© Universiti Tun Hussein Onn Malaysia Publisher's Office

IJIE

Journal homepage: http://penerbit.uthm.edu.my/ojs/index.php/ijie
ISSN : 2229-838X e-ISSN : 2600-7916

The International Journal of Integrated Engineering

Finger Vein Recognition Using Principle Component Analysis and Adaptive *k*-Nearest Centroid Neighbour Classifier

Ng Tze Han[1], Nordiana Mukahar[2*], Bakhtiar Affendi Rosdi[1]

[1]Intelligent Biometric Group,
School of Electrical and Electronic Engineering,
Universiti Sains Malaysia, Nibong Tebal, Seberang Perai Selatan, 14300, Pulau Pinang, MALAYSIA

[2]Faculty of Electrical Engineering,
Universiti Teknologi MARA, Shah Alam, 40450, Selangor, MALAYSIA

*Corresponding Author

DOI: https://doi.org/10.30880/ijie.2021.13.01.016
Received 7 July 2020; Accepted 26 November 2020; Available online 30 January 2021

Abstract: The *k*-nearest centroid neighbour kNCN classifier is one of the non-parametric classifiers which provide a powerful decision based on the geometrical surrounding neighbourhood. Essentially, the main challenge in the kNCN is due to slow classification time that utilizing all training samples to find each nearest centroid neighbour. In this work, an adaptive *k*-nearest centroid neighbour (akNCN) is proposed as an improvement to the kNCN classifier. Two new rules are introduced to adaptively select the neighbourhood size of the test sample. The neighbourhood size for the test sample is changed through the following ways: 1) The neighbourhood size, *k* will be adapted to *j* if the centroid distance of *j*-th nearest centroid neighbor is greater than the predefined boundary. 2) There is no need to look for further nearest centroid neighbours if the maximum number of samples of the same class is found among *j*-th nearest centroid neighbour. Thus, the size of neighbourhood is adaptively changed to *j*. Experimental results on the Finger Vein USM (FV-USM) image database demonstrate the promising results in which the classification time of the akNCN classifier is significantly reduced to 51.56% in comparison to the closest competitors, kNCN and limited-kNCN. It also outperforms its competitors by achieving the best reduction ratio of 12.92% while maintaining the classification accuracy.

Keywords: Classification, *k*-nearest centroid neighbor classifier, finger vein recognition

1. Introduction

Automatic personal recognition based on human vein patterns such as palm vein [1, 2], finger vein [3, 4] and dorsal hand vein [5] has attracted much attention in biometrics community among researchers. In terms of practical application, finger vein patterns possess several advantages in comparison to other human vein traits because adequate vein information can be obtained by using multiple fingers [4]. Besides, the size of the image acquisition device of the finger vein is relatively small and convenient for biometric applications [3].

A finger-based biometric system that utilizes physiological and behavioral features of a human finger is widely used in many applications for years in comparison to other human traits such as the face, iris, *etc* due to their high user acceptance [6]. The intrinsic feature of human finger such as finger vein has several desirable properties as a biometric identifier in comparison to other biometric characteristics. Finger vein holds the following merits [3]: (a) Contactless and user friendly: the contactless capture of finger vein image also ensures both convenience and cleanliness, thus users may find it less intrusive compared to iris scanning systems, i.e., being more user-friendly (b) Live-body identification: the finger-vein patterns can be identified only on fingers of living bodies with blood flow. (c) Immunity to counterfeit: finger-

*Corresponding author: nordi741@uitm.edu.my
2021 UTHM Publisher. All rights reserved.

vein patterns are internal features embedded in the fingers, and it makes vein pattern duplication impossible in practice. Finger vein recognition is composed of four processes: 1) image capturing, 2) image preprocessing, 3) feature extraction and, 4) classification based on the extracted features. A representative of a meaningful feature set is extracted using various techniques and becomes the input to the classifier. Robust classifier does some strategies to determine the class of unknown identity into one of the pre-specified classes accurately [7]. In the settings of classification, two types of classifiers are commonly used in finger vein recognition: parametric-based and non-parametric based. The information of the underlying joint models of the data is necessary for parametric classifiers and certain parameters need to be estimated. Contrary to the parametric classifiers, the non-parametric classifiers are explicitly independent of the data's underlying distributions and do not make any assumption about the shape of the classes [1]. Thus, the non-parametric classifiers have several advantages such as easy implementation, competitive performance and free from any parameter estimations. Besides, several recent researches related to the enhancement performance of non-parametric classifiers [8–13] show the great potential non-parametric classifiers and there is abundant of room for future research.

The k-nearest neighbour (kNN) has been well known and widely used in pattern classification because it is easy to implement in practice. It is independent of any parameter prediction at the learning stage and works well with a small set of samples. If a new sample is added to the original training set, the retraining is not necessary for the kNN [14]. However, the kNN suffers from three main drawbacks which are [15]: 1) the necessity of high storage requirements, 2) the low efficiency obtained during the computation of the decision rule, 3) it presents low tolerance to noise due to its assumption that all data are relevant. To solve the deficiency in the kNN, the kNCN was proposed by [16] which is based on the nearest centroid neighbourhood concept. The nearest centroid neighbourhood was first introduced in [17] to improve the nearest neighbourhood definition in the kNN classifier. The kNCN defines the neighbor based on two properties: 1) it must close to the input sample, and 2) it has symmetrical distribution. It has been proven by several works in the literature that the kNCN outperforms the kNN classifier [18–20].

