JCBI
Journal of Computing & Biomedical Informatics
ISSN: 2710 - 1606
Volume 08 Issue 01
2024
Research Article
https://doi.org/10.56979/801/2024

A lightweight Approach to Finger Vein Authentication based on Contrast-Limited Adaptive Histogram Equalization

Hedi A. Guesmi[1]

[1]Department of Electrical Engineering, College of Engineering, Qassim University, Saudi Arabia.
*Corresponding Author: Hedi A. Guesmi. Email: h.guesmi@qu.edu.sa

Received: August 11, 2024 **Accepted:** October 21, 2024

Abstract: Finger vein recognition is an emerging field that holds a lot of promise for security-sensitive applications. However, most of the past work in this field is dominated by computationally ex- pensive and complex algorithms. In this paper, a new yet simple technique is presented for the preprocessing of finger vein images for biometric authentication. Adaptive histogram equalization, gamma correction, image sharpening, multi-filtering, and contrast adjustment constitute the basic steps of this method [AHistGSFC]. For feature extraction, the Histogram of Oriented Gradients [HOG] is used, and the K-nearest neighbor for the recognition. The main contribution of this research is the introduction of a new preprocessing algorithm that is not only efficient but also has a very low computational time and has given superior results to deep learning approaches on the same data set. 6 fold cross-validation was used for evaluation. The middle fingers of both hands were found to be the most discriminative, giving 99.06% accuracy and 0.0086% EER. Using all fingers provided in the database, we get 96.96% accuracy and 0.0304% EER. Results were also computed for 3-fold and 2-fold cross-validations for comparison with previous works with that data division. The proposed method gives better results than the latest state-of-the-art algorithms in this field.

Keywords: Finger Vein; Authentication; Pattern Recognition; Classification; K-Nearest Neighbor; Histogram of Gradients; SDUMLA Database; CLAHE; Un-sharp Mask; Gamma Correction.

1. Introduction

Most commonly employed biometrics [face, hand geometry, fingerprint, iris, gait, signature, voice, keystroke pattern] are prone to forgery and intruder at- tacks [1; 2]. This has led to the demand for a more secure yet user-friendly biometric authentication system [3]. Vein [finger [4], wrist 5, and palm [6]] authentication is an emerging technique in this field. Vein data is more difficult to forge or obtain without consent. Vein images are acquired by near-infrared optical imaging. Infrared light is used to illuminate the part of the body with required vein patterns. It may be a finger, wrist, hand dorsal side, or other parts. Hemoglobin in the blood absorbs infrared radiation. This makes the veins appear slightly darker than the surrounding muscle background [7].

In finger vein recognition, the prevailing feature extraction methods can be divided into two types [8] non-vein pattern-based and vein pattern-based techniques. The former extracts features from the finger vein image as a whole, and the latter seeks to extract the exact shape of the vein architecture.

Initially developed algorithms in this domain faced difficulty dealing with low- quality images and inadequately designed finger vein sensor devices cite9. Most existing methods face limitations linked to weaknesses in preprocessing algorithms or feature extraction steps. In the case of a naıve preprocessing stage, a sophisticated feature extraction algorithm must be deployed, which can put a computational strain on the system. On the other hand, if the preprocessing stage is efficient, then that can give a clear image of the vein pattern, and even simple feature extraction algorithms can work well. It is a point of great intricacy to strike just the right balance between the sophistication of the preprocessing stage and its computational efficiency/speed. This is what we have aimed for in this research. Our prepossessing algorithm consists of

6 steps. Each step is computationally very efficient, and the complete procedure just takes a fraction of a second, rendering it highly feasible for practical deployment. The results we have achieved are comparable to those obtained by the latest works involving computationally heavy processing, such as deep learning, where the model needs to learn very slowly to converge into the minor details for each category [3].

This research aims to develop a reliable system for finger vein identification and verification that uses computationally light algorithms for image preprocessing, feature extraction, and classification/verification. Experimental results show state-of-the-art results are attained using the proposed preprocessing algorithm with simple low-level feature extraction, Histogram of Gradients [HOG], and with the most basic of classifiers, i.e., K- Nearest Neighbor [KNN].

2. Literature Review

Several personal identity verification methods have been utilized using biometric verification in the research community, including recognition by extrinsic features like fingerprint [45], palm, iris, and face [46] or by adapting verification by way of utilizing fundamental characteristics like the vein of a finger, hand, palm. However, verification using a finger vein or hand or palm vein propounds a progressing level of security as they are difficult to duplicate or to be used without the users' permission. Vein images are often taken using near-infrared- based optical imaging techniques [47] as the hemoglobin present in the blood absorbs the infrared rays showing the veins as a unique black web of veins distinctive to every user.

