Multispectral Palmprint Recognition Using Textural Features

Shervin Minaee
ECE Department
New York University
Brooklyn, New York, USA
shervin.minaee@nyu.edu

AmirAli Abdolrashidi
ECE Department
New York University
Brooklyn, New York, USA
abdolrashidi@nyu.edu

Abstract—In order to use the identification means to the best extent, we need robust and fast algorithms and systems to process the data. Having palmprint as a reliable and unique characteristic of every person, we should extract its features based on its geometry, lines and angles. There are countless ways to define measures for the recognition task. To analyze a new point of view, we extracted textural features and used them for palmprint recognition. Co-occurrence matrix can be used for textural feature extraction. As classifiers, we have used the minimum distance classifier (MDC) and the weighted majority voting system (WMV). The proposed method is tested on PolyU multispectral palmprint dataset of 6000 samples and an accuracy rate of 99.96-100% is obtained for most scenarios which beats all previous works in multispectral palmprint recognition.

I. INTRODUCTION

THERE are many reasons to use identification; to make sure that the person about to receive information or rights is indeed the right one. Several ways of identification include keys, photographs, passwords and even biological samples. Many reasons necessitate the use of biometric characteristics of a person in their identification, including uniqueness, reliability and difficulty to forge. That identification can serve in personalized or secured applications or both. Other methods can be lost, forgotten, stolen or replicated without authorization and their purpose defeated rather easier.

Not surprisingly, there also exist many ways of identification based on biometric data such as fingerprints [1], iris patterns [2], face [3] and palmprints [4]. Among these, palmprints are simpler in the sense of acquisition and do not change over time so much. The key for their recognition is to extract the features of every person out of the prominent lines and wrinkles on their palms. Being a popular area of research, there are many sets of features and different approaches used for palmprint recognition [4]; however, general approaches for palmprint recognition are either transforming palmprints into another domain, namely transform-based approaches, or extracting principal lines and wrinkles and other geometrical characteristics as distinguishing factors.

Of the many researches in this area, a portion is based on transform domain features; for example, in [5], Wu proposed to use the energy of wavelet as features; and Kong implemented a system that uses Gabor-based features for palmprint recognition [6]. There are also quite a few line-based approaches, since palm lines capture the unique characteristics of a palmprint. Jia [8] used robust line orientation code for palmprint verification. Chen [7] extracted creases from palms in a way that they does not need any translations or rotations afterwards, and used them for palmprint matching. Some of the approaches use the palmprint information both in spatial and frequency domains. As an example, in [9], Minaee developed a multispectral pamlprint recognition program using both statistical and wavelet features and achieved a much higher accuracy rate than all the previous works in multispectral palmprint recognition. Also in [10], Xu sought to utilize quaternion principal component analysis for multispectral palmprint recognition which also resulted in a high accuracy.

Here we have decided to follow a new approach for palmprint recognition. We use textural features which are extracted from the co-occurrence matrix of every block whose concept will be elaborated in the next sections. It can incorporate adjacent blocks into the computations as well, sensing the overall texture. To test them, we have used the multispectral palmprint database created by the Polytechnic University of Hong Kong (PolyU) [11] which includes a set of 12 palmprint samples from 500 people taken in two days under four distinct light spectra: red, green, blue and infrared. Multispectral methods require different samples of the same object in order to make a better decision. In this paper, it is assumed that in the image acquisition section, four images of each palm sample are captured using CCDs. These images are preprocessed and the regions of interest (ROI) for each of them are extracted makingFor every spectrum, features are shown by $F_j^{(r)}$, $F_j^{(g)}$, $F_j^{(b)}$ and $F_j^{(i)}$ respectively. further preprocessing unnecessary. Three different palmprint samples from the used dataset are shown in Figure 1.

After feature extraction, we have to use a classification algorithm to identify palmprints. In this work, the two different methods of classification are minimum distance and weighted majority voting classifiers.

