Minutiae Based Palmprint Indexing

Alfredo Muñoz-Briseño(✉), José Hernández-Palancar, and Andrés Gago-Alonso

Advanced Technologies Application Center (CENATAV), Havana, Cuba
{amunoz,jpalancar,agago}@cenatav.co.cu

Abstract. Nowadays, palmprint identification has emerged as a very competitive and used technique in biometric systems. In these applications, efficiency is a very important but challenging problem for some reasons. The area of a palmprint is bigger than the one in a fingerprint. In this way, the amount of minutiae is also bigger and distortions are much more critical. On the other hand, reduction of the search space is essential in the process of identification. In this paper, a new palmprint indexing algorithm based on minutiae is proposed. Minutiae are very used by experts in order to perform manual matching, so this proposal can use features corrected by humans. The presented algorithm also uses a representation of palmprints based on an expanded triangle set, that proves to be very tolerant to minutia displacements on impressions. With this representation, a small set of features is extracted from minutia triplets. This aspect is very critical in the context of palmprints where the amount of minutiae can be over 900. The accuracy reached by this method in the performed experiments, is higher than 99,5 % for any value of penetration rate.

1 Introduction

Biometrics can be defined as the automated use of physiological or behavioural characteristics to identify or verify the identity of a person. Nowadays, palmprints matching is one of the most useful techniques [1,2]. The ridge patterns found on the palms of hands are unique, and they provide enough information to distinguish a specific person from the rest. Since the palm area is much larger, more distinctive features can be captured in comparison with fingerprints. This fact turns palmprints into a suitable technique for identification systems.

Despite having some similarities with fingerprint, the use of palmprints in recognition tasks brings additional problems. These issues are given mainly because the useful area of a palmprint is very large: palmprints usually have more than 900 minutiae. Thence, the computational cost of a matching operation has a significantly increment, since palmprints recognition process can not be implemented by searching over every possible entry stored in a gallery.

For these reasons, the reduction of the search space in which an exhaustive search will be performed is essential. Unlike fingerprints [3], palmprints can not be classified by any standard efficient criteria. Also, even when some proposals are found in the literature [4,5], the number of clusters in which the search

U. Garain and F. Shafait (Eds.): IWCF 2012 and 2014, LNCS 8915, pp. 10–19, 2015.
DOI: 10.1007/978-3-319-20125-2_2

space is divided is small. Therefore, the reduction of the potential candidates is minimal using this approach.

The classical solution for this kind of problem is the use of some indexing and retrieving process. There are many proposals focused on indexing approaches in fingerprint recognition [6–8]; however, in palmprints it is a relatively new topic. Moreover, Delaunay triangulations and other similar structures have never been used in palmprint indexing as in fingerprints identification [9–11].

There are a few proposals in the literature using different techniques for palmprints indexing. Some of these techniques are line features for building hierarchical classifiers [12], texture features [13,14], Haar wavelets and Zernike moments [15], palmprint pre-alignment and coarse matching [16], SIFT points [17] and KD-trees [18]. However, all of these works use images of a palm and they do not use minutiae information.

The main advantage of our proposal, is that, unlike other methods found in the state-of-the-art, it reaches a high accuracy using only minutiae information. These characteristic points are very used by experts in the field, so our method can be used even with manually edited features, in a more intuitive way.

The general structure of all indexing methods is very similar. When a new object (palmprint in our case) is inserted in the gallery, some information in form of indices is extracted from it and stored in an index table. Then, when a query is performed, their indices are obtained in the same way, in order to find correspondences with the index table. Using this information, the algorithm must be able to return a list of candidates ordered by a similarity value (see Fig. 3).

The present work is organized as follows. In Sect. 2, the representation used for processing palmprints is described. Section 3 is dedicated to the selection of the features used and the construction of the index table. In Sect. 4, the process of retrieving is presented and in Sect. 5 some experimental results validating our proposal are shown. Finally, in Sect. 6, our conclusions are given.