Different machine learning approaches have been used for finger vein recognition, such as K-nearest neighbor, fuzzy logic [48], and neural networks. These algorithms perform feature extraction and matching of the biometric after processing the feature dimensions. Although these approaches have demonstrated effective performance, they proved to be efficient in both feature extraction and processing part and match the finger vein biometric of the user.

Still, in most cases, the methods used to verify or match users' biometrics [49] are conventional and have a noticeable error rate compared to modern state-of- the-art machine learning techniques. On the other hand, conventional methods when discussing biometric verification tend more to be hacked or offer less security to the user as the error rate could differ depending on the approach hence leaving a blank space for the danger of duplication of the user's personal bio- metric without his/her consent.

Following the success of machine learning algorithms [50] in the biometric identification field, deep learning [51] has also been used as it offers an enormous level of accuracy, recall, and precision but jeopardizes the computational complexity as it uses many layers of filters for a different kind of process such as feature extraction resulting in slow processing in comparing with machine learning algorithms. In the deep learning approach, convolutional neural networks [CNN] [52] were used to process the finger vein images in both feature extraction and matching process, which gave a quite superior accuracy approaching almost 100%, this accuracy percentage endorses to adopt deep learning methods in different biometric identification as these algorithms demonstrated to be state of the art when discussing regarding accuracy precision and recall combinations.

3. Dataset

Many researchers have tested their finger-vein authentication algorithms on in- house datasets, making it difficult to compare with new research [3]. We, there- fore, used a publicly available database of finger vein images to test our algorithm. There are many open-source databases of a finger vein, but nearly all researchers agree that the most difficult one concerning preprocessing is the SDUMLA-HMT database [10; 11]. This has been attributed to many low- quality images and the small finger area in the images [12]. It has many non- ideal images, and for each finger, varying degrees of translation, shift, and rotation exists between samples captured [13]. Images here are captured in an uncontrolled way, thus making it more challenging to conquer [10].

This extensive database consists of different types of biometric data, for example, face, finger vein, fingerprint, iris, and gait. A total of 106 persons participated in data volunteering. Images are encoded in"BMP" format with a resolution of 320 × 240. The total size of the finger vein database is around 0.85 GB. Some samples images from Table 1 SDUMLA-HMT database are shown in Figure 1.

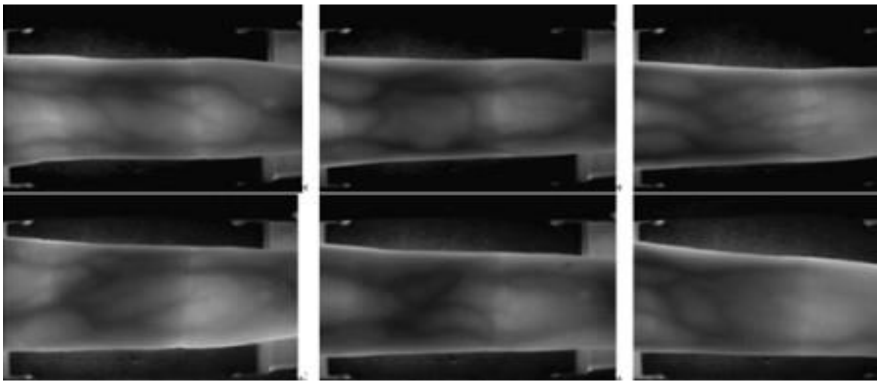

Figure 1. Sample images from the SDUMLA-HMT database

Table 1. SDUMLA-HMT database

Stats	SDUMLA-HMT database
Number of persons	106
Number of hands	2
Number of fingers	3
Number of images per finger	6
Total classes if by considering each person as a class	106
Total classes if we consider each finger as a class	636

4. Methodology

The first step toward finger vein authentication is preprocessing the raw database image. In this section, first, we will discuss the preprocessing steps in detail, then the feature extraction, and finally, the classification and verification stages. We will discuss all preprocessing steps and visualize them following the sample image from the SDUMLA database shown in Figure 2.

4.1. Preprocessing

The basic steps in the preprocessing are Adaptive histogram equalization, gamma correction, sharpening of the image, filtering, and contrast adjustment. So for simplicity, we call it the AHistGSFC algorithm.