In this paper, the distribution of the paper contents is as follows; Section II provides a detailed explanation of the features and how to extract them; the weighted majority voting algorithm and minimum distance classifier are explained in

Fig. 1. Three sample palmprints

Section III; and results of the experiments and comparisons are given in Section IV.

II. FEATURES

Features are an inevitable part of machine learning. The more informative features we have, the greater accuracy we get. Therefore for any classification or regression algorithm, it is particularly essential to extract the right set of features. Feature extraction algorithms have a lot of applications in computer vision and object detection area. The most important step in image classification is that of defining a set of meaningful features to describe the pictorial information from the block of resolution cells. Once these features are extracted, categorization can be executed using any classification technique.

For palmprint recognition, features come from various origins and types with different advantages and disadvantages. Statistical features like mean and deviation of pixels are common. Another popular class of features are transform-based including Fourier-, Gabor- and wavelet-based. Spatial and geometrical features can also prove efficient, especially in medical applications as illustrated by [13] regarding chromosome segmentation.

In human interpretation of color photographs, textural, spectral and contextual features are the three fundamental pattern elements. Textural features contain the spatial information of intensity variation in a single band. Spectral features, on the other hand, describe the average intensity variation in different spectral bands. Contextual features contain information derived from neighboring regions of the area being analyzed [14]. Here we extracted a set of textural features which are based on an outstanding work published in 1973 [14], in which the author introduced a general procedure for extracting textural properties of blocks of image data. These features are calculated in the spatial domain, and the statistical nature of texture is taken into account in this procedure, which is based on the assumption that the texture information in an image is contained in the overall or the "average" spatial relationship that the gray tones in the image have to one another.

To extract the features, each image is divided into non-overlapping blocks of size $N \times N$, the co-occurrence matrix for each block is constructed, and finally 14 features will be extracted from it. These features contain information about textural characteristics of such image, such as homogeneity, gray-tone linear dependencies and structure, contrast, number and nature of boundaries present and the whole complexity of the image.

Before diving into the details of feature extraction, it is noteworthy how the proper block size should be chosen. If the chosen block size is too small, it will not have enough textural information to discriminate images of different people, while if a large block size is selected, that block may have patterns belonging to different categories. Therefore the right block size should not be in the extremes. Here $N = 16$ is chosen by trial and error based on the images in the dataset.

A. Co-occurrence Matrix

As the name suggests, co-occurrence matrix is a matrix defined over an image to measure the distribution of co-occurring intensity values for a given off-set. We can denote any image as a two-dimensional function which maps any pair of coordinate to an intensity value, i.e., $I : X \times Y \to G$, where $X = \{1, 2, 3, ..., N_x\}$ and $Y = \{1, 2, 3, ..., N_y\}$ denote the horizontal and vertical spatial domains respectively and G denotes the set of all grayscale levels (intensities), in the usual images $G = \{0, 1, 2, ..., 255\}$.

Then the co-occurrence matrix P of image I with an offset (Δ_x, Δ_y) can be defined as:

$$P_{\Delta_x, \Delta_y}(i, j) = \sum_{m=1}^{N_x} \sum_{n=1}^{N_y} \delta(I(m, n) - i)\delta(I(m + \Delta_x, n + \Delta_y) - j)$$

where $\delta(x)$ denotes the discrete Dirac function, which is 1 when the argument is zero, and 0 elsewhere. Therefore $P_{\Delta_x, \Delta_y}(i, j)$ counts how many times two pixels with intensities i and j are located in a distance of (Δ_x, Δ_y) from each other. The offset (Δ_x, Δ_y) depends on the direction θ. Here we used $(\Delta_x, \Delta_y) = (1, 0)$. The neighborhood direction θ can be defined accordingly:

$$\theta = tan^{-1}(\frac{\Delta_y}{\Delta_x})$$

It should be noted that the co-occurrence matrix has a size of $N_g \times N_g$, where N_g denotes the number of gray-levels in the image. Here we quantized our images with quantization step-size of 8, therefore $N_g = 32$.

