



PALMPRINT CLASSIFICATION USING A FIXED NUMBER OF KEYPOINTS

ANCA IGNAT¹, IOAN PĂVĂLOI²

Keywords: Scale-invariant feature transform keypoints; Keypoint matching; Palmprint classification.

In this article we use, for palmprint feature extraction, descriptors generated with SIFT (Scale-invariant feature transform) algorithm. The main idea was to generate for each image in the dataset, the same number of keypoints. We deduced an algorithm that, for a given image, computes a fixed number of SIFT keypoints. The matching procedure is based on the nearest neighbor ratio equation. To test the efficacy of our method, we performed experiments on five well-known palmprint databases. The experimental results indicate that this type of approach yields very good classification results. Our results are better than those obtained in some recent papers.

1. INTRODUCTION

In our times, modern technology is present almost everywhere in day-to-day life. Many of these applications require an authentication process. One of the easiest to use (especially on mobile phones) and most reliable biometric characteristics is the palmprint information.

The methods deduced for solving the palmprint recognition problem were adapted depending on the type of image available. Images of palms were acquired using various categories of devices. These devices produced contactless or contact-based images, high resolution or low-resolution images, 2D or 3D images, or images acquired with multiple types of tools.

For the contactless images, the databases with palm prints that are used in experiments can be classified into three categories. In the constrained category, which is the most numerous, the background is uniform, and the hand has a fixed position and orientation for all images. In the semi-constrained group of datasets, the background is non-uniform, or the hand position and orientation are variable, but not both at the same time. Mobile phone cameras usually produce images in this category. In the last category, the unconstrained ones, the background, and the hand pose do not respect any restrictive rules.

To test our method, we employed five well-known datasets, namely: CASIA (Institute of Automation, Chinese Academy of Sciences) [1], CASIA Multi-Spectral [2,3], GPDS (Las Palmas de Gran Canaria University) [4], IITD (Indian Institute of Technology in Dehli) [5] and PolyU (The Hong Kong Polytechnic University) [6]. They contain constrained and semi-constrained images.

The quality of the proposed palmprint recognition methods is commonly evaluated using measures such as EER (Equal Error Rate), AUC (area under the ROC curve), (average) accuracy, and rank 1.

For solving the palmprint recognition problem different approaches were tested by using directional features [7,8], texture features [9], or combinations of these two [10]. In recent research, discriminative features were deduced with a learning process [11–13], and a cross-dataset study [14]. Deep Learning methods were also employed for palmprint recognition [11,14].

In [15–18], one can find detailed information on the methods developed for solving the palmprint recognition

problem.

Keypoint characterization for palmprint images was employed in a few papers. In [19] the authors present a method with three steps, a preprocessing one, the SIFT feature extraction step, and a refinement of the matched keypoints using RANSAC and local palmprint descriptors. In [20] a biometric system that uses both the hand shape features and SIFT extracted palmprint features is proposed. A features fusion process is employed and SVM is used for classification. The SIFT method is applied [21] only after parts of the palmprint with no line information were eliminated. The matching procedure was improved in two ways, by matching keypoints that have only small orientation differences and by eliminating the false matches. In [22] the SIFT keypoints matching process is improved by using the geometric relations between the compared keypoints, thus eliminating false positive matches. In [23] the authors analyze five feature extraction methods (SIFT and SURF included). In a preprocessing step, histogram equalization is performed. SVM and k -NN are used for classification. The results obtained with SIFT are not very encouraging.

The algorithms that use keypoints for image characterization perform a keypoint-matching step for comparing two images. If the same set of parameters for the SIFT procedure [24,25] is employed, different numbers of keypoints are generated for different images. In this paper, we are using a method that computes a fixed number of keypoints for an image. The method we developed is named a fixed number of keypoints (FINUK) and iteratively adapts the contrast threshold SIFT parameter to compute for an image a given number of key points. We show that generating the same number of keypoints for each image involved in the classification procedure leads to very good recognition results, on all the employed datasets.

Our present paper has six sections. Section 2 is dedicated to the five datasets and variants that we are using in the experimental tests. In Section 3 we present the computation of the SIFT keypoint descriptors and the keypoint matching process. In Section 4 we detail the FINUK algorithm that allows us to compute for an image a fixed number of keypoints (when possible). Section 5 is dedicated to the results of our experiments. We end with the section dedicated to conclusions and future directions of research.