Motivated by the kNCN classifier, an extension of the kNCN is proposed in this work called adaptive k-nearest centroid neighbor (akNCN). Despite good classification accuracy, the time spent by the kNCN to classify the test sample is quite slow due to its requirement to examine whole training samples to find nearest centroid neighbours. Therefore, the proposed akNCN classifier proposes a strategy to reduce the classification time in the kNCN. The idea is to stop the nearest centroid neighbour search iteration whenever it is clear that the centroid distance of the j-th NCN is too far from the input sample. The proposed adaptive k-nearest centroid neighbour (akNCN) classifier makes three key contributions to the current state:

- The two rules introduced to allow the akNCN classifier to adaptively select the size of the neighbourhood for different test samples. Contrary to the kNCN classifier, in which the neighborhood size is not adaptively changed for an input sample and has a fixed value of neighbourhood size, k.
- The classification time can be reduced by employing the two rules to adaptively select the neighbourhood size of the input sample. The neighbourhood size for each input sample is varied since it is adaptively adjusted through two rules introduced in the akNCN classifier.
- The empirical comparison is made between the akNCN classifier and benchmark classifiers (kNCN and limited-kNCN) using publicly available finger vein dataset [21].

The remainder of this paper is organized as follows. Section 2 presents the previous works on the finger-vein recognition and the related kNCN-based classifiers. Section 3 explains the development of finger-vein recognition. Section 4 presents the proposed adaptive k-nearest centroid neighbour classifier. Section 5 provides the experimental results from the proposed classifier and competing classifiers on the finger vein image database. The analysis and discussion on the performance of classifiers are reported in Section 5. Section 6 remarks the key conclusion from this paper.

2. Related Work

In this section, a concise explanation of the previous works done on finger vein recognition and the related classification schemes are presented.

2.1 Finger Vein Recognition

A considerable number of works on finger vein biometric recognition have been reported in the literature utilizing different strategies of feature extraction techniques and classification approaches. In this study, the main concern is on the classification strategies of finger vein recognition rather than the feature extraction techniques. The classification strategies proposed in the literature on finger vein recognition can be divided into two types: parametric and non-parametric.

Traditional neural networks are parametric types that consist of various kinds of architecture to be applied for data classification, pattern recognition, etc. Finger vein recognition based on the neural network with radial basis function was proposed by [22, 23] on a small set of finger vein database. Another strategy of using neural networks on the finger vein database proposed in [24, 25] where they applied different models of a neural network such as back propagation and convolutional for classification. For each work, the vein pattern features are extracted with Radon transform [22], PCA [23, 24], LDA [26] and convolutional neural network [25] prior classification step.

Several works used non-parametric based classifiers such as SVM and kNN as an alternative to classify the extracted features of vein [26–33]. The SVM is well known for its ability to handle both linear and nonlinear cases of data distribution [34]. It has been proven in [26, 30, 32, 35] that the outstanding recognition accuracy could be achieved by using the SVM and selected kernel function. However, the successful application of SVM on finger vein recognition relies on the adjustment of the kernel parameters and thus, it requires a training procedure to determine those parameters. The implementation of the kNN is easier than the SVM because it is independent of any parameter prediction at the training set. Works in [27, 28, 36] explore the potential and usability of the kNN classifier which performs the classification on the extracted vein features based on a group of nearest neighbours. The study in [3, 37] implemented Hamming distance and Euclidean distance to calculate matching scores, respectively. The utilization of the non-parametric classifier, for example, kNN in the biometric is very few and has not been explored much. It is noticeable that the recognition performance obtained using non-parametric classifier is comparable with the parametric types and has shown outstanding recognition performance. Based on the outstanding recognition performance shown by previous works and its ease implementation, a novel parametric-based classifier rooted from the kNN is proposed in this work.

2.2 Classification Schemes

The *k*-nearest centroid neighbour (kNCN) is one of the non-parametric classifiers based on the centroid distance. It defines that the nearest neighbours of the test sample must follow these two criteria [16]: 1) it must close enough to the test sample, and 2) the distribution of nearest neighbour must symmetrical around the test sample. However, it is difficult to find neighbours in the feature space that fulfill these two properties equally. In the area of nearest centroid neighbourhood region, the location of selected some nearest centroid neighbours might too far from the test sample but there can be some training samples located closer to the test sample than the farthest distance centroid neighbour. This indicates the kNCN classifier is sensitive to atypical samples.