As an example, consider the image matrix A as:

$$A = \begin{pmatrix} 1 & 1 & 2 & 1 \\ 2 & 3 & 1 & 2 \\ 2 & 1 & 3 & 2 \\ 3 & 3 & 2 & 1 \end{pmatrix}$$

Here the matrix A has only three different grayscale levels. Therefore its co-occurrence matrix has a size of 3×3. The co-occurrence matrix of A for $(\Delta_x, \Delta_y) = (1, 0)$ will be:

$$C = \begin{pmatrix} 1 & 2 & 1 \\ 3 & 0 & 1 \\ 1 & 2 & 1 \end{pmatrix}$$

Here, for example $C(1, 2)$ counts how may times the cases $A(i, j) = 1$ and $A(i + 1, j) = 2$ occur in matrix A, which is twice.

B. Textural Feature Extraction From Co-occurrence Matrix

After the co-occurrence matrix has been extracted, the following 14 textural features for each block may be extracted with ease. These features, which are described below, are similar to those in [14]. For notation brevity, we first define the following terms derived from the matrix which will be used in the definition of the used features:

$$p(i,j) = P(i,j)/R, \quad \text{Normalized Co-occurrence Matrix}$$

$$p_x(i) = \sum_{j=1}^{N_g} p(i,j) \quad \text{Marginal Probability}$$

$$p_y(j) = \sum_{i=1}^{N_g} p(i,j) \quad \text{Marginal Probability}$$

$$p_{x+y}(k) = \sum_{i+j=k} \sum p(i,j), \quad k = 2, 3, ..., 2N_g.$$

$$p_{x-y}(k) = \sum_{|i-j|=k} \sum p(i,j), \quad k = 0, 1, ..., N_g - 1.$$

$$HXY = -\sum_i \sum_j p(i,j) \log\big(p(i,j)\big)$$

$$HXY1 = -\sum_i \sum_j p(i,j) \log\big(p_x(i)p_y(j)\big)$$

$$HXY2 = -\sum_i \sum_j p_x(i)p_y(j) \log\big(p_x(i)p_y(j)\big)$$

$$Q(i,j) = \sum_k \frac{p(i,k)p(j,k)}{p_x(i)p_y(k)}$$

Now we define the following 14 textural features using these terms.

$$f_1 = \sum_i \sum_j \big[p(i,j)\big]^2, \qquad \text{Angular Second Moment}$$

$$f_2 = \sum_{k=0}^{N_g-1} k^2 p_{x-y}(k), \qquad \text{Contrast}$$

$$f_3 = \frac{\sum_i \sum_j ijp(i,j) - \mu_x\mu_y}{\sigma_x\sigma_y}, \qquad \text{Correlation}$$

$$f_4 = \sum_i \sum_j (i-\mu)^2 p(i,j), \qquad \text{Variance}$$

$$f_5 = \sum_i \sum_j \frac{1}{1+(i-j)^2} p(i,j), \quad \text{Inverse Diference Moment}$$

$$f_6 = \sum_{k=2}^{2N_g} k p_{x+y}(k), \qquad \text{Sum Average}$$

$$f_7 = \sum_{k=2}^{2N_g} (k-f_6)^2 p_{x+y}(k), \qquad \text{Sum Variance}$$

$$f_8 = -\sum_{k=2}^{2N_g} p_{x+y}(k) \log(p_{x+y}(k)), \quad \text{Sum Entropy}$$

$$f_9 = -\sum_i \sum_j p(i,j) \log(p(i,j)), \quad \text{Entropy}$$

$$f_{10} = \sum_{k=0}^{N_g-1} (k-\mu_{x-y})^2 p_{x-y}(k), \quad \text{Diference Variance}$$