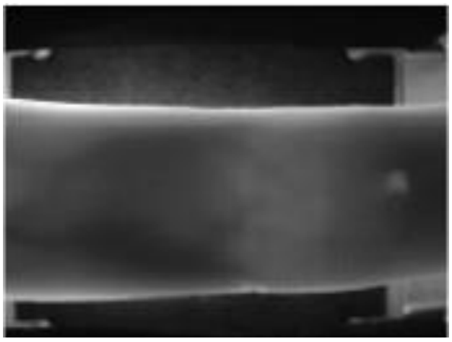

Figure 2. Image from SDUMLA database

- **Step 1: Detection and cropping of the finger region**

The approach mentioned in [15] is used to localize the finger region using a mask filter. The Finger region was normalized, and texture was extracted from it. It is based on the principle that the finger region is brighter than the back- ground region since infrared light passes through it. The mask used to segment the finger region is shown in Figure 3.

For each x-position, the masking value is calculated in the y-direction. The point where the masking value gets maximum is the boundary of the finger in the y-direction [16]. After the finger region is localized using masks, we crop the detected finger as depicted in Figures 4 and Figure 5.

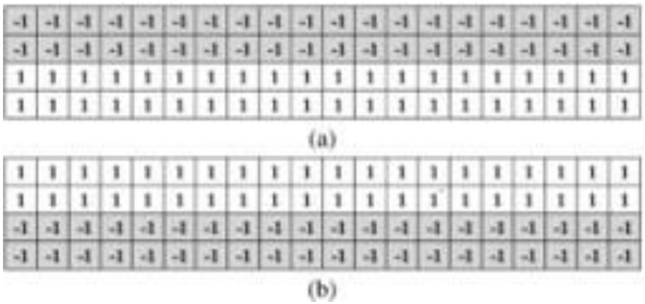

Figure 3. Masks for localizing the finer regions of the captured images. [a] Mask for detecting the upper region of the finger. [b] Mask for detecting the lower region of the finger.

Figure 4: Finger mask

Figure 5: Cropped detected finger

- **Step 2: Contrast-Limited Adaptive Histogram Equalization [CLAHE]**

CLAHE is used as a next step, we perform Contrast-Limited Adaptive Histogram Equalization [CLAHE] shown in Figure 6. [17] With uniform distribution on the detected finger region. Simple adaptive histogram equalization [AHE] may result in noise over-amplification. CLAHE works to prevent that [18]. CLAHE tunes the contrast of an image to bring out the minute details hidden in darkened pixels. CLAHE is robust in dealing with non-uniform light conditions, especially in biomedical images [19; 20; 21].

Figure 6. Contrast-Limited Adaptive Histogram Equalization

- **Step 3: Gamma Correction**

Gamma correction [22; 23] is applied as the next step. It performs a nonlinear manipulation of pixel intensities to enhance the dynamic range of pixels. Details are further enhanced as underlying information is highlighted to improve visibility and quality [24; 25]. It steers the average brightness of the image to the desired direction in a much better way than histogram equalization [15]. The homogeneity of the co-occurrence matrix is minimized to optimize the local gamma values of the original pixels.

CLAHE, when followed by gamma correction, has proven to give good results on other kinds of medical images as well [26]. In our experiments, we found that the best recognition results were obtained when the value of the gamma parameter was kept to 4. Gamma correction is shown in Figure 7.

Figure 7. Gamma correction

- **Step 4: Image sharpening**

The gamma-corrected image was then sharpened using the un-sharp masking method. Sharpness is the contrast between different colors, and image sharpening is a means of improving this contrast [27]. It can bring out the image information that is otherwise missed by the naked eye [28]. In the un-sharp masking technique, an un-sharp [also called blurred] version of the image is subtracted from itself to bring out the edges to create an edge mask. The edge contrast of the original image is improved by using this edge mask.

An un-sharp mask is produced by spatially filtering the image with a Gaussian filter. This filter can be treated as a convolution operation of an image with a kernel mask. The standard deviation of the Gaussian low-pass filter was kept to 6 in our experiments. This value controls the region's size around the edge pixels influenced by sharpening. A significant value sharpens wider regions around the edges, whereas a small value sharpens narrower regions. The strength of the sharpening effect was kept to be 5. The sharpened image is shown in Figure 8.

Figure 8. Sharpened image

- **Step 5: Multi filtering**

The sharpened image is then passed through a median filter followed by a Weiner filter. The superiority of this multi-filter approach is that it effectively removes random noise while preserving thin lines and edges of the original image [29] shown in Figure 9. Median filtering is a nonlinear process that reduces noise impulsive noise and salt-and-pepper noise. In a median filter, a window slides along the image, and the median intensity value of the pixels within the window becomes the output intensity of the pixel being processed [30] shown in Figure 10.