Despite good classification performance offers by the kNCN classifier, it suffers from the slow classification time due to its requirement to check all training samples in the nearest centroid neighbour searching process [38]. To alleviate the disadvantage of the kNCN classifier, the limited-kNCN proposed in [18] is introduced to reduce the time consumes to look for nearest centroid neighbours by reducing the set of the training samples. There are two variants of a technique devised in [18]: limited-kNCN.v1 and limited-kNCN.v2. According to the rules, it needs to find a fraction of training samples and this value of the fraction is determined during the learning phase. The first variant approach, limited-kNCN.v1 set the *m* value as the maximum rank (based on the nearest distance) among the *k*-nearest centroid neighbours of all training samples with *N* is the total count of training samples. Thus, the desired fraction is calculated as

$$f_{v1} = m/(N-1) \tag{1}$$

However, the *m* value that based on the maximum rank is susceptible to the noisy samples because one of the *k*-nearest centroid neighbours of training samples may have the largest rank according to the Euclidean distance or nearest neighbour's rank. The second variant is known as limited-kNCN.v2 proposed m_{robust} to reduce the influence of atypical or noisy samples. The m_{robus} is defined as the optimum rank values such that 95% of training samples set their farthest k -nearest centroid neighbours not exceeding m_{robus} value. The fraction is computed as in Eq. 2.

$$f_{v2} = m_{robust}/(N-1) \tag{2}$$

The effort to calculate the exact amount of fraction to use is done during the learning phase with the kNCN rule. However, the use of training samples to obtain the fraction is suspicious as the training sample might not have the same distribution as the test sample. Nevertheless, the estimation of the fixed percentage of training samples does not count the data sparseness and variance of the distribution. It should be done appropriately by considering the characteristic of the data set.

3. The Development of the Finger Vein Recognition

This paper proposes a finger vein recognition which utilizing meaningful features possess by finger vein pattern for identification. The developed finger vein recognition as illustrated in Fig.1 consists of three main stages including image acquisition, preprocessing (the segmentation of the region of interest and feature extraction) and classification.

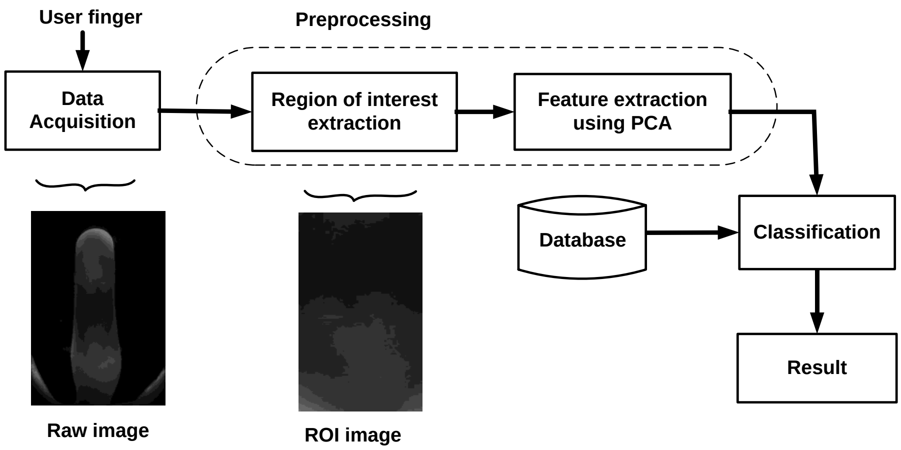

Fig.1 - The design of the finger vein recognition.

The data acquisition process involves capturing raw vein images of the user's hand using image acquisition devised developed by [21] and the brief explanation detail on how it is working also can be found in [21]. Subsequently, it is followed by the preprocessing stage in which the region of interest (ROI) is segmented with the size 300×100 and 256 grey levels. The segmented ROI image is rescaled into a smaller size with a resize ratio of 0.1 to eradicate the pixel noise. The Principle Component Analysis (PCA) is a traditional subspace-based algorithm for feature extraction and it has been used extensively in the context of various traits of biometric recognition, for examples: speech emotion recognition [39], face recognition [40], and finger recognition [3]. It is observed in [41] that the increment number of principle components yields a corresponding improvement in the classification performance. It means that the employment of PCA give a positive impact on the success of classification performance.

The collection of finger vein images contains 5,904 ROI images taken from 123 individuals. The size of the captured image is 300×100 pixels and 256 grey levels. The collection of finger vein images is captured using the image acquisition device developed in [21] and 123 volunteered individuals took part to contribute the image fingers. Image of four different fingers (left index, left middle, right index and right middle) are captured for each subject. The process of capturing finger vein images involves two different sessions in which the images taken during the first session are considered as registration and the set of images of the second session is considered as a test image. The Finger Vein USM image database used in this work is available and can be freely downloaded at the website: http://drfendi.com. Finger vein images from FV-USM image database are selected to be used in the experiment since most of the pattern recognition established method had been evaluated by using this database [3, 42–46].

The main aim of this work is to effectively classify finger vein patterns by using a robust classifier and consequently, achieve outstanding recognition performance with reliable classification time. As an extension to the k -nearest centroid neighbour (kNCN) classifier, the adaptive k-nearest centroid neighbour (akNCN) classifier is devised to reduce the classification time while preserving the good classification accuracy.

4. The Proposed Adaptive *k*-Nearest Centroid Neighbour Classifier

The adaptive k-nearest centroid neighbor classifier is proposed to improve the classification time by adaptively adjusting the number of nearest centroid neighbors for each input sample, and at the same time maintains the good classification accuracy performance. Two rules are introduced in the akNCN classifier and these rules are devised such that the neighborhood size is dynamically selected by following the rules. Two rules of the proposed akNCN are described as below:

Rule 1: The akNCN classifier searching process reaches a stable searching state only if the j-th nearest centroid distance is more than a predefined boundary which is the product of multiplier, l_k and the distance of the first nearest centroid, $y_{ncn,1}$ to the test sample, x, $d(x, y_{ncn,1})$. The neighborhood size, k will be adapted to j value and this rule is presented in Eq .3.

$$d(x, y_i^c) > l_k \times d(x, y_{ncn,1}) \tag{3}$$

Where $d(x, y_j^c)$ is the nearest centroid nearest distance between the test sample, x and j-th centroid point, y_j^c. The multiplier, l_k is set to be greater or equal to 1 and the first nearest centroid distance is, $d(x, y_{ncn,1})$.