$$f_{11} = -\sum_{k=0}^{N_g-1} p_{x-y}(k) \log\big(p_{x-y}(k)\big), \quad \text{Difference Entropy}$$

$$f_{12} = \frac{HXY - HXY1}{max\{HX, HY\}}$$

$$f_{13} = \sqrt{1 - exp[-2(HXY2 - HXY)]}$$

$$f_{14} = \sqrt{\text{Second largest eigenvalue of } Q}$$

Here μ_x and σ_x denote the mean and standard deviation of the marginal distribution P_x respectively. The same applies to μ_y and σ_y. Some of these features are related to the entropy of the co-occurrence matrix distribution. We have provided a very short introduction to entropy and mutual information in Appendix I. In the original paper, these 14 features have been defined, but it is suggested to calculate them for 4 angular co-occurrence matrices and take the average and range of each feature as a new feature, resulting in 28 features to be used. Here we use the 14 features for $\theta = 0$. The feature vector can be denoted as $\mathbf{f} = (f_1, f_2, ..., f_{14})^{\mathsf{T}}$. It is necessary to find the mentioned features for each block of a palmprint. If each palm image has a size of $s_1 \times s_2$, the total number of non-overlapping blocks will be:

$$M = \frac{s_1 s_2}{N^2}$$

Therefore there are M such feature vectors, $\mathbf{f}^{(m)}$. Similarly, they can be put in the columns of a 2-dimensional matrix to produce the feature matrix of that palmprint, $\mathbf{F}$:

$$\mathbf{F} = [\mathbf{f}^{(1)}; \mathbf{f}^{(2)}; ...; \mathbf{f}^{(M)}]$$

Therefore there will be $14 \times \mathbf{M}$ features for each image (Here $\mathbf{M} = 64$).

III. RECOGNITION ALGORITHM

After capturing the features of all people, a classifier should be used to compare the features of each test palmprint to all the training samples available and find its closest match. In this paper, two different classifiers are employed for this task. Weighted majority voting is inspired by counting votes from the features to the subjects. The minimum distance classifier, on the other hand, finds the minimum distance between the feature matrices of the training samples and test subjects. They are both explained in this section. Since we have a large dataset, our only objective is to minimize the recognition error on the test samples, but when using a small dataset, one should try to find a model which maximizes the accuracy and minimizes the risk of over-fitting. One such work is presented in [15] where the author proposed a measure related to over-fitting and tried to jointly minimizes the error and over-fitting measure.

A. Weighted Majority Voting

In voting, there are referees that decide the answer by themselves and their votes are taken into account based on their importance, or weight. This scheme is very popular in learning algorithms and artificial intelligence. Unweighted voting is when we know all features should have the same effect on the outcome, but usually, each feature should use a different weight, either fixed or adaptive. When added, the total score will decide which person is the owner of the test image. Here the voters are the used features and they are weighted in a fixed manner. Apart from its simplicity, it also takes little time.

First, the images of every single person are rearranged in the database so that the training part can use uniform data from all of the set of the 12 images. The training features are then collected, averaged and stored. Later, the other images are used as test subjects and the distance between the average feature matrix and that of every subject is measured. The case with the least distance with the subject is given points equal to the weight of the feature. In the end, the person gaining the maximum points is the winner.

For every $\mathbf{f}_i^{(t)}$, the voting result is:

$$k^*(i) = \arg\min_k ||\mathbf{f}_i^{(t)} - \mathbf{f}_i^k||_2$$

When $\mathbf{f}_i$ finds the person with minimum distance to the test subject, that person receives a point. If the score of person j based on $\mathbf{f}_i$ is denoted by $w_i S_j(i)$ or $w_i I(j = \arg\min_k |\mathbf{f}_i^{(t)} - \mathbf{f}_i^k|)$, where w_i is the weight of the feature i and $I(x)$ is an indicator function, the total score of the j-th training sample based on all the features in the scope of all the colors can be computed.

