A SCORE BASED INDEXING SCHEME FOR PALMPRINT DATABASES

Ashish Paliwal, Umarani Jayaraman and Phalguni Gupta

Department of Computer Science and Engineering
Indian Institute of Technology Kanpur, Kanpur 208016, INDIA
E-mail:*(apaliwal,umarani,pg)@cse.iitk.ac.in*

ABSTRACT

This paper proposes an efficient retrieval technique which uses a new indexing scheme for a large palmprint database. For a given palmprint query image, the proposed indexing scheme makes use of match score vectors to determine an index and uses this index to reduce the search space of the database. Finally, the retrieval technique uses palmprint texture to find the top t best matches from the reduced search space. The proposed technique has been tested on publicly available palmprint database *viz* PolyU [1] of 7752 images and is found to be an efficient technique based on score vector indexing scheme. The test reveals that *hitrate* of 98.01% is acheived for the bin-success rate of 98.28%. The proposed indexing scheme is found to be more powerful than any other known indexing scheme based on match score vectors.

Index Terms— Indexing, VA+ File, Match Score, SURF, Palmprint Biometrics.

1. INTRODUCTION

The palmprint based biometric system provides an automated method to authenticate an individual with the help of palmprint textural patterns. For a query palmprint image, the problem of palmprint identification system is to find the top t best matches from the database consisting of N palmprint images. For a large database, it needs too much time to make $1 : N$ searches. In order to design an efficient system, the matching engine needs to search in a reduced space.

Most of the indexing schemes use the features of the image to index the database [2][3][4]. However, there exist a few indexing schemes based on match scores. First attempt to index the database based on score matrix has been made in [5]. This approach relies on pre-computed values of one-to-one match score for each image in the database. Let A_{ij} be the match score between image i and image j in the database where A_{ij} has the value between 0 and 100. It can be stated that the matrix $A = \{A_{ij}\}$ is a symmetric matrix with diagonal elements having value 100. This algorithm is an iterative one. At the i^{th} iteration, match score between a query image q and the image I_j of the database is obtained where I_j is the image index obtained based on nearest match with respect to the match score in the score matrix A. At the first iteration, $I_j = I_1$. The algorithm terminates when the match score between a query q and the image index I_j is larger than the threshold. This indexing scheme is able to reduce the number of images to be compared by skipping out the images which are irrelevant to a query image. However, it takes linear time in the worst case. Also, this approach requires large amount of space to store $N \times N$ matrix for a database of size N. This leads to increase in space complexity. Further, when a new subject is to be enrolled, this matrix has to be updated and as a result, it takes considerable amount of time.

Another approach to index the multimodal biometric database is based on match scores generated by the matcher [6]. This indexing scheme relies on the use of a small set of reference images for each modality. For a particular modality, match score is generated by matching all images in the database against a reference image which results a set of match scores. Let $R = \{r_1, r_2, ...r_m\}$ be an ordered set of reference images and N be the number of images in the database where $|R| << N$. Then clearly there are $|R| \times N$ match scores obtained for each modality. Let $S_i(R)$ be the score vector of image i in the database against each reference images in R. In case of identification, for a query image q, the score vector $S_q(R)$ is generated by comparing the query q to each reference image in R. This method determines all images with index i, $i = 1, 2, ..., N$ such that the correlation between $S_i(R)$ and $S_q(R)$ is greater than the threshold value. These images are considered likely to be matched with the query image q. This method has been extended for multimodal biometrics where either score vectors are concatenated to each other at various stages or union or intersection of all the images is selected for different modalities. For larger biometric databases, matching between two images is much slower than computing the correlation coefficient between two vectors. However, the drawback of this indexing scheme is that it has to read the whole match score vectors for computing correlation coefficient. Thus the performance of this indexing scheme largely depends on the size of match score vectors. If the match score vectors are too large then performing linear scan on this requires considerable amount of time. Also, the performance of biometric identification system may be poor if the matching is done based only on the match score vectors.

This paper proposes an indexing scheme which reduces the amount of match score vectors that have to be read while computing correlation coefficient through some dynamic index structure and then palmprint texture for retrieval from the reduced search space which improves the overall performance of palmprint recognition system. It uses VA+ file [7] which is an approximation-based organization of high-dimensional match score vectors which make the unaviodable sequential scan as fast as possible. The indexing scheme has been discussed in Section 2. It reduces the search space of the large database effectively. In order to obtain the top t best matches from the reduced search space, texture based palmprint retrieval technique has been given in Section 3. Performance of the proposed technique is analysed in the next section. Conclusion is given in the last section.