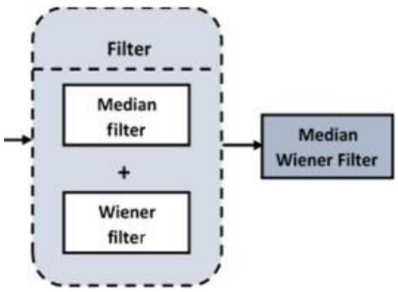

Figure 9. Multi-filter [29]

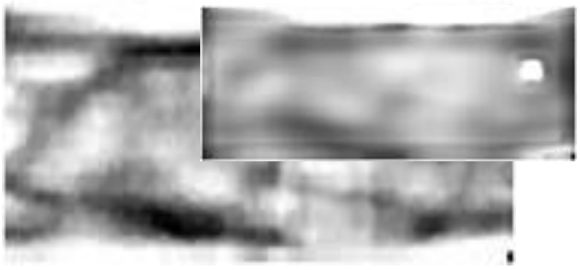

Figure 10. Weiner Filter

- **Step 6: Contrast Adjustment**

As the next step, the perceptual quality of the multi-filtered image was enhanced by adjusting contrast [31] shown in Figure 12.

Figure 11. Contrast adjusted image

4.2 Histogram of Gradients [HOG] Feature extraction

The next stage is preprocessing. The algorithm used was the Histogram of Gradients [HOG]. This technique is sensitive to the shift and rotation of images [32], thus making it a suitable low-level feature descriptor.

The image is first divided into small portions called 'cells.' For each cell, HOG counts the occurrences of edge orientations in the neighboring local region [34]. For 'unsigned gradients,' histogram channels are spread evenly over 0- 180 and for 'signed' over 0- 360. The histogram counts are then scaled to compensate for the effect of illumination. This is accomplished by adding the measure of energy for each local histogram over the larger connected regions. In this way, all the block-cells are normalized. These combined histograms make up the HOG feature vector for a given image.

Scale-space extrema detection is used to extract features from only the key- points [34]. This makes the algorithm immune to scaling and rotation effects in case of rotation normalization. Next, an Orientation assignment is used to find the dominant gradient orientation. As the final step, a feature vector is compiled by considering all orientation counts in the computed dominant direction of gradients. Gradient directions for a sample image patch are depicted in Figure 13.

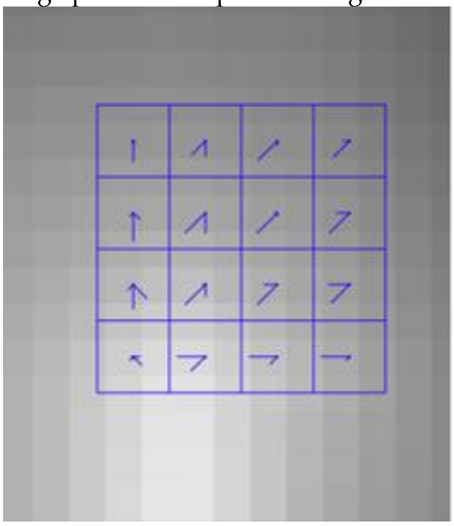

Figure 12. Example HOG descriptors, patch size=8 8. Each cell of the patch shows the orientation of the gradients.

So a HOG feature descriptor contains information about histograms of slopes of object edges in an image. It will yield better results for a preprocessed image with enhanced texture and sharpened edges than for an unprocessed image. This is why HOG feature extraction was preceded by the six-level preprocessing described above. If k is the total number of overlapping blocks in an image, then the HOG feature descriptor can be represented by the equation [35]:

V HOG= [F 1, F 2, F 3, F k]

Here F is the normalized block vector of ith block. Bins make up a cell, and cells make up a block. If C represents the total number of cells in a block and B represents the total number of bins in a cell, then:

F i= [h [1,i],h [2,i],h [3,i],h [C*B,i]]

The article in [35] illustrated this process in Figure 14 using a sample image of size 32x32.

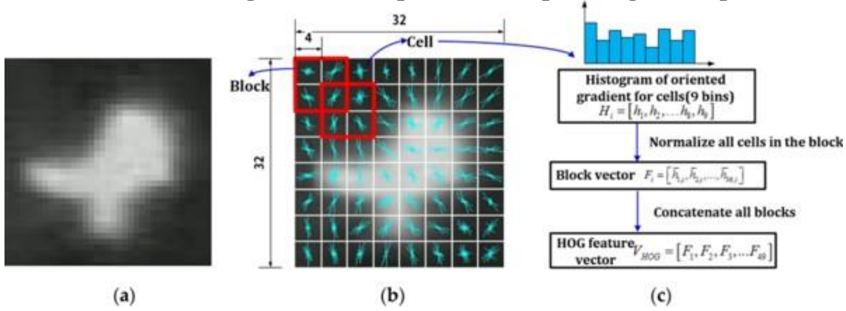

Figure 13. [a] Sample image [b] Visualization of gradients. Each block [red outline] contains four cells [white outline] [c] Process of generating HOG feature vector [35]