$$S_j = \sum_{All\ colors} \sum_{i=1}^{i_{max}} w_i I(j = \arg\min_k |\mathbf{f}_i^{(t)} - \mathbf{f}_i^k|)$$

In the end, the identification factor j^* is:

$$j^* = \arg\max_j \left[S_j\right] = \arg\max_j \left[\sum_{All\ colors} \sum_i w_i S_j(i)\right]$$

B. Minimum Distance Classifier

The minimum distance classification is quite popular in the template matching area. It finds the distance between the features of the training samples and those of an unknown subject, and picks the training sample with the minimum distance to the unknown as the answer. To put it in equation, if we show the features of the test subject as F^* and those of the test sample i with F_i, the test subject is matched to the sample that satisfies the following:

$$i^* = \arg\min_i \left[dis(F^*, F_i)\right]$$

Here, each feature matrix will have a size of 14×64 due to the size of the images and the blocks. M of the 12 samples from every person are assigned as training and the rest as test cases, adding up to $500(12 - M)$ subjects. The feature matrix is defined as the average of the feature matrices of the

M training images. For an unknown sample with the feature matrix F^*, the following distance will be:

$$dis(F^*, F_i) = \sum_{m=1}^{14} \sum_{n=1}^{64} w_m \alpha_m (F_{mn}^* - F_{i,mn})^2$$

Now each row has a weight of $w_m \alpha_m$, where α_m is a feature normalizing factor trying to map all features into the same range and is defined as the reciprocal of the mean value of the corresponding feature of all training samples, while w_m is the feature importance factor which is higher as the usefulness of the feature increases. Here w_m is defined as the recognition accuracy when the m-th row of the feature matrix is used for recognition. We should find the distance above for all the spectra by comparing the images in the same color. Next, the distance between a test image and the i-th training sample will be defined as the average of the distances of their corresponding spectra. In the end, the prediction for a test image with the feature matrix F^* is:

$$i^* = \arg\min_i \left[dis(F^*, F_i)\right]$$

IV. RESULTS

In our dataset, each image is preprocessed and aligned and has the resolution of 128×128. In each setting of our experiment, we have performed recognition for various combinations of training data and test subjects. Whenever the test data does not match our expectation, it is an ID fail or misidentification.

The results from majority voting and minimum distance classifications are shown in Table I. For majority voting, due to the much shorter time it takes, every test is repeated 10 times and their average is recorded. For the minimum distance classifier, the image permutations are adjacent. For example, two neighbor minimum distance cases are common in all their training data selections but one.

TABLE I
ACCURACY RATES OF MINIMUM DISTANCE CLASSIFIER AND WEIGHTED
MAJORITY VOTING ALGORITHM

Training sample fraction	Minimum Distance		Majority Voting	
	No feature weight	Weighted features	No feature weight	Weighted features
4/12	96.45	96.35	99.96	99.96
5/12	95.71	97.46	100	99.99
6/12	98.07	97.80	100	100
7/12	97.40	96.92	100	100
8/12	98.65	98.35	99.99	99.99
9/12	97.60	97.33	100	100
10/12	98.60	98.40	100	100

Table II shows a comparison of the results of our work and those of three other accurate and relatively newer algorithms. Note that the blank spaces under QPCA are due to them being not reported in the source.

Table II shows that the performance of the majority voting classifier is much more efficient than the minimum distance classifier. On the whole, our algorithm has a higher accuracy rate compared to previously done works and also slightly

TABLE II
COMPARISON WITH OTHER ALGORITHMS FOR PALMPRINT RECOGNITION

Training sample fraction	QPCA [10]	Hybrid feature [13]	Stat/Wave* [9] (MDC)	Proposed method (WMV)
6/12	98.13%	98.88%	100%	100%
5/12	-	98.45%	99.77%	99.99%
4/12	-	98.08%	99.65%	99.96%