¹ University “Alexandru Ioan Cuza” of Iași, Romania, Faculty of Computer Science, str. Berthelot 16, 700483, ancai@info.uaic.ro

² Institute of Computer Science, Romanian Academy, Iași Branch, str. T. Codrescu, nr. 2, 700481, Iași, Romania, ioan.pavaloi@iit.academiaromana-is.ro

2. DATABASES

We tested our method using five constraint palmprint datasets: CASIA, CASIA Multi-Spectral, GPDS, IITD, and PolyU.

The CASIA dataset (denoted CASIA in this paper) has 5502 .jpg images, each one of 640×480 pixels. There are grayscale palmprint images for several 312 persons. For each person, for each hand, there are a minimum of 8 and a maximum of 11 images.

CASIA Multi-Spectral Palmprint image database (denoted CASIA-MS in this paper) contains images from 100 different people. The images are captured using self-designed multiple spectral imaging devices. Each palm image is 8 bit gray-level JPEG file. For each hand, the palm images are captured in two different sessions, the time interval between them being larger than one month. In each session six palm images are acquired, three for each hand. These are captured using six different electromagnetic spectrums at the same time; between two samples being allowed a variation in hand posture.

The GPDS dataset has images for 100 persons, ten for each one, so it consists of 1000 images. All the images are color images for the right palm. The images are in .bmp format. The size of each image in the dataset is 1600×1200. In the acquisition process, there was no rule for the hand's pose. In our experiments, we used three versions of this dataset: the original dataset, denoted GPDS-F, the second one that contains segmented grayscale images (401×401 pixels), denoted GPDS-S, and a third dataset that contain also segmented images but of size 128 × 128 pixels, named GPDS-S-S.

The IITD palmprint database consists of 2601 palmprint images, 1300 for the right hand and 1301 for the left hand. It was captured from 230 people, 14 and 15 years old. There are five or six images for each left and right palm. The images are in .jpg format, 1600x1200 pixels. So, there are a total of 460 palms in the IITD dataset. The original dataset (denoted by IITD) was automatically segmented and normalized (denoted as IITD-S in this paper). We used in our experiments the original dataset and the segmented images, BMP grayscale images of dimensions 150×150 pixels.

The PolyU dataset contains 7752 images, for left and right hands, from 193 subjects. So, there are a total of 386 palm images, each with several 11 to 27 samples. The images are in .bmp format, dimension 384 × 284 pixels and are acquired in two different sessions with about 60 days between the two sessions. In Figure 1 are one sample from each dataset.

3. SIFT FEATURE EXTRACTION AND KEYPOINT MATCHING

For keypoint generation, we use SIFT algorithm developed by Lowe in [24,25]. For an image, SIFT algorithm computes in four steps, a variable number of keypoints. The number of computed keypoints depends on the content of the image. Then, one associates with each such a point a feature vector with 128 elements.

Using the SIFT features, a test image classification is made depending on the number of matching keypoints between the test image and the training images.

The matching keypoints procedure between two images is described below. We first compute SIFT descriptors (keypoints) for each image of the dataset. Let's consider

two images, U and V with m and respectively n descriptors, denoted u_1, u_2, \dots, u_m and v_1, v_2, \dots, v_n .

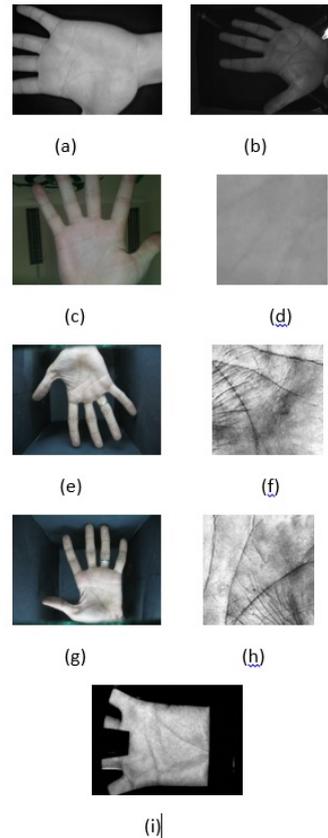


Fig. 1 – Example of images from datasets: (a) - CASIA, (b) - CASIA Multi-Spectral, (c) - GPDS, (d) - GPDS_S, (e) - IITD Left, (f) - IITD Left segmented, (g) - IITD Right, (h) - IITD Right segmented, (i) -PolyU

Matching the keypoints of the two images is computed in the following way:

1. Calculate the distances between each feature vector associated with image U and each feature vector associated with image V ; $m \times n$ distances are computed.
2. The keypoint with feature vector u_i from image U matches the keypoint v_k of the V image if the following condition is fulfilled:

$$\text{dist}(u_i, v_k) \leq T \text{dist}(u_i, v_j), j \neq k \quad (1)$$

The parameter T is a real positive value, less than 1. This threshold parameter lets us control the keypoints matching procedure. It is very important to notice that a keypoint from image U can match only one keypoint from image V . More details about the keypoint extraction and the matching process can be found in [26].