In the proposed algorithm, when HOG is applied to a preprocessed image, we get the results shown below:

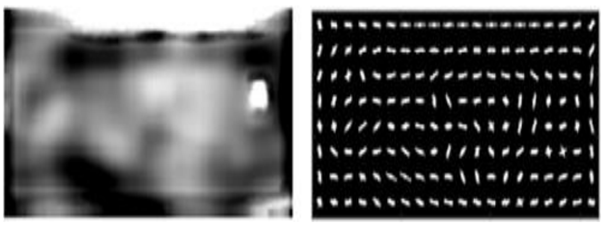

Figure 14: Visualization of extracted HOG features

5. Results

The main aim of this study is to provide an image preprocessing algorithm that is strong enough to give good authentication results even on low-level features like HOG and the simplest of all classifiers, K nearest Neighbors [KNN]. It is a computationally simple algorithm that requires no explicit training stage [36]. It is intuitively simple and gives good performance for practical purposes [37]. We used one nearest neighbor in our experiments.

SDUMLA database was used to test our algorithm because it has been regarded as the toughest of all, as described in the database section of this paper. A stratified 6-fold cross-validation was done to evaluate the results. The experimental computations were performed on a core i7 processor with 8 GB RAM and a 64-bit version of Windows 10. The parallel computing feature of MATLAB was used to run the experiments, which significantly reduced the execution time. The average time to preprocess an image was 0.016190 seconds.

Two basic types of classification problems are dealt with in this research:

1. The first one, simply the person's ID is recognized. This is our actual problem and is the one addressed in all major research that has taken place in this domain. There are a total of 106 classes in this setting.
2. In the second classification problem analyzed, not only the person ID was recognized but also the hand and finger type, that is:

(a) Person ID
(b) Hand type [right/ left]
(c) Finger type [middle/ ring/ index and their combination].

It can also be regarded as a person classification problem where each finger of each hand is assumed to have come from a unique individual. There are a total of 106*2*3 = 636 classes in this experimental setting. Experiments showed a mere difference of only + -1% in the results of these two experimental scenarios. This goes on to stress the robustness of the proposed technique. It gives highly accurate results even if the number of classes is in- creased from 106 to 636. This is of paramount importance as far as real-world applications go since they need to be designed for many users. Extensive and in-depth experimentation was conducted for the two experimental setups described above. Results are summarized in figure 15.

Hand	Finger	Identification Accuracy (%) on a person type	Identification Accuracy (%) on person type + finger type + hand type	Person Verification EER %	EER on person type + finger type + hand type
Right	Middle	99.0566	99.0566	0.0089	0.0089
	Ring	98.8994	98.8994	0.0104	0.0104
	Index	98.4277	98.2704	0.0148	0.0148
	Middle + Ring + Index	97.6939	97.1174	0.0272	0.0087
Left	Middle	99.0566	99.0566	0.0089	0.0089
	Ring	97.6415	97.6415	0.0222	0.0222
	Index	97.9560	97.9560	0.0193	0.0193
	Middle + Ring + Index	97.1698	96.8029	0.0282	0.0112
Right + Left	Middle	97.5629	98.3491	0.0230	0.0082
	Ring	97.5629	97.4057	0.0267	0.0124
	Index	96.9340	97.0126	0.0230	0.0133
	Middle + Ring + Index	96.9602	96.2002	0.0304	0.0073

Figure 15. Experimental Results on Stratified 6 Fold Cross-Validation

From Table 2, it can be observed that the middle finger of each hand is the most distinguishable in terms of identification and verification. If only the middle finger of either hand is used for enrollment and testing, that gives a near-perfect system with more than 99% accuracy and negligible EER. These results are in agreement with [38] concerning the fact that if multiple fingers are used to identify a person, the system performance may be quite different than when users use different fingers to do recognition.

To present a fair comparison, we have compared our results against some latest works that performed stratified 6- fold validation on the same SDUMLA database shown in figure 16. Some authors have evaluated their methods on 3-fold and 2-fold cross-validations on the SDUMLA database. To compare our results against theirs, we performed two other rounds of experimentation with the same k-validation to present unbiased comparisons.

From the above table, we can see that the proposed method surpasses even Deep learning [34; 14] in terms of EER results. It gives not only better results but also computationally faster. The CNN method of [14] uses preprocessing, the CNN method of [40] did not use the preprocessing stage, and the method for CNN training is computationally heavy. Our proposed method, on the other hand, uses lightweight preprocessing, and the training stage imposes a minimal strain on the system in figure 17.