2 Representation

In this proposal, a previously introduced representation [8] in the literature is used, in order to assign a unique topological structure to each palmprint. This representation, based on the Delaunay triangulation, has been successfully used in fingerprints. Formally, a generalization of the Delaunay triangulation [19] can be defined as follows:

Definition 1. *Let $P = \{p_1, \cdots, p_n\}$ be a set of points in $\mathbb{R}^2$, for $p_i, p_j, p_k \in P$:*

- *$(\overline{p_i p_j})$ is a Delaunay edge of order k if there is a circumcircle to $(\overline{p_i p_j})$ that contains at most k point of P.*
- *$\triangle p_i p_j p_k$ is a Delaunay triangle of order k is its circumcircle contains at most k points of P.*
- *A triangulation of P is a Delaunay triangulation of order k if every triangle in the triangulation is from Delaunay of order k and will be denoted as $TD_k(P)$.*

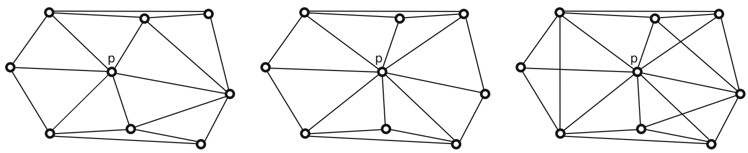
The Delaunay triangulation of order k is unique for a specific set of points if there is no circumcircle with three or more points of P on the border. This characteristic is very useful in many areas of image and shape recognition.

As was mentioned earlier, some proposals build representations based on the previous definition, by constructing the set P with minutiae coordinates [6,8]. However, Delaunay triangulations may suffer great local changes in some cases. This situation may occur even with small variations on the points coordinates. In the process of palmprint recognition, these changes may represent a serious issue. The human skin is elastic, so the position of some minutiae may be different in two impressions of the same palm. This problem is more evident in palmprints since the area of interest is much bigger than in fingerprints. In Fig. 1(a) and (b) we can see the structural change in a triangulation when a point p is slightly displaced.

In order to deal with this problem, an expanded triangle set $ET_k(P)$ [8], defined as follows is used:

$$ET_k(P) = TD_0(P) \bigcup TD_k(P) \tag{1}$$

As it is shown in Fig. 1(c), this representation generates the triangles that would be found if minutiae were displaced. In this way, some matches will be found even if there are variations in coordinates produced by noise.



(a) Delaunay triangulation (b) With a small displacement of p (c) Set of triangles $ET_k(P)$

Fig. 1. Triangles generated with the expanded representation.

The expanded triangle set has many advantages regarding other existing representations. Besides dealing with displacements of minutiae, the number of triangles generated in $ET_k(P)$ remains linear over the size of P, as is enunciated in the following property:

Theorem 1. *The number of triangles in* $ET_k(P)$ *is smaller than* $8|P| - 5|CH(P)| - 8$ *for any value of* k*, where* $|P|$ *and* $|CH(P)|$ *are the number of points of the set* P *and the number of edges in* $CH(P)$*, respectively (see proof in Muñoz-Briseño et al.* [8]*).*

This property is very desirable in palmprints recognition, since the number of minutiae found in a palm is considerably bigger than in a finger. Delaunay triangles of order k are obtained using a variation of the algorithm described by Gudmundsson et al. [19].

As in a previous proposal [8], a strategy was used in order to eliminate the triangles of $ET_k(P)$ that may be affected by noise in the impression. To perform this operation, bad quality zones of the palmprints are computed by using a state-of-the-art method [20].

Finally, those triangles belonging to $ET_k(P)$ with sides crossing more than r pixels of a bad quality zone in the palmprint, are eliminated. In Fig. 2 the final representation used in the present proposal can be seen.

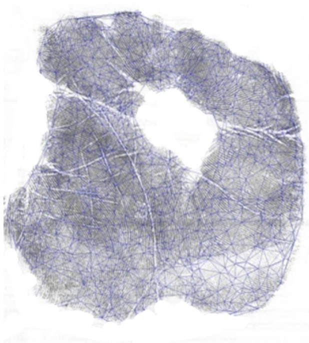


Fig. 2. Palmprint representation.

3 Indexing Stage

In this stage, the index table H that will be used in the retrieving stage is designed. The values of the keys introduced in H are indices that describe distinctive characteristics of the palmprints stored in the database.