The classification process is performed as follows. For a test image one computes the matching keypoints with the images in the training set. The image from the training set that has a maximal number of matched keypoints with the test image provides the label. If there is more than one image in the training set with this property, one chooses the image at a minimum distance from the test image. This distance is computed as the average of distances between the locations of the matched keypoints.

In the experiments, we tested three distances in the first step, Manhattan, Euclidean, and Canberra. All results presented in this paper are for the Manhattan distance, which provided better results in faster computing time.

In our first experiments we used the default values in the parameters of SIFT algorithm. Then we studied the influence on the classification results of the number of octaves in each layer, the contrast related threshold and the edge related threshold. We deduced that the most significant influence is provided by the contrast related threshold, cT . So, in all our computations we employed pre-established, default values for all parameters except this one.

If we use a pre-established, fixed set of SIFT parameters, there exist images with no keypoints associated, images with very few key points and other with a big number of keypoints are computed. This situation has the following inconveniences:

- if an image has no keypoints it cannot be classified;
- images with a small number of associated keypoints are often misclassified;
- it seems not to be reasonable to perform a matching procedure between an image with very few keypoints and an image with a large number of keypoints.

The same problem was encountered when applying keypoint characterization on occluded iris images [27].

We have to note that the number of computed keypoints for any two images we intend to compare, influences the computing time in the matching algorithm. So, the consuming classification time for images with large number of keypoints is big because the matching procedure is computationally expensive.

Computing manually the set of SIFT parameters for each image in the dataset is not an option.

We decided to address this issue from a different point of view. We propose an iterative method that computes a given number of keypoints (or a number in a narrow interval), by finding the appropriate value for the cT parameter. In the next section we'll detail the proposed method, named a fixed number of keypoints (FINUK).

4. FINUK METHOD FOR KEYPOINT GENERATION

The FINUK method iteratively generates a fixed number of SIFT descriptors by adjusting the value of the contrast threshold parameter cT , the other parameters remaining unchanged. This method is applied on each image.

One chose three natural numbers, K , I_{max} and Δ_K , where K is the desired number of descriptors that one wants to compute, for an image, I_{max} is the maximal number of iterations that FINUK performs. I_{max} is a safety parameter, introduced to avoid endless computations. If after I_{max} iterations the algorithm does not compute the desired value for cT , the algorithm stops returning the contrast threshold value that computes several SIFT descriptors closest to K . In the successful case, the algorithm stops if the computed cT value generates several descriptors the interval $[K-\Delta_K, K+\Delta_K]$. For $\Delta_K = 0$, the algorithm either allows computing exactly K descriptors or it stops after I_{max} steps. In our computations, we used the following assignments $I_{max} = 49$ and $\Delta_K = 1$.

The algorithm computes the cT value as follows. One chooses an initial value for cT (we used $cT = 0.04$ as a default starting value). The idea behind the algorithm is simple. One searches two values for the cT parameter, cT_1 and cT_2 , such that the first one obtains k_1 descriptors with $k_1 < K-\Delta_K$, and for cT_2 , k_2 descriptors with $k_2 > K + \Delta_K$. Then one generates the set of descriptors for the cT value that verifies the equation:

$$\frac{(cT - cT_1)}{(cT_2 - cT_1)} = \frac{(K - k_1)}{(k_2 - k_1)} \quad (2)$$

If the number of computed descriptors satisfies the desired condition, the algorithm stops. Otherwise, computations continue in the following way:

- if we get fewer descriptors than $K-\Delta_K$, one continues the computations using the interval (cT, cT_2) ;
- If we get more descriptors than $K+\Delta_K$, one continues the computations using the interval (cT_1, cT) .

When computing the initial values for cT_1 and cT_2 one considers that when one computes keypoints with decreasing values for the cT the number of keypoints increases and when the value of cT is successively increased one obtains fewer and fewer descriptors. Let k be the number of descriptors obtained for the default value $cT=0.04$:

- if k is in $[K-\Delta_K, K+\Delta_K]$, the algorithm stops;
- if $k > K + \Delta_K$ then $cT_2 = cT$; the desired value for cT_1 is found by successively dividing by 2 the cT value, $cT = cT/2$, until one computes less then $K-\Delta_K$ descriptors; this process stops if the maximal number of iterations is achieved;
- if $k < K-\Delta_K$ then $cT_1 = cT$; the desired value for cT_2 is computed by successively multiplying by 2 the current cT value, $cT = 2*cT$, until one computes more than $K+\Delta_K$ descriptors; this process stops if the maximal number of iterations is achieved.