Value of K in stratified K fold cross-validation	Reference	Method	Accuracy (%)	EER (%)
K= 6	[8]	Tri-Branch Vein Structure Analysis	-	3.46
	[12]	Anatomy Structure Analysis	-	1.39
	[10]	Deformation information	94	0.0268
	Proposed method	**AHistGSFC**	**97.6939**	**0.0073**
K= 3	[39]	Stable and Discriminative Super pixels	92.71	0.0194
	[40]	Convolutional Neural Networks (CNN) without preprocessing of images	-	0.8
	Proposed	**AHistGSFC**	**97.4843**	**0.03411**
K= 2	[14]	Convolutional Neural Networks (CNN) with preprocessing of images	-	3.0653
	proposed	**AHistGSFC**	**95.2830**	**0.0608**

Figure 16. Comparison of Latest Research Works that Treated each Finger as a Separate Class (636 classes)

Stage	Technique	Computational time (seconds)
Preprocessing	1. Detection of finger	0.026826
	2. CLAHE	0.016184
	3. Gamma correction	0.015100
	4. Image sharpening	0.017738
	5. Multi filtering	0.027670
	6. Contrast adjustment	0.015656
Feature extraction	HOG	0.016561
Training	KNN	0 (KNN requires no training)
Matching/testing	KNN	0.0018
Total time taken		0.1375

Figure 17. Average Computational Time for a Single Image

6. Conclusion

This paper proposes a computationally simple yet highly efficient method to preprocess finger vein images. The preprocessing approach is built upon CLAHE. It successfully creates a unique preprocessed image with nearly the same structure for each registered class. The enhancement algorithm is robust enough to give similar-looking images for each class. We validated our algorithm on the SDUMLA database, which has images of poor quality, and same-finger samples also suffer from substantial shift and rotation. HOG feature extraction is sensitive to the shift and rotation of images [32]. This leads to HOG being a suitable low-level feature descriptor for the purpose. K- Nearest neighbor was used for evaluation because we wanted to check the algorithm's efficiency on the simplest classifiers. It was observed that the middle fingers are the most distinguishable for any hand with SDUMLA data. The most striking feature of this algorithm is that this method involves no complex computation. A set of basic preprocessing steps has been formulated that, although all lightweight in terms of computational complexity, gives excellent results without any high-level feature extraction and deep learning. Future work can use some high-level techniques of feature extraction and classification with the described preprocessing algorithm to improve accuracy given the suggested preprocessing algorithm of AHistGSFC. This algorithm may also be tested for other types of vein images like wrist, palm, and dorsal hand veins in future extensions of this work.

Acknowledgments: The Researchers would like to thank the Deanship of Graduate Studies and Scientific Research at Qassim University for financial support [QU-APC-2024-9/1].

Conflicts of Interest: "The authors declare no conflict of interest".