In order to index a palmprint using its respective representation $ET_k(P)$, a feature vector $f(s_t, a_1, a_2, a_2, c_{ij}, c_{jk}, c_{ki})$ of each triangle $\triangle p_i p_j p_k \in ET_k(P)$ made up by the following features is extracted:

- s_t: triangle sign. $s_t = 0$ if the expression $x_i(y_j - y_k) + x_j(y_k - y_j) + x_k(y_i - y_j)$ is less than 0, and $s_t = 1$ otherwise.
- a_1, a_2 and a_3: relative directions of the minutiae represented by the points p_i, p_j y p_k, with respect to their opposite side in $\triangle p_i p_j p_k$, with $0 \leq a_i \leq 7$ since the value of this feature is discretized.
- c_{ij}, c_{jk} and c_{ki}: ridge counters between the segments $(\overline{p_i p_j})$, $(\overline{p_j p_k})$ and $(\overline{p_k p_i})$, respectively.

Using the minimum amount of bits for representing each component of f, the feature vector can be represented with 22 bits. Taking this into account the index function is defined as follow.

Definition 2. *The index function of a feature vector f is defined as $h : \Phi \rightarrow K_{22}$, such that $h(f(s_t, a_1, a_2, a_2, c_{ij}, c_{jk}, c_{ki}))$ is the integer number obtained by concatenating the binary representation of each component of f.*

With the previous definition the index table H is built. In Algorithm 1 this process is illustrated, given a set of tuples $E = \{\langle ET_{k1}(P_1), ID_1\rangle, \langle ET_{k2}(P_2), ID_2\rangle, \ldots, \langle ET_{kn}(P_n), ID_n\rangle\}$, where $ET_{ki}(P_i)$ is an extended triangle set, and ID_i is the identifier assigned to the palmprint represented by $ET_{ki}(P)$.

Algorithm 1: Indexing process.

```
Input: E
Output: H
H ← ∅
foreach ⟨ET_ki(P_i), ID_i⟩ ∈ E do
    foreach △p_l p_m p_n ∈ ET_ki(P_i) do
        f ← extractFeatureVector(△p_l p_m p_n)
        H ← H ∪ {h(f), ⟨△p_l p_m p_n, ID_i⟩}
    end
end
return H;
```

Firstly, an empty table H is initialized (line 1). Then, for each $\triangle p_l p_m p_n$ in each $\langle ET_{ki}(P_i), ID_i\rangle$ the respective feature vector f_i is obtained. Using this information, an entry of the form $\langle \triangle p_l p_m p_n, ID_i\rangle$ is inserted in H under the key $h(f_i)$ (lines 2–7). It is important to note that more than one entry can be stored in H under the same key.

4 Retrieving Stage

In the retrieving stage, the candidate list to match with a given palmprint (query) is constructed using a correspondence count structure called R. Each element of R is composed by a key and a numerical counter. The whole process is detailed in Algorithm 2.

The first step is to obtain the representation $ET_{kq}(P_q)$ of the query, using the same process explained in previous sections. Then, for each $\triangle p_l p_m p_n \in ET_{kq}$ the following process is performed. Let f be the feature vector computed from $\triangle p_l p_m p_n$, and let $L = \{\langle \triangle p_{l1} p_{m1} p_{n1}, ID_1\rangle, \langle \triangle p_{l2} p_{m2} p_{n2}, ID_2\rangle, \ldots, \langle \triangle p_{lr} p_{mr} p_{nr}, ID_r\rangle\}$ be the set of tuples of H stored under the index $h(f)$.

Using this matches, new tuples $\gamma_i = \langle ID_i, T_i\rangle$ are built where T_i is a geometric transformation performed on $\triangle p_l p_m p_n$ composed by the values of translation and rotation that minimize the average euclidean distance between the transformed points of $\triangle p_l p_m p_n$ and their corresponding points in $\triangle p_{li} p_{mi} p_{ni}$ (lines 5–6). Then, γ_i is used as key to insert a new entry in R. If an element with the same key already exists in R, the associated counter is incremented by 1. Otherwise, a new element is created in R with the counter set to 0, using γ_i as key (lines 7–11). Notice that if there are no entries stored with the key $h(f)$ in H, no operation is performed. In Fig. 3 the general scheme of this proposal is illustrated.