We use the FINUK method in the recognition process in the following way. First, we compute, for a given value of the cT parameter, the average number of keypoints for the images in the training set. Using FINUK, we compute for each image a value cT that will approximately generate this average number of keypoints. This type of computations are denoted FINUK(cT).

5. RESULTS

We used two types of tests. One is a Leave-One-Out (LOO) approach: we consider each image as a test image and the other formed the training set. In the second one we placed in the training set q images per person with $q = 1, 2, 3$, and 4, the remaining images were placed in the test set. For each q , we considered 10 selections of training sets. In this case the results are presented as average accuracy and standard deviation. In our experiments we used for the threshold parameter T from eq. (1) values from 0.1 to 0.9, with a 0.1 step. Because the best results were obtained for $T = 0.6$ and 0.7 (especially for 0.6), we choose to present in this paper the experimental results obtained for these values of the threshold parameter. We used the SIFT method implemented in OpenCV [28].

We first computed some statistical values: the total number of keypoints for all the images in the dataset, the minimal, the maximal, and the average numbers of keypoints, the max/min ratio, the difference max-min and standard deviations of keypoint numbers value. For the total number of keypoints for a dataset, we present approximate values that we obtained by dividing the exact values by 10^3 . In Table 1 are these statistical values for IITD-F with SIFT contrast threshold $cT \in \{0.03, 0.04, 0.05, 0.07, 0.08, 0.10\}$.

For $cT = 0.10$ there is an image for which only 4 descriptors were computed while there are 18 images with more than 40 descriptors associated (10 times more). For $cT = 0.04$ there are 462 images that have over 100

descriptors (88 among them have over 150 keypoints and 16 have over 200 keypoints) and 364 images with at most 50 descriptors. For $cT = 0.03$, SIFT computes at least 250 descriptors for 233 images and 722 images have less than 100 descriptors (15 among them have at most 50 descriptors). The difference between the maximum and minimum number of descriptors varies from 51 (for $cT = 0.10$) to 568 ($cT = 0.03$). As can be seen in Table 1, the ratio between the maximal and minimal number ranges from 6.60 (for $cT = 0.08$) to 18.75 (for $cT = 0.03$). One also notes that for decreasing values of cT one obtains increasing values for the standard deviation.

Table 1
Statistics for IITD-F collection

Statistics	cT values					
	0.03	0.04	0.05	0.07	0.08	0.10
total	388	203	138	88.9	77.3	62.9
min	32	21	16	11	10	4
max	600	266	146	77	66	55
avg	148.8	78.2	53	34.2	29.7	24.19
max/min	18.75	12.67	9.13	7.00	6.60	13.75
max-min	568	245	130	66	56	51
std. dev.	73.9	31.0	16.6	8.06	6.68	5.50

These big differences between the numbers of descriptors generated for the different images are not specific to the IITD dataset only. The same phenomenon is present in the other palmprint datasets. For the contrast threshold $cT = 0.04$, we generated the SIFT descriptors for all the images in IITD, CASIA, CASIA-MS, and PolyU and then computed the same statistical values as in Table 1. The obtained values are in Table 2.

Table 2
Statistics for $cT = 0.04$

	IITD-F	GPDS-F	CASIA	CASIA-MS	PolyU
No elem.	2601	1000	5502	7200	7752
No descr.	203.2	183.8	169.6	569.6	1335.2
Min	21	5	7	23	55
Max	266	2442	428	654	511
Avg.	78.2	183.81	30.82	79.11	172.24
max/min	12.67	488.4	61.14	28.43	9.29
max-min	245	2437	421	631	456
std dev.	31.04	243.41	25.12	29.95	60.62

For the GPDS-F dataset, there is an average of 184 descriptors per image, 15 images have over 1000 descriptors, and one of them has 2442 keypoints. On the other side, 60 images have less than 20 descriptors, among them, there are 4 images with less than 10 descriptors. 260 images have at most 50 descriptors while 92 images have more than 500 descriptors, and 80 have over 800 descriptors. The big differences between the minimal number and the maximal number of generated descriptors were observed also for CASIA, CASIA-MS, and PolyU datasets. Note that the standard deviation has very big values, especially for GPDS-F. From our tests, we found out that images with a small number of descriptors usually are misclassified. With these two tables, containing statistical values, we want to point out the discrepancies between the images in the datasets in what concern the number of computed keypoints.