*Statistical and wavelet features

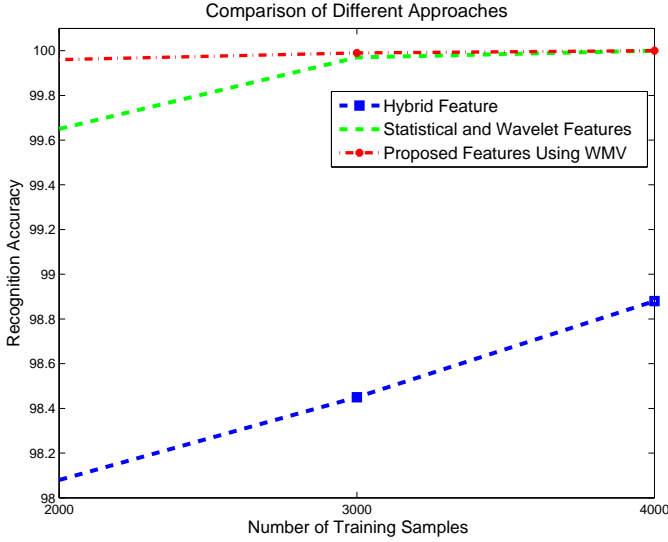outperforms the results from [9]. The comparison between our work and some of previous ones is illustrated in Figure 2.

Fig. 2.   Comparison of different palmprint recognition approaches

The method is tested using MATLAB on a laptop with Windows 7 and Core i7 running at 2GHz. The calculation time for weighted majority voting is 0.06s per test, while minimum distance classifier takes 0.09s per test.

CONCLUSION

This paper proposed a set of textural features based on co-occurrence for palmprint recognition. This method senses the textures of the images and extracts 14 features from them. Two different classifiers, weighted majority voting and minimum distance classifiers, are also used to perform the recognition. The proposed scheme has advantages over many older popular methods. It has a very high accuracy rate as well as a low processing time, making it possible to use in real-time applications. The calculation of the features are also straightforward. There are many speculations for the future including applying the same features to other biometrics such as fingerprints and iris patterns.

ACKNOWLEDGMENTS

APPENDIX I. ENTROPY AND MUTUAL INFORMATION

Entropy measures the average amount of information in one or more than one random variables. It can be interpreted of as the amount of uncertainty in a random variable. The idea is that the less likely an event is, the more information it provides when it occurs. For example, if the probability of head in a coin is 1, flipping a coin and observing head does not provide any information, because it is already expected. The entropy which we will explain here is proposed by Claude E. Shannon in his 1948 paper "A Mathematical Theory of Communication" [16]. Based on a few mathematical assumptions, it makes sense to define information as the negative of the logarithm of the probability distribution. The entropy H of a random variable $X = \{x_1, x_2, ..., x_n\}$ with the probability mass function of $P(X)$ is defined as:

$$H(X) = -\sum_{i=1}^{n} p(x_i) \log p(x_i)$$

We can define the joint and conditional entropy of two random variables X and Y as:

$$H(X,Y) = -\sum_{i} \sum_{j} p(x_i, y_j) \log p(x_i, y_j)$$

$$H(X|Y) = -\sum_{i} \sum_{j} p(x_i, y_j) \log p(x_i, y_j)$$

Another important concept in information theory is the mutual information which measures the mutual dependence between two variables. Mutual information between the variables X and Y is denoted by $I(X;Y)$ and is defined as:

$$I(X;Y) = H(X) - H(X|Y) = H(Y) - H(Y|X) =$$

$$\sum_{i} \sum_{j} p(x_i, y_j) \log \left(\frac{p(x_i, y_j)}{p(x_i)p(y_j)} \right)$$