Finally, the candidate list is obtained from R by sorting their elements in decreasing order, by the values of the counters. This method is based on the idea that if indices are extracted from two different impressions of a same palm, they will match with very similar geometric transformations.

The retrieving process of our approach is very efficient since every operation in this stage can be performed by using some kind of index. Both structures, H and R, can be implemented using hashtables or similar data structures, so every operation over them has a constant order. In addition, the size of L is considerably smaller than the amount of elements contained in H in a real environment. In this way, the final number of operations depends on the amount of indices generated by the queries.

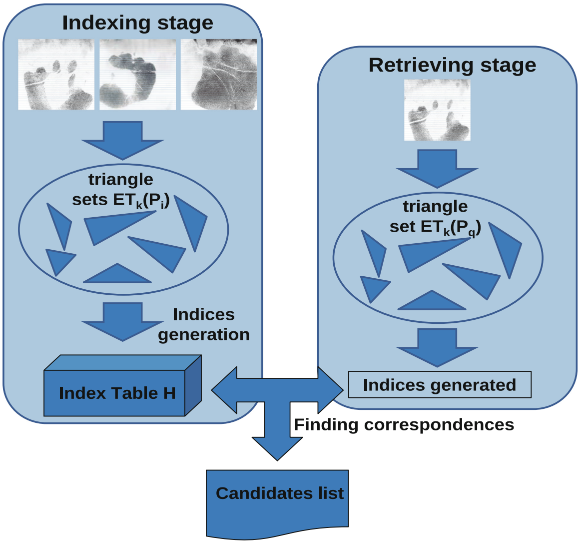


Fig. 3. Final scheme of the indexing and retrieving stages.

5 Experimental Results

In our research we could not find any free palmprint dataset in order to perform our experiments. The only dataset available was provided by the authors of a previous work [16]. That is why comparisons where performed with this proposal as reference. In order to test the accuracy of our proposal, experiments were performed using the first 100 palmprints of a dataset introduced by Yang et al. [16]. As in the mentioned work, the results reported were obtained by selecting one impression for each palmprint to create the gallery, while the other seven were used as queries (700 palmprints in total). As can be seen in Fig. 4, some images of this dataset are very rotated. Also, 12 616 single impressions were added as background, totalling 13 316 images in the gallery.

Algorithm 2: Retrieving process.

Input: $ET_{kq}(P_q)$
Output: R - count structure
foreach $\triangle p_l p_m p_n \in ET_{kq}$ **do**
  $f \leftarrow extractFeatureVector(\triangle p_l p_m p_n)$
  $L \leftarrow FindTuplesUnderIndex(h(f), H)$
  foreach $\langle \triangle p_{li} p_{mi} p_{ni}, ID_i \rangle \in L$ **do**
    $T_i \leftarrow ComputeTrans(\triangle p_l p_m p_n, \triangle p_{li} p_{mi} p_{ni})$
    $\gamma_i \leftarrow \langle ID_i, T_i \rangle$
    if $\gamma_i \in R$ **then**
      $IncrementCounter(R[\gamma_i])$
    end
    else
      $R \leftarrow R \cup \{\gamma_i, 0\}$
    end
  end
end
return R;

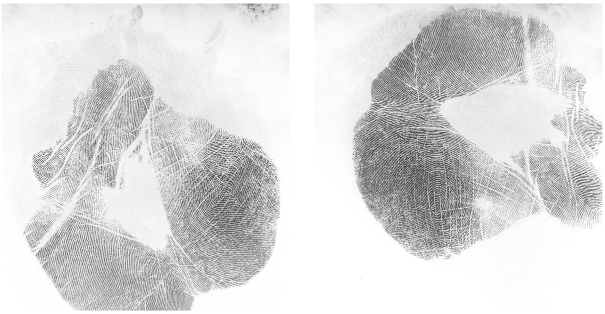

(a) (b)

Fig. 4. Two rotated impressions of the same palmprint contained in the test dataset.