It is extremely difficult to choose from the start, a value for the contrast threshold parameter that will be suitable for all the images in the datasets. For $cT = 0.04$ we have an average of 31 descriptors for CASIA, 184 for GPDS-F, and 172 for PolyU.

From Table 2, we have $cT = 0.04$, for the images in IITD, an average of 79 descriptors per image. A natural problem that we addressed was the following: if we generate for all the images in IITD the same number of descriptors, let's say 79, do we obtain better recognition? We intend to use the FINUK algorithm to study this problem.

In Table 3 are the results (in percentages) for the IITD dataset, using the LOO type of test. The parameter cT takes the following values $\{0.04, 0.05, 0.06, 0.08, 0.10\}$, and the matching threshold T in (1) has two values, $T=0.6$ and $T=0.7$, respectively. In this table are the recognition results when no calibration of the number of keypoints was performed and when FINUK method was employed.

We observe from the results in Table 3 that applying FINUK, although the keypoints computing time increases, the recognition results are superior. Thus, for $cT = 0.04$ and $T = 0.6$ using FINUK leads to an increase in the recognition results from 98.81 % to 99.27 %, and for $T = 0.7$ from 96.35 % to 98.82 %. For $cT = 0.08$ the difference is even bigger, 89.58% to 93.00% for $T = 0.6$ (almost 3.5 %) and for $T = 0.7$ from 77.97 % to 86.70 % (almost 9 %).

Improving the recognition results was observed for the other datasets too. Thus, for CASIA, with $cT = 0.04$, one obtains using the standard approach (with no parameter tuning) a result of 94.33 %. Using FINUK(cT), we obtained 98.58 % for $T = 0.6$. For $T = 0.7$, the difference in results is from 83.33 % (standard computations) to 97.33% (FINUK).

We noted that when the values of the contrast threshold parameter decrease, SIFT will generate an increasing number of descriptors. This fact leads to better recognition results even when no FINUK is applied. It should be noted, however, that the classification time is proportional to the number of descriptors. This obviously led us to a new problem: is it possible to obtain good recognition results by using FINUK method with small numbers of descriptors for each image? This would implicitly lead to superior performance at a shorter processing time.

Table 3

Recognition results for IITD with different values of the parameters

cT	0.04	0.05	0.06	0.08	0.10
$T=0.6$	98.81	96.89	95.39	89.58	82.93
$T=0.7$	96.35	92.81	88.74	77.97	65.28
FINUK(cT), $T=0.6$	99.27	98.08	96.50	93.00	89.16
FINUK(cT) $T=0.7$	98.92	97.15	94.58	86.70	77.85

In Table 4 are the results (in percentages) obtained when FINUK method was applied to generate 10, 20, 30, 40 and 50 descriptors for each image in the dataset, for all the datasets.

Note the results over 99% obtained for CASIA and CASIA-MS when using 40 and 50 descriptors per image. For IITD datasets, the original one and the segmented one, the results obtained using 30 descriptors are similar (the difference is under 1%). For the GPDS things change, the smallest segmented variant, GPDS-S-S, provided the best results. One possible explanation for these big differences in recognition results for the three variants of GPDS is the non-uniform background in GPDS. A preprocessing of the background could lead to improving the recognition results.

Table 4

Recognition results for all datasets

Datasets	Number of generated descriptors				
	10	20	30	40	50
IITD	60.05	85.93	93.00	96.35	98.00

IITD-S	80.93	92.12	93.96	96.12	97.35
GPDS	64.60	81.70	85.20	87.40	89.30
GPDS-S	77.40	86.70	90.60	92.60	93.30
GPDS-S-S	95.10	97.70	98.40	98.50	98.90
CASIA	60.81	92.77	98.58	99.38	99.56
CASIA-MS	92.19	98.61	99.83	99.94	99.96
PolyU	25.76	72.38	91.42	95.98	97.91

Obviously, increasing the number of descriptors/images leads to obtaining better results. For example, for IITD-S dataset, when we generated 100 keypoints per image, we obtained a recognition rate of 98.62%. For IITD, with 78 descriptors per image, we obtained a result of 99.27%. For GPDS-S-S, with 75 keypoints the recognition rate is 99.60%. Note that the recognition results in the LOO scenario are very good for all datasets.