Rule 2: The akNCN classifier searching process reaches a stable searching state only if the total number of samples per class, N_i is found among the j nearest centroid neighbours and the total number of samples per class for a competing class is less than $N_i - 1$. In Rule 2, the neighborhood size, k will be adapted to j when these two conditions are met and it is defined as follows

$$(\forall NCN_j)(\exists T_{\omega_i})\wedge(\exists T'_{\omega_i}) \quad (4)$$

where $NCN_j(x) = \{ncn_x^y | ncn_x^y \in T\}_{y=1}^{j}$ denotes the j nearest centroid neighbors of the test sample, x. $T_{\omega_i} = \{y_y^i | y_y^i \in T\}_{y=1}^{N_i}$ denotes a class subset of T from class ω_i with the number of training samples, N_i and $T'_{\omega_i} = \{y_y^i | y_y^i \in NCN_j\}_{y=1}^{N'_i}$ denotes a class subset of T' from competing class, ω_i with the number of training samples, N'_i less than $N_i - 1$. The rationale behind the second rule is that when the first majority of nearest centroid neighbours is found from the same class, then it is unnecessary to look further nearest centroid neighbours. It is more likely that the new coming sample is belonging to the first class found with majority count.

5. Experiments

The proposed akNCN classifier is developed to improve the existing kNCN classifier as an improved version of kNCN as presented in the previous section. Performance evaluation is performed between the developed akNCN and benchmark classifiers (kNCN and limited-kNCN) on the Finger Vein USM (FV-USM) image database.

5.1 Determination the parameter of neighbourhood size,*k*

The first experiment intends to find the optimum value of neighbourhood size, k for the original kNCN classifier [16] by exploring different values of k from 3 to 21. The corresponding value of neighbourhood size, k which resulting the best classification accuracy will be considered as an optimum value of k. The best value of k which yields the highest classification accuracy will be used for the next experiment. Table 1 depicts the classification accuracy and classification time with different values of neighbourhood size, k. It can be observed that the classification accuracy of the kNCN is rapidly increased with the increment values of the k. The classification time of the kNCN classifier is $O(nk)$ [18] where n and k are representing the training samples and neighborhood size, k respectively. In this case, it is expected that with a fixed value of n and different values of k, classification time of the kNCN will gradually increase. This trend can be seen in Table 1 that the classification time is increasing when the neighbourhood size, k becomes large. Bold values in Table 1 denote that the highest classification accuracy is achieved at 85.33% when k =19 with the corresponding classification time 10,600 s. Thus, the neighbourhood size, $k = 19$ is used for the subsequent experiments for the empirical performance comparisons.

Table 1 - The classification accuracy(%) and classification time(s) of kNCN classifier with different values of *k*.

Size of neighbourhood, k	Classification accuracy (%)	Classification time (s)
3	79.27	1,163.10
5	81.54	2151.20
7	82.89	3,862.00
9	84.18	5,686.00
11	84.35	6,782.00
13	84.72	7,434.00
15	84.79	7,861.70
17	85.09	9,325.80
19	**85.33**	**10,600.00**
21	84.28	15,891.00

5.2 Empirical Performance Comparison Between the akNCN.v1 and the Original kNCN Classifiers

Different values of the multiplier, l_k are explored from 0.5 to 5 with step 0.5 to select the best value such that the akNCN.v1classifier obtains the best classification accuracy. The maximum value of nearest centroid neighbours is set to 19 which is equivalent to the optimum value of the size of the neighbourhood, k for the original kNCN obtained from the previous experiment. Nevertheless, the adapted value of the neighbourhood size of the akNCN.v1 might be varying for each test sample and it is possibly less than a fixed setting value of k in the original kNCN. According to the first rule introduced in Section 4, the neighbourhood size, k is equal to j if the j-th NCN centroid distance is found to be over a limit, and that limit is set as in Eq.3.

The performance comparison is made with a competing classifier, original kNCN [16] to observe the effect of introducing the first rule in the proposed akNCN classifier to the classification accuracy and classification time. The best accuracy with its classification time obtained from the previous experiment for the kNCN classifier is chosen for a fair comparison. Fig. 2 shows that at first when $l_k < 1$, the classification accuracy of the aKNCN.v1 is lower than the original kNCN. It then achieves the best classification accuracy at 85.64% when $l_k = 1$ with 0.3% higher than the original kNCN classifier. The classification accuracy of the akNCN.v1 nearly keeps stable with the original kNCN with an increasing value of multiplier value, l_k. A significant reduction in classification time can be observed in Fig. 3 resulting from the first rule introduced in the akNCN.v1 which indicates that the optimum value of neighbourhood size for each input sample can be adaptively selected by employing the first rule. The classification time which corresponds to the best accuracy at $l_k = 1$ provides approximately 41.21% reduction in comparison to the original kNCN. Thus, the optimum setting of the multiplier, l_k in this experiment is attained as 1 to be used in the subsequent experiment. The results show that the first rule introduced in the akNCN.v1 can elevate the classification accuracy while significantly reduces the classification time.