To extract minutiae from the elements of the dataset a similar method to the one reported in literature was used [21]. This approach is based on the computation of black-white transition count around each point in the skeletonized image of palmprints. A very simple segmentation of the images was performed in order to eliminate blank areas. For the calculation of ridge counters a simple method that uses the binarized image was developed.

Table 1 illustrates the average amount of minutiae and triangles computed in the extraction stage with different values of k in our palmprints representation. As we can see, the proportion remains linear.

The classical measure used for the evaluation of indexing algorithms, is the trade-off between Penetration Rate (PR) and Correct Index Power (CIP). For this reason, the accuracy of our results is expressed in function of this. More formally we can define $CIP(N) = 100 \times c(N)/E$ and $PR(N) = 100 \times N/E$,

Table 1. Efficiency.

	Average number of minutiae	Average number of triangles
k = 0	912	1726
k = 1		2496
k = 5		4243
k = 10		4796

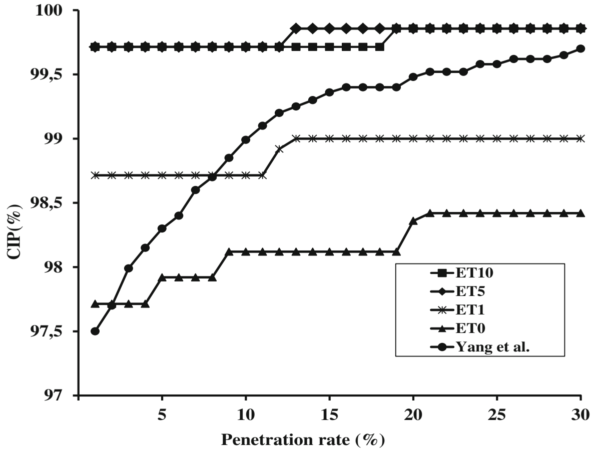


Fig. 5. Comparison of indexing palmprints algorithms.

where E is the number of experiments and $c(N)$ is the number of times that the correct result is within the list of the first N hypothesis.

In Fig. 5 the results obtained by our approach using representations with different values of k is shown. From these results, we can conclude that for $k \geq 5$ the accuracy is almost the same. Also, we can see that our proposal outperforms a method proposed by Yang et al. [16]. In fact, we return the correct impression in first place in the candidate lists of 698 queries. The two remaining queries are very degraded palmprints in which our method fails to collect proper features.

The introduced approach not only reaches a high accuracy but also presents a good efficiency compared with the proposal of Yang et al. [16], as evidenced by Table 2. Execution times in both processes, features extraction and retrieving are

Table 2. Average execution time (in seconds) of features extraction and retrieving processes for each palmprint.

	Proposed approach	Yang et al.
Features extraction	2.3 s	22 s
Retrieving	0.15 s	0.22 s

lower than the previously mentioned work. The same programming environment was used to compute the average execution time of both algorithms.

6 Conclusions

In the present work, a new palmprint indexing algorithm is proposed. One of the advantages of the introduced method is that it only works with minutia information in order to build indices. Minutiae are also the main features used by experts in manual matching. In this way, unlike other state-of-the-art algorithms, our proposal can be implemented in an environment of interaction with humans in the features extraction stage. Moreover, the used palmprint representation is able to deal with very rotated and distorted impressions without losing accuracy. The use of the expanded triangle set also has the advantage that the amount of indices keeps being linear regarding the number of minutia found.

Experimental results show that our proposal achieves high accuracy and good efficiency in a dataset composed by very rotated impressions, outperforming other state-of-the-art proposals. As future works, we will test our method in palmprints with other kind of degradation or noise. In addition, some other triangle features will be evaluated in order to increase the final accuracy.