In Tables 5 and 6 we present (average) recognition results (in percentages) when using the second scenario, i.e., we place in the training set q images for each person, the rest of the images are for testing. Because for each n , we chose 10 random selections of training sets, the results are of the form average recognition rates \pm standard deviation. The results in Table 5 are for CASIA dataset, for which we generated 10, 20, 30, 40, or 50 descriptors/images. In Table 6 we performed the same type of computations for the GPDS-S-S dataset.

The recognition results obtained by using only 1, 2, 3, or 4 images/person in the training set are very similar to those obtained using the LOO framework.

Table 5

Recognition results for CASIA, $q = 1, 2, 3, 4$ images/person in training set

q	Number of generated descriptors				
	10	20	30	40	50
1	60.86 \pm 0.61	92.76 \pm 0.38	98.58 \pm 0.16	99.38 \pm 0.10	99.56 \pm 0.07
2	61.01 \pm 1.06	92.79 \pm 0.64	98.58 \pm 0.26	99.39 \pm 0.18	99.56 \pm 0.12
3	61.38 \pm 1.21	92.96 \pm 0.79	98.62 \pm 0.27	99.42 \pm 0.20	99.58 \pm 0.13
4	60.94 \pm 1.34	92.64 \pm 0.85	98.51 \pm 0.35	99.34 \pm 0.25	99.52 \pm 0.20

Table 6

Recognition results for all GPDS-S-S, for $q = 1, 2, 3, 4$ images/person in training set

q	Number of generated descriptors				
	10	20	30	40	50
1	95.10 \pm 0.48	97.70 \pm 0.34	98.40 \pm 0.22	98.50 \pm 0.22	98.90 \pm 0.18
2	95.06 \pm 0.84	97.68 \pm 0.69	98.39 \pm 0.44	98.49 \pm 0.41	98.90 \pm 0.35
3	94.36 \pm 0.46	97.19 \pm 0.32	98.14 \pm 0.30	98.27 \pm 0.27	98.77 \pm 0.30
4	94.87 \pm 1.18	97.52 \pm 0.85	98.25 \pm 0.64	98.37 \pm 0.63	98.80 \pm 0.51

Note the very interesting fact that we obtain similar results regardless of the value of q (which influences the size of training and test sets). For CASIA, the difference between the best and the worst result from Table 5 when generating 50 descriptors/image is only 0.06 % (0.06 % represents only 3 images from this dataset), and for GPDS-S-S this difference is 0.13 % (that represents just one image). These results suggest that this method is appropriate to be used when in the training set are very few images for each person.

We studied another aspect of this method. Is it possible to use it as a preliminary, selection step that computes a reduced subset of images like the test one? Our idea is to use FINUK with a small number of keypoints and select top p images from the training set, similar with the test image. On this top p subset, we intend to perform the final classification step, this time using a finer method for image characterization. One of the options is to use on top p also SIFT with FINUK, but this time using a considerably larger

number of descriptors. In this paper, we studied if in the first 100 most similar images generated with SIFT-FINUK method one can find images similar to the test image. We applied FINUK to generate for each image 10, 20, 30, 40, and respectively 50 keypoints. In Table 7 are the results expressed as the percent of images that have no similar image in the first 100 images generated by SIFT-FINUK algorithm.

Table 7

Retrieval results for all datasets – percent of images that do not have similar images in the top 100

Datasets	Number of generated descriptors				
	10	20	30	40	50
IITD	1.58	0.35	0.23	0.15	0.12
IITD-S	4.69	1.46	1.00	0.69	0.58
GPDS	1.30	1.00	0.10	0.10	0.20
GPDS-S	6.00	3.50	2.30	1.40	1.00
GPDS-S-S	0.50	0.20	0.40	0.20	0.20
CASIA	1.42	0.24	0.05	0.05	0.04
CASIA-MS	0.36	0.44	0.00	0.00	0.00
PolyU	10.69	0.92	0.12	0.03	0.01

Note that for IITD with 50 descriptors/image only 3 images have no similar image in the first 100 selected images (when using 80 descriptors this number is reduced to zero).

One obtains very good results for all the datasets, the best one is for CASIA-MS (which is natural because this is the largest dataset consequently the training set is also large). For CASIA and GPDS only 2 images have no similar image in the first 100 selected using 50 descriptors/image. For IITD this number is 3 and for PolyU just one image is not retrieved in the first 100 selected with SIFT-FINUK method.

We compared the recognition results obtained by applying SIFT-FINUK method described in this paper on CASIA and GPDS-S-S (results computed by generating 50 descriptors/image) with the results reported on these two datasets in [7,12]. The ALDC-M method reported in [7] and SDDL M from [12], are one of the best results on these two datasets. The results are in Table 8 for CASIA and Table 9 for GPDS-S-S and are for the scenario when n images/person were placed in the training set. The results are in the same form as those in Table 5 and Table 6.