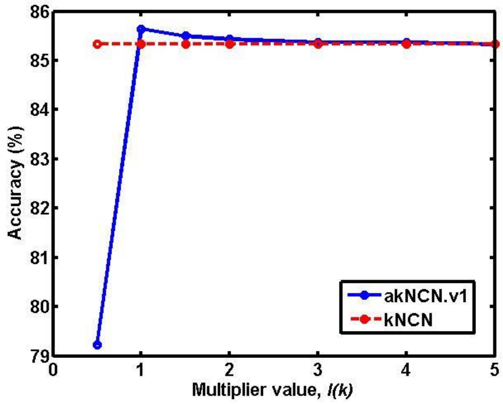

Fig. 2 - The classification accuracy of the akNCN.v1 classifier by changing the multiplier value, l_k

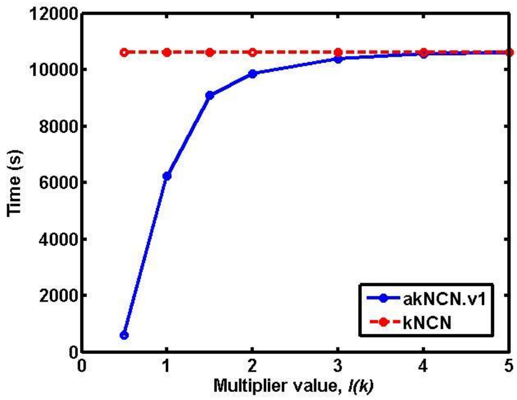

Fig. 3 - The classification time of akNCN.v1 classifier by changing the multiplier value, l_k

5.3 Empirical performance comparison between the akNCN.v2 and kNCN classifiers

The third experiment aims to evaluate the effect of employing both rules explained in Section 4 to the classification accuracy and classification time of the akNCN.v2 classifier. Two parameters involve in this experiment are set as follows: 1) the multiplier, l_k is set to 1 (based on the best value obtained from the previous experiment) and, 2) the neighborhood size, k is varying from 3 to 21 with steps 2. The size of neighborhood size, k determines the maximum number of nearest centroid neighbors in the akNCN.v2 rule. Since the akNCN.v2 is employing both rules, the size of its neighborhood is adapted to a certain value once it met the rules and it could be less than a fixed setting value of k .

The classification accuracy of the akNCN.v2 and the original kNCN classifiers is illustrated in Fig. 4 and it is noticeable that the akNCN.v2 classifier performs better than the kNCN with an increase of k. The classification differential between the aKNCN.v2 and kNCN is very significant especially when the value of k greater than 7. The original kNCN classifier uses a fixed value of neighborhood size, k to search nearest centroid neighbors and apply majority vote to decide the class. With a fixed parameter of k, there exist is a possibility that the kNCN classifier may select the samples located far away from the test sample and some of the training samples that have been considered as nearest centroid neighbors are from a wrong class which leads to misclassification. This explains the increment of 0.36% at $k = 19$ in the classification accuracy when using the akNCN.v2 classifier in comparison to the kNCN classifier. The optimum value of k which resulting in the best accuracy for the akNCN.v2 classifier is similar to the kNCN classifier with 0.3% higher than the original kNCN classifier.

Meanwhile, akNCN.v2 constantly achieves lower classification time than those of kNCN with the increment values of neighbourhood size, k as shown in Figure 5. At the optimum value of $k = 19$, the kNCN consumes 10,600 s to process 2,952 finger vein images, while in contrast to the proposed akNCN.v2 only need 5153.5 s to process the same number of images. It means that the implementation of both rules in the akNCN.v2 yields 51.38% of the reduction in classification time while improving the classification accuracy.

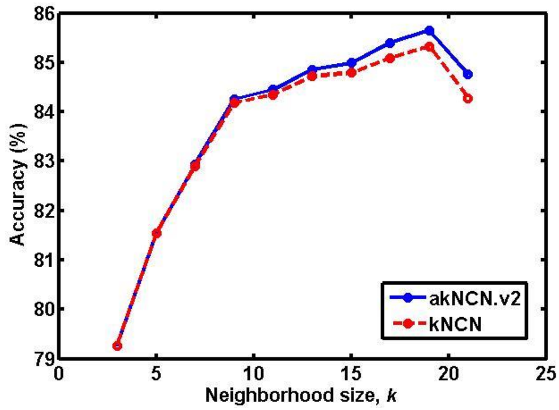

Fig. 4 - The classification accuracy of the akNCN.v2 classifier by changing the value of k

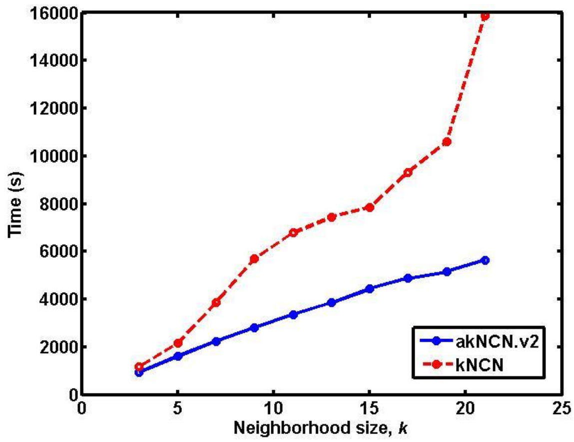

Fig. 5 - The classification time of akNCN.v2 classifier by changing the value of k