References

1. S. N. M. S. L. S. Z. Marcel, Handbook of Biometric Anti-Spoofing; Trusted Biometrics under Spoofing Attacks, Advances in Computer Vision and Pattern Recognition, 2014.
2. D. Menotti, G. Chiachia, A. Pinto, W. R. Schwartz, H. Pedrini, A. X. Falc˜ao, and A. Rocha,"Deep Representations for Iris, Face, and Finger- print Spoofing Detection," IEEE Transactions on Information Forensics and Security, vol. 10, no. 4, 2015.
3. E. P. E. M. P. C. Rig Das, "Convolutional Neural Network for Finger-Vein- based Biometric Identification," IEEE TRANSACTION ON INFORMATION FORENSICS AND SECURITY, 2018.
4. Y. Z. Ajay Kumar,"Human Identification using Finger Images," IEEE Transactions on Image Processing, 2012.
5. A. Kumar and K. V. Prathyusha,"Personal Authentication Using Hand Vein Triangulation and Knuckle Shape," IEEE Transactions on Image Pro- cessing, 2009.
6. Y. Zhou and A. Kumar,"Human Identification Using Palm-Vein Images," IEEE Transactions on Information Forensics and Security, 2011.
7. J. Yang, Y. Shi, and G. Jia,"Finger-vein image matching based on adaptive curve transformation," Elsevier, 2017.
8. L. Yang, G. Yang, X. Xi, X. Meng, C. Zhang, and Y. Yin, "Tri-Branch Vein Structure Assisted Finger Vein Recognition," IEEE Access, vol. 5, 2017.
9. G. Y. X. X. Lu Yang,"Finger Vein Recognition with Anatomy Structure Analysis," IEEE Transactions on Circuits and Systems for Video Technology, 2018.
10. X. M. X. Y. Yin, "Science China Information Sciences," Finger vein recognition based on deformation information, 2018.
11. Y. Yin and L. L. X. Sun,"SDUMLA-HMT: A Multimodal Biometric Database," in Chinese Conference on Biometric Recognition, 2011.
12. [G. Y. Y. Y. a. X. X. Lu Yang,"Finger Vein Recognition with Anatomy Structure Analysis," IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS FOR VIDEO TECHNOLOGY, 2017.
13. Y. Yang, G. Yang, and S. Wang,"Finger Vein Recognition Based on Multi- instance," International Journal of Digital Content Technology and its Applications, 2012.
14. J. M. S. a. K. R. P. Wan Kim, "Multimodal Biometric Recognition Based on Convolutional Neural Network by the Fusion of Finger-Vein and Finger Shape Using Near-Infrared [NIR] Camera Sensor," Sensors MDPI, 2018
15. H. C. L. K. R. P. Eui Chul Lee,"Finger Vein Recognition Using Minutia- Based Alignment and Local Binary Pattern-Based Feature Extraction."
16. K.-R. Park, Y.-K. Jang and B.-J. Kang,"A study on touchless finger vein recognition robust to the alignment and rotation of finger," The KIPS Transactions: PartB, 2008.
17. S. A and S. k. k.s,"Finger Vein Biometric Approach for Personal Identification Using IRT Feature and Gabor Filter Implementation," International Research Journal of Engineering and Technology [IRJET], 2016.
18. G. Yadav, S. Maheshwari, and A. Agarwal,"Contrast limited adaptive histogram equalization based enhancement for the real-time video system," International Conference on Advances in Computing, Communications and Informatics [ICACCI], 2014.
19. B. B. Singh and S. Patel,"Efficient Medical Image Enhancement using CLAHE Enhancement and Wavelet Fusion," International Journal of Computer Applications, 2017.
20. L. Li, Y. Si, and Z. Jia, "Medical Image Enhancement Based on CLAHE and Unsharp Masking in NSCT Domain," Journal of Medical Imaging and Health Informatics, 2018.
21. A. D. M. D. S. M. Randa Boukhris Trabelsi,"A New Multimodal Biometric System Based on Finger Vein and Hand Vein Recognition," International Journal of Engineering and Technology [IJET], 2013.
22. S. A. Amiri and H. Hassanpour, "A Preprocessing Approach For Image Analysis Using Gamma Correction," International Journal of Computer Applications, 2012.
23. K. Somasundaram and P. Kalavathi, "MEDICAL IMAGE CONTRAST ENHANCEMENT BASED ON GAMMA CORRECTION," International Journal of Knowledge Management e-Learning, 2011.
24. C. W.. S. L.. M. R.. T. R.. N. K.. H. S. Y.-H. Y. T. W. G. Jianga, "Image contrast enhancement with brightness preservation using an optimal gamma correction and weighted sum approach," Journal of Modern Op- tics, 2015.
25. H. Hassanpour, N. Samadiani, and S. Salehi, "Using morphological trans- forms to enhance the contrast of medical images," The Egyptian Journal of Radiology and Nuclear Medicine, 2015.