Table 8

Comparative results for CASIA

	$q = 1$	$q = 2$	$q = 3$	$q = 4$
FINUK	99.56 \pm 0.07	99.56 \pm 0.12	99.58 \pm 0.13	99.52 \pm 0.20
ALDC-M	86.16 \pm 1.03	92.03 \pm 0.97	93.65 \pm 2.18	94.64 \pm 1.35
SDDL M	85.63 \pm 1.03	97.01 \pm 0.13	97.70 \pm 0.68	98.87 \pm 0.39

Table 9

Comparative results for GPDS-S-S

	$q = 1$	$q = 2$	$q = 3$	$q = 4$
FINUK	98.90 \pm 0.18	98.90 \pm 0.35	98.77 \pm 0.30	98.80 \pm 0.51
ALDC-M	85.53 \pm 1.82	92.85 \pm 1.09	95.06 \pm 0.93	97.70 \pm 0.52
SDDL M	85.75 \pm 1.14	89.66 \pm 0.32	96.5 \pm 0.28	99.5 \pm 1.01

The method proposed in this article produces very good results. These results are similar regardless of the number of images for each person that are placed in the training set. From Table 8 and Table 9 we can see that we have very good recognition results when we only have one image per person in the training set. The results in this situation are much better than those provided by using the methods described in [7] and [12]. A further improvement for

CASIA dataset can be obtained by segmenting the palms. We intend to adapt the threshold methods described in [29] for palmprint segmentation.

It should be noted that our method is very stable; the dispersion is smaller in almost all cases for the FINUK method. The same good results are obtained for the other datasets used in our tests, *i.e.*, the results obtained using only one image per person in the training set are superior to the others obtained with the ALDC_M and SDDLMM methods. Thus, for IITD, with 50 descriptors generated for each image and one test image for each person an average classification percentage of 98.10 % is obtained, compared with 85.07 % for ALDC_M and 85.11 % for SDDLMM.

Finally, a few remarks can be made regarding the computing time required by FINUK method. The operation of generating descriptors is time-consuming. However, for a given dataset this operation is performed only once. It should also be noted that for extremely large collections it is possible to generate descriptors on several computers in parallel.

6. CONCLUSIONS

We presented in this paper a method for the palmprint recognition problem, that uses fixed numbers of SIFT keypoints that are computed. For generating a fixed number of keypoints we developed FINUK method, which adapts the contrast threshold parameter of SIFT algorithm. We show that using this method the palmprint recognition results are significantly improved. We also show that this method can be used to select a subset of candidates for a test image. On this subset, one can further apply a finer characterization method, thus obtaining better recognition results. This type of approach will be addressed in a future paper. We study the recognition rates on palmprint datasets with full hand images and on variants that contain segmented palmprints.

We have presented results obtained for five palmprint datasets and we show that the results are very good, in some situations are even better than those obtained in some recent papers. The computational time for large datasets is big, but, as we have already stated above, these computations are performed only once, and the generation process can be done using parallel computations.

We intended to study the proposed FINUK method behavior on images with missing information and on noisy palmprint images.