In the proposed akNCN.v2 classifier, the number of nearest centroid neighbours found for each test sample is varying. It is due to the implementation of both rules introduced in the proposed akNCN.v2 classifier that limiting the nearest centroid neighbour search under certain circumstances. Fig. 6 shows the number of test samples with its corresponding nearest centroid neighbours using akNCN.v2 classifier when the neighbourhood size, k is set to 19. In this experiment, a total 2,952 of test samples have a fixed number of nearest centroid neighbours ($k = 19$) if using the original kNCN classifier. In contrast to the proposed akNCN.v2 classifier, only 887 out of 2,952 test samples that have neighbourhood size, $k = 19$ and this corresponds to 30% of the total size of test samples. Among the remaining 70% of test samples that are affected by the rules introduced in akNCN.v2 classifier, 783 of test samples have neighbourhood size, $k = 2$. It is the first rule that effectively works on these 783 of test samples because the second rule requires a minimum of nearest centroid neighbours since the finger vein image database used has 6 samples per class. When the centroid distance of the 2nd nearest centroid neighbour is exceeding the boundary defined by the first rule, the number of nearest centroid neighbours found is limited to 2. Therefore, with only two nearest centroid neighbours to vote, the tiebreaker dictates that the membership of the input sample is decided based on the least accumulated Euclidean distance. Hence, the first nearest centroid neighbour is selected for the case $k = 2$. This observation is highly interesting as the akNCN.v2 is similar to a 1-nearest neighbor (1-NN) solution for these 783 cases.

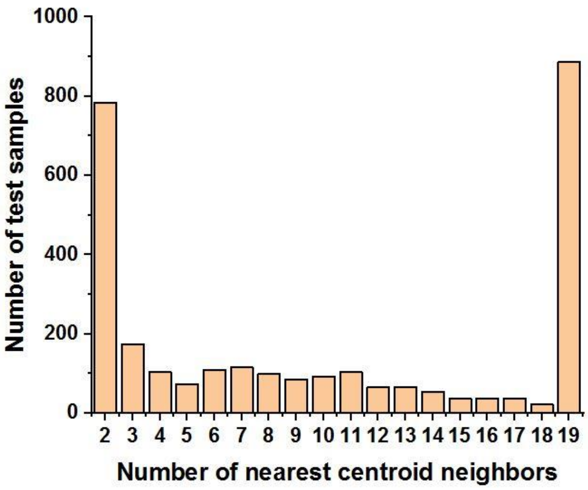

Fig. 6 - The number of test samples with its corresponding total number of nearest centroid neighbors using akNCN.v2 classifier

5.4 An Empirical Comparison of the akNCN, kNCN and Limited-kNCN Classifiers

Further empirical comparison is made in terms of the classification accuracy and classification time with other training set reduction technique for the kNCN classifier proposed in [18]. The limited-kNCN has two variants [18]: limited-kNCN.v1 and limited-kNCN.v2. The first version, limited-KNCN.v1 introduced the m value as the reduced number of training samples for nearest centroid neighbor search. It defines the m value is the maximum rank of nearest neighbors to find the kNCN during the learning phase. For the second approach, the limited-kNCN.v2 defined the m_{robust} value as the optimum rank of nearest neighbors belongs to 95% of training samples to find the kNCN during the learning phase. In this experiment, the values for m and m_{robust} are obtained in the learning phase as 2808 and 1527, respectively. Meanwhile, the neighborhood size, k is set to 19.

Table 2 shows the detailed classification accuracy and classification time achieved by the proposed akNCN classifiers and competing classifiers. It also includes for each measure the ranking in parentheses according to its performance. In this performance comparison, the best classification accuracy with its corresponding classification time is chosen from the previous experiments for the kNCN and akNCN classifiers. It is noticeable that both variants of the proposed akNCN classifiers achieve slightly greater accuracy than the original kNCN classifier with approximately 0.04% increment for each classifier. Furthermore, the proposed akNCN.v2 classifier is much faster than the akNCN.v1, limited-KNCN.v1, limited-kNCN.V2 and kNCN classifiers with the highest ranking. The result shows the effectiveness of the rules introduced in the aKNCN to adaptively select the neighborhood size for the input sample. On the other hand, the limited-KNCN.v1 classifier able to maintain the accuracy as the original kNCN while using only a fraction value, m of training samples during nearest centroid neighbors search. Even though the number of training samples has been reduced, the limited-kNCN.v1 suffers in classification time which takes 100.15% greater than the classification time required for the proposed akNCN.v2 classifier. On the other hand, the m_{robust} introduced in the limited-kNCN.v2 improves the classification time of the original kNCN tremendously with 44.17%. However, the accuracy of the limited-kNCN.v2 is slightly lower than the original kNCN with a 48.3% reduction in the original training set. The idea of using m_{robust} in the accuracy of the limited-kNCN.v2 is to make the classifier more robust against atypical data samples by taking the 95th percentile instead of the maximum value [18]. In this experiment, a large number of training samples that contain some useful information for a class decision might be removed, thus such reduction may degrade the classification accuracy of the limited-kNCN.v2.