2. THE PROPOSED INDEXING SCHEME

This section proposes a score based indexing scheme which makes use of Vector Approximation (VA+) file to index very high-dimensional data. In the proposed technique, high-dimensional data are the match score vectors generated between each image in the database with all the images in training set which contains images having distinct characteristics. Fig. 1 shows the process flow of the proposed indexing scheme.

2.1. Selection of Training Set

Creation of the training set T plays a very crucial role in indexing scheme and retrieval technique. If the size of T is very small, the score vector generated out of T does not provide enough information. On the other hand, if it is very large, it increases the computation time. Hence an optimum size of training set T having distinct characteristics should be able to provide enough information for identification but with miminum computational time. For n subjects, define a set $A = \{a_1, a_2, ...a_n\}$ satisfying that a_i is a set of p_i images of subject i such that $\bigcup_{i=1}^{n} p_i$ is the total number of images in the database, N. For each subject i, the image having maximum variance in a_i is selected. Let S be the set of all selected images. For each image in S, the match score is determinined against all other images in S and variance of these match scores is computed. Experimentally it has been obtained that the values of variance obey normal distribution. All the image from set S whose variance is greater than the mean (μ) are selected for the training set T. Let the size of training set T be M. The images in T are kept in such a way that variance of image i is greater than or equal to that of image j in T for $i < j$. The algorithm 1 explains the process of creation of training set.

2.2. Generating Score Vectors

Let $D = \{d_1, d_2, .., d_N\}$ be the database of N images. Then for each image d_i, match score is obtained against each image in the training set T. Let $S_i(T)$ be the vector of M elements containing these match scores. It can be noted that if the two images d_i and d_j belong to the same subject then the difference between their score vectors $S_i(T)$ and $S_j(T)$ is comparatively less. This difference can be used to identify the corresponding candidate from the database for a query image q. Once the score vectors for all the images in the database are computed, these score vectors are indexed based on VA+ file.

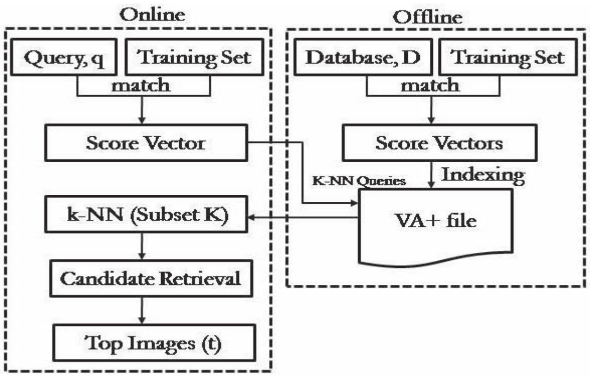

Fig. 1. Process Flow of Proposed Indexing Scheme

Algorithm 1 SELECTION OF TRAINING SET

```
Input: The set $S = \{s_1, s_2, .., s_n\}$ is the set of images of $n$ subjects and $s_i$ is the image having maximum variance in $a_i, i = 1, 2, .., n$.
Output: The training set $T = \{t_1, t_2, ..., t_M\}$ $(T \subset S)$ where $|T| << |S|$.
For each $s_i$ and $s_j$ in $S$, compute match score $m_{ij}$ between $s_i$ and $s_j$
Similarly, For each $s_i$ in $S$, compute variance $v_i$ of $[m_{i1}, m_{i2}, ..., m_{in}]$
Compute mean $\mu$ of $v_1, v_2, ..., v_n$.
Select an image $s_i$ having $v_i \geq \mu$ for the training set $T$.
Arrange the images in $T$ such that these variances are in decreasing order.
```

2.3. Indexing Score Vectors

The VA+ file is a flat array of approximations which can be used to index very high-dimensional data. The idea of VA+ file is to compress the score vectors by using some approximation which reduces the amount of data that must be read during the search. Also scanning of these approximation bits is faster and of less overhead than the actual score vectors. The VA+ file divides the vector space into 2^b rectangular cells where b denotes a user specified number of bits. Instead of hierarchically organizing these cells like tree based structures, the VA+ file allocates a unique bit-string of length b to each cell and approximates score vectors that fall into a cell by that bit-string. The VA+ file is an array of these compact approximations of score vectors. The *k-NN queries* are processed by first scanning the entire approximation file and then excluding the vast majority of score vectors from the search based only on these approximations. For a palmprint query image q, the score vector $S_q(T)$ is obtained after matching q with all the images in T and then *k-NN queries* proposed in [7] is called to obtain a subset K of size k images which are the nearest to the query image q. The subset K contains all images satisfying,

$$\|q - i\| \leq \|q - n\|, \forall i \in K \text{ and } \forall n \in (D - K) \quad (1)$$

where $\|.\|$ is a distance measure. Note that the space required to store the VA+ file is very small as compared to score matrix proposed in [5][6]. This, in turn, means that VA+ file can be handled efficiently in primary memory.

