., Yang, M., Ranzato, M. A., & Wolf, L. (2014). Deepface: Closing the gap to human-level performance in face verification. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 1701-1708).
[8] Rai, H., & Yadav, A. (2014). Iris recognition using combined support vector machine and Hamming distance approach. Expert systems with applications, 41(2), 588-593.
[9] Raja, K. B., Raghavendra, R., Vemuri, V. K., & Busch, C. (2015). Smartphone based visible iris recognition using deep sparse filtering. Pattern Recognition Letters, 57, 33-42.
[10] Ismail, A. I., Ali, H. S., & Farag, F. A. (2015, February). Efficient enhancement and matching for iris recognition using SURF. In Information Technology: Towards New Smart World (NSITNSW), 2015 5th National Symposium on (pp. 1-5). IEEE.
[11] Kovacs-Vajna, Z. M. (2000). A fingerprint verification system based on triangular matching and dynamic time warping. IEEE Transactions on Pattern Analysis and Machine Intelligence, 22(11), 1266-1276.
[12] Luo, X., Tian, J., & Wu, Y. (2000). A minutiae matching algorithm in fingerprint verification. In Pattern Recognition,

2000. Proceedings. 15th International Conference on (Vol. 4, pp. 833-836). IEEE.
[13] Cappelli, R., Maio, D., Maltoni, D., Wayman, J. L., & Jain, A. K. (2006). Performance evaluation of fingerprint verification systems. IEEE transactions on pattern analysis and machine intelligence, 28(1), 3-18.
[14] He, Y., Tian, J., Luo, X., & Zhang, T. (2003). Image enhancement and minutiae matching in fingerprint verification. Pattern recognition letters, 24(9-10), 1349-1360.
[15] Hong, L., Wan, Y., & Jain, A. (1998). Fingerprint image enhancement: Algorithm and performance evaluation. IEEE transactions on pattern analysis and machine intelligence, 20(8), 777-789.
[16] Delac, K., & Grgic, M. (2004, June). A survey of biometric recognition methods. In Electronics in Marine, 2004. Proceedings Elmar 2004. 46th International Symposium (pp. 184-193). IEEE.
[17] Mahajan, P., Malekar, M., More, A., Wairagade, A., & Mahalakshmi, B. (2016) Secured Internet Banking Using Fingerprint Authentication. International Journal of Innovative Research in Computer and Communication Engineering, 4(3), 6199-6208.
[18] Priya, R., Tamilselvi, V., & Rameshkumar, G. P. (2014, July). A novel algorithm for secure Internet Banking with finger print recognition. In Embedded Systems (ICES), 2014 International Conference on (pp. 104-109). IEEE.
[19] Baldisserra, D., Franco, A., Maio, D., & Maltoni, D. (2006, January). Fake fingerprint detection by odor analysis. In International Conference on Biometrics (pp. 265-272). Springer, Berlin, Heidelberg.
[20] Nikam, S. B., & Agarwal, S. (2009). Ridgelet-based fake fingerprint detection. Neurocomputing, 72(10-12), 2491-2506.
[21] Choi, H., Kang, R., Choi, K., Jin, A. T. B., & Kim, J. H. (2009). Fake-fingerprint detection using multiple static features. Optical Engineering, 48(4), 047202.
[22] Bosco, A., & Mancuso, M. (2008). U.S. Patent No. 7,369,165. Washington, DC: U.S. Patent and Trademark Office.
[23] Lukac, R., Plataniotis, K. N., & Hatzinakos, D. (2005). Color image zooming on the Bayer pattern. IEEE Transactions on Circuits and Systems for Video Technology, 15(11), 1475-1492.
[24] Ramponi, G., Strobel, N. K., Mitra, S. K., & Yu, T. H. (1996). Nonlinear unsharp masking methods for image contrast enhancement. Journal of Electronic Imaging, 5(3), 353-367.
[25] Chikkerur, S., Wu, C., & Govindaraju, V. (2004). A systematic approach for feature extraction in fingerprint images. In Biometric Authentication (pp. 344-350). Springer, Berlin, Heidelberg.
[26] Gao, W., Zhang, X., Yang, L., & Liu, H. (2010, July). An improved Sobel edge detection. In Computer Science and Information Technology (ICCSIT), 2010 3rd IEEE International Conference on (Vol. 5, pp. 67-71). IEEE.
[27] Babatunde, I. G., Charles, A. O., & Olatunbosun, O. (2013). Uniformity Level Approach to Fingerprint Ridge Frequency Estimation. International Journal of Computer Applications, 62(22).
[28] Rosin, P. L. (2001). Unimodal thresholding. Pattern recognition, 34(11), 2083-2096.
[29] Canny, J. (1986). A computational approach to edge detection. IEEE Transactions on pattern analysis and machine intelligence, (6), 679-698.

Sajid Khan received the B. S. degree in Telecom Engineering from the FAST-NUCES University, Pakistan, in 2011 and M. S leading to Ph. D. degree in Electronics and Communication Engineering from Hanyang University, Ansan, South Korea in 2017.
From 2011 to 2012, he was a Software Engineer at Gameview Studios, Pakistan. Since 2017, he has been as Assistant Professor with the Computer Science Department, Sukkur IBA University, Sukkur, Pakistan. His research interest include image denoising, edge detection, interpolation, deinterlacing, fingerprint detection and biomedical image processing.

Ahmad Waqas is an Associate Professor in the Department of Computer Science, Sukkur IBA University Pakistan. He has obtained his PhD from the Department of Computer Science, Faculty of Information and Communication Technology, International Islamic University Malaysia. Prior to that, he got his MS (Computer Communication and Networks) from Sukkur IBA Pakistan, and MCS (Master in Computer Science) from University of Karachi in with second position in faculty. His area of research interest is Distributed Computing and Cloud Computing Security.

Muhammad Asif Khan received the Bachelor of Engineering degree in Computer Systems from QUEST University, Pakistan, in 2007 and M. S from Sukkur IBA University in 2012. Currently he is employed as Assistant Professor in Sukkur IBA University Department of Electrical Engineering since 2013. His research interest include Computer vision for Robots Navigation, edge detection, interpolation.

Arbab Waheed Ahmad received the B.Sc. degree in electrical engineering from the NWFP University of Engineering and Technology, Peshawar, Pakistan, in 2008, the M.S. degree in electronics, electrical, control, and instrumentation engineering and the Ph.D. degree in electronics and communications engineering from Hanyang University, Seoul, South Korea, in 2012 and 2016, respectively. His research interests include on scheduling (power control and resource allocation) and location awareness in modern wireless communication technologies, i.e., 5G communication.