26. G. S. A. R. A. A.-D. T. S. a. a. M. A.-R. Zohair Al-Ameen, ”An innovative technique for contrast enhancement of computed tomography images using normalized gamma-corrected contrast-limited adaptive histogram equalization,” Journal on Advances in Signal Processing, 2015.
27. S. Kansal, S. Purwar, and R. K. Tripathi, ”Image contrast enhancement using unsharp masking and histogram equalization,” Multimedia Tools and Applications, 2018.
28. R. J. C. M. J. H. P. F. T. D. M. Spring K, ”Unsharp mask filtering. In: Interactive tutorials-optical microscopy primer,” 2016.
29. K. Akintoye, M. S. M. Rahim and A. H. Abdullah, ”ENHANCEMENT OF FINGER VEIN IMAGE USING MULTIFILTERING ALGORITHM,” ARPN Journal of Engineering and Applied Sciences, 2018.
30. M. Nagu and N. Shanker, ”Image De-Noising By Using Median Filter and Weiner Filter,” International Journal of Innovative Research in Computer and Communication Engineering, 2014.
31. K. Gu, G. Zhai, W. Lin, and M. Liu, ”The Analysis of Image Contrast: From Quality Assessment to Automatic Enhancement,” IEEE Transactions on Cybernetics, 2016.
32. A. Vashaee, R. Jafari, D. Ziou and M. Mehdi Rashidi, ”Rotation Invariant HOG for Object Localization in Web Images,” Signal Processing, 2016.
33. C. Hang, F. Hu, A. E. Hassanien and K. Xiao, ”Texture-based rotation- invariant Histograms of Oriented Gradients,” in 11th International Com- puter Engineering Conference [ICENCO], 2016.
34. O. Deniz, G. Bueno, J. Salido and F. D. l. Torre, ”Face recognition using Histograms of Oriented Gradients,” Pattern Recognition Letters, 2011.
35. Y. Ma, X. Wu, G. Yu, Y. Xu and Y. Wang, ”Pedestrian Detection and Tracking from Low-Resolution Unmanned Aerial Vehicle Thermal Im- agery,” Sensors, 2016.
36. S. Zhang, X. Li, M. Zong, X. Zhu, and R. Wang, ”Efficient kNN Classification With Different Numbers of Nearest Neighbors,” EEE Transactions on Neural Networks and Learning Systems, 2018.
37. M. Kibanov, M. Becker, J. Mueller, and Martin, ”Adaptive kNN using expected accuracy for classification of geo-spatial data,” Proceedings of the 33rd Annual ACM Symposium on Applied Computing, 2018.
38. Y. Y. L. N. S. L. W. L. He Zheng, ”Which Finger is the Best for Finger Vein Recognition?,” IEEE 8th International Conference on Biometrics Theory, Applications and Systems [BTAS], 2016.
39. G. Y. Y. Y. L. Y. a. K. W. Lizhen Zhou, ”Finger Vein Recognition Based on Stable and Discriminative Superpixels,” International Journal of Pattern Recognition and Artificial Intelligence, 2016.
40. W. Liu, W. Li, L. Sun, L. Zhang and P. Chen, ”Finger vein recognition based on deep learning,” in 2017 12th IEEE Conference on Industrial Elec- tronics and Applications [ICIEA], 2017.
41. H. T. Van and T. T. Thai, ”ROBUST FINGER VEIN IDENTIFICATION BASE ON DISCRIMINANT ORIENTATION FEATURE,” in Seventh International Conference on Knowledge and Systems Engineering, 2015.
42. G. Seshikala, U. Kulkarni and M. Giriprasad, ”Biometric Parameters Palm Print Recognition,” International Journal of Computer Applications [0975– 8887], 2012.
43. Y. Lu, S. J. Xie, S. Yoon, and D. S. Park, ”Finger Vein Identification Using Polydirectional Local Line Binary Pattern,” International Conference on ICT Convergence [ICTC], 2013.
44. J. H. T. K. Thian Song Ong and A. B. J. Teoh, ”Multi-instance Finger Vein Recognition Using Minutiae Matching,” in 6th International Congress on Image and Signal Processing [CISP 2013], 2013.
45. Sero, D., Garachon, I., Hermens, E., Liere, R. V., Batenburg, K. J. [2021]. The study of three-dimensional fingerprint recognition in cultural heritage: trends and challenges. Journal on Computing and Cultural Heritage [JOCCH], 14[4], 1-20.
46. Abdul-Al, Mohamed, et al.”Performance of Multimodal Biometric Systems Using Face and Fingerprints [Short Survey].” 2021 IEEE 26th International Workshop on Computer Aided Modeling and Design of Communication Links and Networks [CAMAD]. IEEE, 2021.
47. Boyraz, O. F., Guleryuz, E., Akgul, A., Yildiz, M. Z., Kiran, H. E., Ahmad, J. [2022]. A novel security and authentication method for infrared medical image with discrete time chaotic systems. Optik, 267, 169717.
48. Moradi, M., Moradkhani, M., Tavakoli, M. B. [2022]. A Real-Time Bio- metric Encryption Scheme Based on Fuzzy Logic for IoT. Journal of Sensors, 2022.
49. Guven, Gokhan, Umit Guz, and Hakan Gu¨rkan. ”A novel biometric identification system based on fingertip electrocardiogram and speech signals.” Digital Signal Processing 121 [2022]: 103306.

50. Shand, S. [2022, April]. Importance of Human Gait Analysis in Biometric Recognition using Machine Learning: A Review. In 2022 6th International Conference on Trends in Electronics and Informatics [ICOEI] [pp. 1431- 1436]. IEEE.
51. Siroya, Navani, and Manju Mandot. ”Biometric Identification Using Deep Learning for Advance Cloud Security.” Deep Learning Approaches to Cloud Security [2022]: 1-14.
52. Aung, H. M. L., Pluempitiwiriyawej, C., Hamamoto, K., Wangsiripitak, S. [2022]. Multimodal Biometrics Recognition Using a Deep Convolutional Neural Network with Transfer Learning in Surveillance Videos. Computation, 10[7], 127.