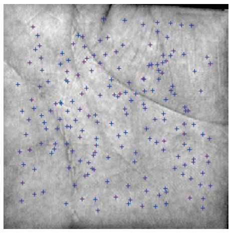

Fig. 2. Detected Keypoints of Palmprint Image from PolyU

3. CANDIDATE RETRIEVAL

This section discusses the technique to retrieve top t matches from a subset K based on the palmprint texture patterns. The texture pattern of palmprint image can help to improve the matching performance. After indexing based on score vector for query q, if its corresponding identity exists in the subset K then there is a high possibility to get it in the top t matches. Such an improvement of palmprint recognition system has happened due to the integration of a local feature descriptor named Speeded-Up Robust Features (SURF) [8]. Local features are extracted by finding the key points in an image and forming descriptor vector around each detected key point. SURF has more discriminative power than any other local feature descriptors such as SIFT [9]. Also it can be computed more efficiently and results lower dimensional features and hence the matching is much faster. In contrast to SIFT which approximates Laplacian of Gaussian (LOG) with difference of Gaussians (DOG), SURF approximates second order Gaussian derivatives with box filters. For each pixel in palmprint image, Hessian matrix at scale σ is obtained. The determinant of Hessian matrix is used to select location and scale. The local maxima found using approximated Hessian matrix determinant are interpolated in scale and image space. Fig. 2 shows detected SURF key points of a palmprint image of PolyU database [1]. SURF descriptors are obtained by taking a rectangular window around each detected key point. The window is splitted into 4 × 4 sub-regions. For each sub-region, Haar wavelet responses are extracted. The wavelet response in horizontal (d_x) and vertical (d_y) directions are summed up for each sub-region. The absolute values of wavelet responses $|d_x|$ and $|d_y|$ are summed up to find the polarity of image intensity changes. Hence, feature vector for each sub-region is given by

$$f = \left(\sum d_x, \sum d_y, \sum |d_x|, \sum |d_y|\right) \quad (2)$$

Addition of the descriptor vectors from all 4 × 4 sub regions results feature descriptor of length 64. The descriptor vector of all key points forms the feature vector F_J of J^{th} palmprint image in the subset K. Finally, the descriptor vectors are matched between query q and images of subset K. The matching is based on distance between two descriptor vectors. The images which have the maximum matching points are displayed as the top t best matches for a given query image.

4. PERFORMANCE EVALUATION

The proposed system has been tested on PolyU palmprint database [1] which contains 7752 gray scale images of 386 subjects. Around 20 samples from each of these palms have been collected in two sessions where about 50% of samples are captured in the first session. In each subject, one image is considered for testing and remaining images are considered for training. To quantify the performance of the proposed indexing scheme, two measures namely *bin-success rate* and *hitrate* are used.

- The *bin-success rate* (B_s) is defined by $B_s = \frac{T}{Q} \times 100\%$. Here T is the number of cases, where corresponding identity of a query image lie in the subset K and Q is the total number of distinct queries made. Thus *bin-miss rate* (B_r) is given by $B_r = 100 - B_s$.
- The *hitrate* (H_r) is the ratio of the number of times (H) that the corresponding identity of a query q has been found as one of the top t matches to the total number of attempts made (L) $i.e.$, $H_r = \left(\frac{H}{L}\right) \times 100\%$.

The experiment has been conducted to determine the suitable size of the training set T such that one can achieve maximum bin-success rate with the use of minimum number of nearest neighbors (k) for VA+ file. It has been observed that there exists a trade-off relationship among these three factors. For various size of nearest neighbors and that of training set, the bin-success rates are plotted in Fig. 3. It is observed from the figure that as the number of nearest neighbors for any size of training set increases from 30 to 100, the bin-success rate increases drastically and for the number of nearest neighbors larger than 100, the improvement over bin-success rate is not that significant. Again, it is seen that for the increase in the size of training set from 50 to 171 for any size of nearest neighbors, there is a significant increase in the bin-success rate. However, maximum bin-success rate which is 98.28% is achieved for the size of training set lying between 171 and 250 and for the size of nearest neighbors lying between 250 and 350. Thus the bin-success rate is almost saturated at 98.28% for the training set size of 171 and the number of nearest neighbors of 250. Further, the proposed technique has been tested on a PC which has 2GB RAM, intel dual core of speed 2.8 GHz and time required to get the top t matches against a query image for various training set size and various