References

1. Zhang, D., Zuo, W., Yue, F.: A comparative study of palmprint recognition algorithms. ACM Comput. Surv. **44**(1), 2 (2012)
2. Somvanshi, P., Rane, M.: Survey of palmprint recognition. Int. J. Sci. Eng. Res. **3**, 1–7 (2012)
3. Liu, M.: Fingerprint classification based on adaboost learning from singularity features. Pattern Recogn. **43**(3), 1062–1070 (2010)
4. Fang, L., Leung, M.K.H., Shikhare, T., Chan, V., Choon, K.F.: Palmprint classification. In: SMC, pp. 2965–2969 (2006)
5. Wu, X., Zhang, D., Wang, K., Huang, B.: Palmprint classification using principal lines. Pattern Recogn. **37**(10), 1987–1998 (2004)
6. Liang, X., Bishnu, A., Asano, T.: A robust fingerprint indexing scheme using minutia neighborhood structure and low-order delaunay triangles. IEEE Trans. Inf. Forensics Secur. **2**(4), 721–733 (2007)
7. Cappelli, R., Ferrara, M., Maltoni, D.: Fingerprint indexing based on minutia cylinder-code. IEEE Trans. Pattern Anal. Mach. Intell. **33**(5), 1051–1057 (2011)
8. Muñoz-Briseño, A., Gago-Alonso, A., Hernández-Palancar, J.: Fingerprint indexing with bad quality areas. Expert Syst. Appl. **40**(5), 1839–1846 (2013)
9. Bebis, G., Deaconu, T., Georgiopoulos, M.: Fingerprint identification using delaunay triangulation. In: IEEE International Conference on Intelligence, Information, and Systems, ICIIS, pp. 452–459 (1999)
10. Ross, A., Mukherjee, R.: Augmenting ridge curves with minutiae triplets for fingerprint indexing. In: Proceedings of SPIE Conference on Biometric Technology for Human Identification IV, vol. 6539 (2007)
11. Iloanusi, O.N.: Fusion of finger types for fingerprint indexing using minutiae quadruplets. Pattern Recogn. Lett. **38**, 8–14 (2014)

12. Li, F., Leung, M.K.H.: Hierarchical identification of palmprint using line-based Hough transform. In: ICPR, vol. 4, pp. 149–152 (2006)
13. Paliwal, A., Jayaraman, U., Gupta, P.: A score based indexing scheme for palmprint databases. In: ICIP, pp. 2377–2380 (2010)
14. Li, W., You, J., Zhang, D.: Texture-based palmprint retrieval using a layered search scheme for personal identification. IEEE Trans. Multimedia **7**(5), 891–898 (2005)
15. Nagasundara, K.B., Guru, D.S.: Article: Multi-algorithm based palmprint indexing. In: IJCA Proceedings on International Conference and Workshop on Emerging Trends in Technology (ICWET 2012) vol. 2, pp. 7–12 Published by Foundation of Computer Science, New York, USA, March 2012
16. Yang, X., Feng, J., Zhou, J.: Palmprint indexing based on ridge features. In: IJCB, pp. 1–8 (2011)
17. Badrinath, G.S., Gupta, P., Mehrotra, H.: Score level fusion of voting strategy of geometric hashing and SURF for an efficient palmprint-based identification. J. Real Time Image Process. **8**(3), 265–284 (2013)
18. Anitha, M.L., Rao, K.A.R.: An efficient palmprint identification system based on an indexing approach. In: IEEE ICACCI, pp. 688–693 (2013)
19. Gudmundsson, J., Hammar, M., van Kreveld, M.J.: Higher order delaunay triangulations. Comput. Geom. **23**(1), 85–98 (2002)
20. Maltoni, D., Maio, D., Jain, A.K., Prabhakar, S.: Handbook of Fingerprint Recognition, 2nd edn. Springer Publishing Company Incorporated, London (2009)
21. Kasaei, S., Boashash, B.: Fingerprint feature extraction using block-direction on reconstructed images. In: IEEE Region TEN Conference on Digital Signal Processing Applications, TENCON, pp. 303–306 (1997)

http://www.springer.com/978-3-319-20124-5

Computational Forensics
5th International Workshop, IWCF 2012, Tsukuba, Japan, November 11, 2012 and 6th International Workshop, IWCF 2014, Stockholm, Sweden, August 24, 2014, Revised Selected Papers
Garain, U.; Shafait, F. (Eds.)
2015, X, 213 p. 104 illus., Softcover
ISBN: 978-3-319-20124-5