Received on 12 January 2022

REFERENCES

- Z. Sun, T. Tan, Y. Wang, S.Z. Li, *Ordinal palmprint representation for personal identification*, Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR'05), **1**, pp. 279-284 (2005).
- Y. Hao, Z. Sun, T. Tan, C. Ren, *Multispectral palm image fusion for accurate contact-free palmprint recognition*, 15th IEEE International Conference on Image Processing, San Diego, California, USA pp. 281-284 (October 2008).
- Y. Hao, Z. Sun, T. Tan, *Comparative studies on multispectral palm image fusion for biometrics*, Asian Conference on Computer Vision, Tokyo, Japan, pp. 12-21 (November 2007).
- Grupo de Procesado Digital de Señales (GPDS) Palmprint image database: <http://www.gpds.ulpgc.es>.
- Indian Institute of Technology Delhi Palmprint Database (version 1): https://www4.comp.polyu.edu.hk/~csajaykr/IITD/Database_Palm.htm.
- The Hong Kong Polytechnic University Palmprint Database (version 2.0): <http://www.comp.polyu.edu.hk/~biometrics>.
- L. Fei, B. Zhang, W. Zhang, S. Teng, *Local apparent and latent direction extraction for palmprint recognition*, Information Sciences, **473**, pp. 59-72 (2019).
- L. Fei, B. Zhang, Y. Xu, D. Huang, W. Jia, J. Wen, *Local discriminant direction binary pattern for palmprint representation and recognition*, IEEE Transactions on Circuits and Systems for Video Technology, **30**, 2, pp. 468-481 (2019).
- A. El Idrissi, Y. Ruichek, *Palmprint recognition using state-of-the-art local texture descriptors: a comparative study*, IET Biometrics, **9**, 4, pp. 143-153(2020).
- L. Wu, Y. Xu, Z. Cui, Y. Zuo, S. Zhao, L. Fei, *Triple-Type Feature Extraction for Palmprint Recognition*, Sensors, **21**, 14, pp. 1-15 (2021).
- S. Zhao, B. Zhang, *Deep discriminative representation for generic palmprint recognition*, Pattern Recognition, **98**, pp. 1-11 (2020).
- S. Zhao, B. Zhang, *Learning salient and discriminative descriptor for palmprint feature extraction and identification*, IEEE Transactions on Neural Networks and Learning Systems, **31**, 12, pp. 5219-5230 (2020).
- S. Zhao, B. Zhang, *Learning complete and discriminative direction pattern for robust palmprint recognition*, IEEE Transactions on Image Processing, **30**, pp. 1001-1014 (2020).
- S. Zhao, B. Zhang, *Joint constrained least-square regression with deep convolutional feature for palmprint recognition*, IEEE Transactions on Systems, Man, and Cybernetics: Systems, **52**, 1, pp. 511-522 (2022).
- L. Fei, G. Lu, W. Jia, S. Teng, D. Zhang, *Feature extraction methods for palmprint recognition: A survey and evaluation*, IEEE Transactions on Systems, Man, and Cybernetics: Systems, **49**, 2, pp. 346-363 (2018).
- D. Zhong, X. Du, K. Zhong, *Decade progress of palmprint recognition: A brief survey*, Neurocomputing, **328**, pp. 16-28 (2019).
- A.S. Ungureanu, S. Salahuddin, P. Corcoran, *Toward unconstrained palmprint recognition on consumer devices: A literature review*, IEEE Access, **8**, pp. 86130-86148 (2020).
- V. Roșca, A. Ignat, *Quality of pre-trained deep-learning models for palmprint recognition*, 22nd International Symposium on Symbolic and Numeric Algorithms for Scientific Computing (SYNASC), pp. 202-209 (September 2020).
- X. Wu, Q. Zhao, W. Bu, *A SIFT-based contactless palmprint verification approach using iterative RANSAC and local palmprint descriptors*, Pattern Recognition, **47**, 10, pp. 3314-3326 (2014).
- N. Charfi, H. Trichili, A.M. Alimi, B. Solaiman, *Bimodal biometric system for hand shape and palmprint recognition based on SIFT sparse representation*, Multimedia Tools and Applications, **76**, 20, pp. 20457-20482 (2017).
- A.S. ELayed, H.M. Ebeid, M.I. Roushdy, Z.T. Fayed, *Masked SIFT with align-based refinement for contactless palmprint recognition*, IET Biometrics, **8**, 2, pp. 150-158 (2019).
- J. Almaghtuf, F. Khelifi, *Self-geometric relationship filter for efficient SIFT key-points matching in full and partial palmprint recognition*, IET Biometrics, **7**, 4, pp. 296-304 (2018).
- P. Poonia, P.K. Ajmera, A. Bhalerao, *Palm-print recognition based on scale-invariant features*, 16th India Council International Conference (INDICON), Rajkot, India, pp. 1-4 (December 2019).
- D.G. Lowe, *Object recognition from local scale-invariant features*, Proceedings of the 7-th IEEE International Conference on Computer Vision, Kerkyra, Greece, **2**, pp. 1150-1157 (September 1999).
- D.G. Lowe, *Distinctive image features from scale-invariant keypoints*, International Journal of Computer Vision, **60**, 2, pp. 91-110 (2004).
- I. Păvăloi, A. Ignat, *Iris image classification using SIFT features*, Procedia Computer Science, **159**, pp. 241-250 (2019).
- A. Ignat, I. Păvăloi, *Occluded iris recognition using SURF features*, VISGRAPP (5: VISAPP), pp. 508-515 (2021).
- OpenCV Library - <https://opencv.org/>
- A. Păsărică, R. G. Bozomitu, D. Tărniceanu, G. Andrușeac, H. Costin, C. Rotariu, *Analysis of eye image segmentation used in eye tracking applications*, Rev. Roum. Sci. Techn.–Électrotechn. et Énerg., **62**, 2, pp. 215-222 (2017).