Table 2 - Classification accuracy (%) and classification time (s) of the proposed akNCN classifiers and competing classifiers with the corresponding ranking in parentheses

Classifier	Classification accuracy (%)	Classification time (s)
Original kNCN [16] ($k = 19$)	85.33 (3.5)	10,600 (5)
akNCN.v1 ($l_k = 1$)	85.64 (1.5)	6,321.90 (3)
akNCN.v2 ($l_k = 1, k = 19$)	85.64 (1.5)	5,153.50 (1)
Limited-kNCN.v1 [18] ($m = 2808, k = 19$)	85.33 (3.5)	10,315 (4)
Limited-kNCN.v2 [18] ($m_{robust} = 1527, k = 19$)	85.09 (5)	5,918 (2)

6. Conclusion

In this study, an adaptive nearest centroid neighbour technique is proposed for the finger vein image classification problem. The proposed technique has proposed two rules: the use of the adaptive selection in determining effective size of neighbourhood and the reduction of classification time. Experiments conducted on the Finger Vein USM (FV-USM) database by using the aKNCN classifier show that the proposed aKNCN classifier achieves significant speed improvement in the classification time and also better classification accuracy than other existing classifiers, the original kNCN [16] and limited-kNCN classifiers [18]. In conclusion, the results demonstrate that the akNCN classifier can adaptively determine the effective number of neighbourhood size during classification task.

References

[1] Yan, X., Kang, W., Deng F., Wu, Q. (2015). Palm vein recognition based on multi-sampling and feature-level fusion. Neurocomputing, 151, 798–807

[2] Wu, K. S., Lee, J. C., Lo, T. M., Chang, K-C, Chang C-P. (2013). A secure palm vein recognition system. Journal of Systems and Software, 86, 2870–2876

[3] Qiu, S., Liu, Y., Zhou, Y., Huang, J., Nie, Y. (2016). Finger-vein recognition based on dual-sliding window localization and pseudo-elliptical transformer. Expert Systems with Applications, 64, 618–632

[4] Xi, X., Yang, L., Yin, Y. (2017). Learning Discriminative Binary Codes for Finger Vein Recognition. Pattern Recognition, 66, 26–33

[5] Gupta, P., (2015). Multi-modal fusion of palm-dorsa vein pattern for accurate personal authentication. Knowledge-Based Systems, 81, 117–130

[6] Rosdi, B. A., Jaafar, H., Ramli, D. A. (2015). Finger Vein Identification using Fuzzy-based k-Nearest Centroid Neighbor Classifier. AIP Conference Proceedings, American Institute of Physics, 1643, 649–654

[7] Liu, M. (2010). Fingerprint classification based on Adaboost learning from singularity features. Pattern Recognition, 43, 1062–1070

[8] Zhang, Y., Ma, Y., Dai, X., Li, H., Mei, X., Ma, J.. (2021). Locality-constrained sparse representation for hyperspectral image classification. Information Sciences, 546, 858–870

[9] Kumar, K., Seal, A. (2020). Spectral embedded generalized mean based-nearest neighbors clustering with S-distance. Expert Systems With Applications, 114326

[10] Li, W., Chen, Y., Song, Y. (2020). Boosted K-nearest neighbor classifiers based on fuzzy granules. Knowledge-Based Systems, 195, 105606

[11] Mailagaha Kumbure, M., Luukka, P., Collan, M. (2020). A new fuzzy k-nearest neighbor classifier based on the Bonferroni mean. Pattern Recognition Letters, 140, 172–178

[12] Pan, Z., Wang, Y., Pan, Y. (2020). A new locally adaptive k-nearest neighbor algorithm based on discrimination class. Knowledge-Based Systems, 204, 106185

[13] Rastin, N., Jahromi, M. Z., Taheri, M. (2020). A Generalized Weighted Distance k-Nearest Neighbor for Multi-label Problems. Pattern Recognition, 107526

[14] Sarkar, M., Leong, T. Y. (2000). Application of K-nearest neighbors algorithm on breast cancer diagnosis problem. Proceedings of the AMIA Symposium American Medical Informatics Association, 759–763

[15] Herrera, F. (2012). Prototype Selection for Nearest Neighbor Classification : Taxonomy and Empirical Study. IEEE Transactions on Pattern Analysis and Machine Intelligence, 34, 417–435

[16] Sánchez, J. S., Pla, F., Ferri, F. J. (1997). On the use of neighbourhood-based non-parametric classifiers. Pattern Recognition Letters, 18, 1179–1186

[17] Chaudhuri, B. B. (1996). A new definition of neighborhood of a point in multi-dimensional space. Pattern Recognition Letters, 17, 11–17

[18] Grabowski, S. (2004). Limiting the Set of Neighbors for the K-NCN Decision Rule : Greater Speed with Preserved Classification Accuracy. Proceedings of the International Conference Modern Problems of Radio Engineering, Telecommunications and Computer Science, 511–514

[19] Gou, J, Yi., Z, Du, L., Xiong, T. (2012). A local mean-based k-nearest centroid neighbor classifier. Computer Journal, 55, 1058–1071