For better illustration, the individual $H(X)$ and $H(Y)$, joint $H(X,Y)$ and conditional entropies, $H(X|Y)$ and $H(Y|X)$, and mutual information of a pair of correlated random variables X and Y are shown in Figure 3. The intersection of supports of two random variables denotes the mutual information.
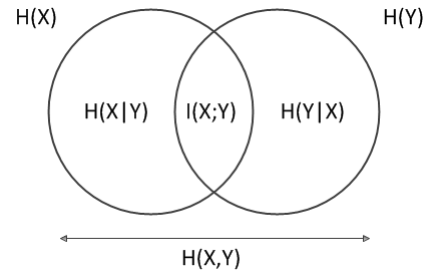

Fig. 3.   Information diagram of two random variables

Both $H(X)$ and $I(X;Y)$ are positive values for discrete random variables X and Y. Based on the introduction here, HXY in the co-occurrence matrix is in fact the joint entropy.

$HXY1$ will be similar to $I(X;Y) + H(X,Y)$ where random variables X and Y denote the probability distribution along rows and columns respectively, and $HXY2$ will be similar to $H(X) + H(Y)$.

REFERENCES

[1] A. K. Hrechak and J. A. McHugh, "Automated fingerprint recognition using structural matching," Pattern Recognition 23.8 (1990): 893-904.

[2] D. M. Monro, S. Rakshit and D. Zhang, "DCT-based iris recognition," Pattern Analysis and Machine Intelligence, IEEE Transactions on 29.4 (2007): 586-595.

[3] W. Zhao, R. Chellappa, P. J. Phillips and A. Rosenfeld, "Face recognition: A literature survey," ACM Computing Surveys (CSUR) 35, no. 4 (2003): 399-458.

[4] S. A. Mistani, S. Minaee and E. Fatemizadeh, "Multispectral palmprint recognition using a hybrid feature," arXiv preprint arXiv:1112.5997 (2011).

[5] X. Wu, K. Wang and D. Zhang, "Wavelet energy feature extraction and matching for palmprint recognition," Journal of Computer Science and Technology 20.3 (2005): 411-418.

[6] W. K. Kong, D. Zhang and W. Li, "Palmprint feature extraction using 2-D Gabor filters," Pattern recognition 36.10 (2003): 2339-2347.

[7] J. Chen, C. Zhang and G. Rong, "Palmprint recognition using crease," Proceedings, 2001 International Conference on Image Processing, Vol. 3. IEEE, 2001.

[8] W. Jia, D. Huang and D. Zhang, "Palmprint verification based on robust line orientation code," Pattern Recognition 41.5 (2008): 1504-1513.

[9] S. Minaee and A. Abdolrashidi, "Highly Accurate Multispectral Palmprint Recognition Using Statistical and Wavelet Features," arXiv preprint arXiv:1408.3772 (2014).

[10] X. Xu and Z. Guo, "Multispectral palmprint recognition using quaternion principal component analysis," IEEE Workshop on Emerging Techniques and Challenges for Hand-Based Biometrics, pp. 15, 2010.

[11] http://www.comp.polyu.edu.hk/˜biometrics/MultispectralPalmprint/MSP.htm

[12] D. Zhang et al., "An online system of multispectral palmprint verification," Instrumentation and Measurement, IEEE Transactions on 59.2 (2010): 480-490.

[13] S. Minaee, M. Fotouhi and B. H. Khalaj, "A geometric approach for fully automatic chromosome segmentation," arXiv preprint arXiv:1112.4164 (2011).

[14] R. M. Haralick, K. Shanmugam and I. Dinstein, "Textural features for image classification," IEEE Transactions on Systems, Man and Cybernetics, no. 6 (1973): 610-621.

[15] S. Minaee, Y. Wang and Y. W. Lui, "Prediction of Longterm Outcome of Neuropsychological Tests of MTBI Patients Using Imaging Features," Signal Processing in Medicine and Biology Symposium (SPMB), IEEE, 2013.

[16] C. E. Shannon, "A mathematical theory of communication," ACM SIGMOBILE Mobile Computing and Communications Review 5.1 (2001): 3-55.