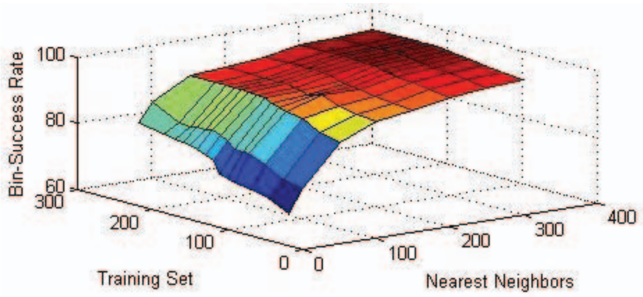

Fig. 3. Bin-Success Rate against Training Set and Nearest Neighbors

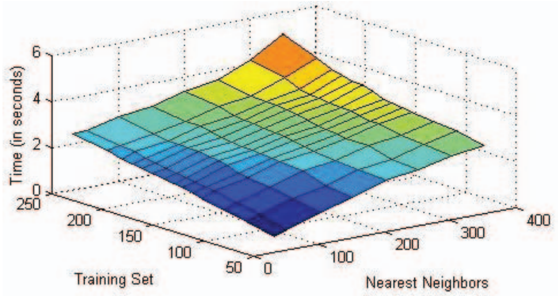

Fig. 4. Time against Training Set and Nearest Neighbors

number of nearest neighbors is measured. Fig. 4 shows the 3D plot giving the information of time to get top t matches against training set size and the number of nearest neighbors and vice versa. It has been observed that the time is drastically increasing while the training set size increases from 171 to 250 and the number of nearest neighbors from 250 to 350. Thus the time required to process a query on the machine with a bin-success rate of 98.28% using the training set of size 171 and 250 nearest neighbors is 3.22 sec.

It is also observed that bin-success rate of 73% is acheived in 1.6 sec. Multiplicative factors of time required to achieve various bin-success rate with reference to the time needed to achieve 73% bin-success rate are plotted in Fig. 5. This factor is small for the bin-success rate less than 85% and increases sharply while bin-success rate is more than 97%. Given a bin of 250 nearest neighbors for a query image obtained with the help of a training set of size 171, we have used SURF descriptor to obtain top t matches. Table 1 gives the percentage of identifying the image correctly within top t matches (hitrate) for different values of t.

(C=171 and K=250), Top Matches (t)					
Hitrate %	1	3	5	10	15
	96.01	97.35	97.61	97.88	98.01

Table 1. Top Matches against Hitrate

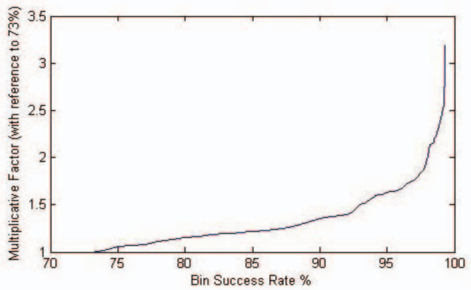

Fig. 5. Bin Success Factor

5. CONCLUSION

This paper proposes an indexing scheme for palmprint biometric database and has shown its effectiveness to reduce the search space. In the proposed method the matching between two score vectors is much faster instead of matching between two images. Again, an approximation-based organization of high-dimensional match score vectors through VA+ file make the unaviodable sequential scan as fast as possible and as a result, the response time is faster. Finally SURF local descriptor which is used for palmprint retrieval in order to get the top t matches further enhances the performance of palmprint recognition system.

6. REFERENCES

[1] "Palmprint Database," *http://www4.comp.polyu.edu.hk/ biometrics/*.

[2] J Umarani, P Surya, and P Gupta, "Indexing Multi. Biometric Data. Using Kd-Tree with Feat. Lev. Fusion," 2008, pp. 221–234.

[3] Fang Li and Maylor K. H. Leung, "Hierarchical Identification of Palmprint Using Line-Based Hough Transform," in *Proc of ICPR '06*, 2006, pp. 149–152.

[4] You J. Li, W. and D Zhang, "Texture-Based Palmprint Retrieval Using a Layered Search Scheme For Pers. Identi.," *IEEE. Tran. Multimedia*, pp. 891–898, 2005.

[5] M Takuji, M Masahito, S Koichi, and Y Yasushi, "Identi. Algo. Using a Matching Score Matrix," *IEICE Trans Inf Syst*, vol. 84, no. 7, pp. 819–824, 2001.

[6] A Gyaourova and A Ross, "A Coding Scheme for Indexing Multi. Bio. Databases," 2009, pp. 93–98.

[7] R Weber, H Schek, and S Blott, "A Quanti. Study for Simila. Search in High-Dimen. Spa.," 1998, pp. 194–205.

[8] H Bay, A Ess, T Tuytelaars, and L Vangool, "SURF," *CVIU*, vol. 110, no. 3, pp. 346–359, 2008.

[9] D.G Lowe, "Object Recogn. from Local Scale-Invariant Features," 1999, pp. 1150–1157.