[20] Gou, J. (2012). Weighted K -nearest Centroid Neighbor Classification. Journal Of Computational Information Systems, 2, 851–860

[21] Mohd Asaari, M. S., Suandi, S. A., Rosdi, B. A. (2014). Fusion of Band Limited Phase Only Correlation and Width Centroid Contour Distance for finger based biometrics. Expert Systems with Applications, 41, 3367–3382

[22] Wu, J. D., Ye, S. H. (2009). Driver identification using finger-vein patterns with Radon transform and neural network. Expert Systems with Applications, 36, 5793–5799

[23] Yu, C. B., Tan, J., Yu, L., Tian, Y. L. (2013). A Finger Vein Recognition Method Based on PCA-RBF Neural Network. Applied Mechanics and Materials, 325–326, 1653–1658

[24] Wu, J. D., Liu, C. T. (2011). Finger-vein pattern identification using principal component analysis and the neural network technique. Expert Systems with Applications, 38, 5423–5427

[25] Radzi, S. A, Hani, MK, Bakhteri, R. (2016). Finger-vein biometric identification using convolutional neural network. Turkish Journal of Electrical Engineering & Computer Sciences, 24, 1863–1878

[26] Wu, J. D., Liu, C. T. (2011). Finger-vein pattern identification using SVM and neural network technique. Expert Systems with Applications, 38, 14284–14289

[27] Vlachos, M., Dermatas E. (2010). Supervised and Unsupervised Finger Vein Segmentation in Infrared Images Using KNN and NNCA Clustering Algorithms. XII Mediterranean Conference on Medical and Biological Engineering and Computing, 741–744

[28] Saw, T., Rosdi, B. A. (2011). Finger-Vein Identification using Pattern Map and Principal Component Analysis. IEEE International Conference on Signal and Image Processing Applications (ICSIPA), 530–534

[29] Lu, Y., Xie, S. J., Park, D. S. (2013). Finger Vein Identification Using Polydirectional Local Line Binary Pattern. 61–65

[30] Tan, D., Yang, J., Shi ,Y., Xu C. (2013). A Hierarchal Framework for Finger-Vein Image Classification. 2nd IAPR Asian Conference on Pattern Recognition, 833–837.

[31] Khellat-kihel, S., Cardoso, N., Monteiro, J., Benyettou, M. (2014). Finger Vein Recognition Using Gabor Filter and Support Vector Machine. International image processing, applications and systems conference, 1–6

[32] Murukesh, C., Nadesh, T. K., Thanushkodi, K. (2014). Secured Finger Vein Authentication System Using Contourlet Transform and SVM. Applied Mechanics and Materials, 573, 465–470

[33] Gholami, A., Hassanpour, H. (2014). Finger Vein Recognition in Radon Space Using Local Entropy Thresholding and Common Spatial Pattern. International Journal of Engineering-Transactions A: Basics, 28, 25–34

[34] Wang, D., Zhang, X., Fan, M., Ye, X. (2015). Hierarchical mixing linear support vector machines for nonlinear classification. Pattern Recognition, 1–13

[35] Khellat-kihel, S., Cardoso, N., Monteiro, J., Benyettou, M. (2014). Finger Vein Recognition Using Gabor Filter and Support Vector Machine. Image Processing, Applications and Systems Conference (IPAS), 2014 First International, 1–6

[36] Rakideh, M., Dardel, M., Pashaei, M. H. (2013). Finger Vein Recognition in Radon Space Using Local Entropy Thresholding and Common Spatial Pattern. 26, 1433–1444

[37] Kejun, W., Jingyu, L., Weixing, F. (2010). Finger Vein Identification Based On 2-D Gabor Filter. 21, 10–13.

[38] Sánchez, J. S., Pla, F., Ferri, F. J. (1998). Improving the k-NCN classification rule through heuristic modifications. Pattern Recognition Letters, 19, 1165–1170

[39] El Ayadi, M., Kamel, M. S., Karray, F. (2011). Survey on speech emotion recognition: Features, classification schemes, and databases. Pattern Recognition, 44, 572–587

[40] Mahalingam, G., Kambhamettu, C. (2012). Face verification of age separated images under the influence of internal and external factors. Image and Vision Computing, 30, 1052–1061

[41] Chuang, Z. J., Wu, C. H. (2004). Emotion recognition using acoustic features and textual content. IEEE International Conference on Multimedia and Expo (ICME), 53–56

[42] Hou, B., Yan, R. (2019). Convolutional Auto-Encoder Model for Finger-Vein Verification. IEEE Transactions on Instrumentation and Measurement, 69, 2067–2074

[43] Shazeeda, S., Rosdi, B. A. (2019). Nearest Centroid Neighbor Based Sparse Representation Classification for Finger Vein Recognition. IEEE Access, 7, 5874–5885

[44] Kamaruddin, N. M., Rosdi, B. A. (2019). A New Filter Generation Method in PCANet for Finger Vein Recognition. IEEE Access, 7, 132966–132978

[45] Shaheed, K., Liu, H., Yang, G., Qureshi, I. Gou, J., Yin, Y. (2018). A systematic review of finger vein recognition

techniques. Information, 9, 213
[46] Kauba, C., Prommegger, B., Uhl, A. (2019). Combined fully contactless finger and hand vein capturing device with a corresponding dataset. Sensors, 19, 5014