# Adaptation of SIFT Features for Robust Face Recognition

Janez Križaj, Vitomir Štruc, Nikola Pavešić

Faculty of Electrical Engineering, University of Ljubljana,
Tržaška 25, SI-1000 Ljubljana, Slovenia
{janez.krizaj,vitomir.struc,nikola.pavesic}@fe.uni-lj.si

**Abstract.** *The Scale Invariant Feature Transform* (SIFT) is an algorithm used to detect and describe scale-, translation- and rotation-invariant local features in images. The original SIFT algorithm has been successfully applied in general object detection and recognition tasks, panorama stitching and others. One of its more recent uses also includes face recognition, where it was shown to deliver encouraging results. SIFT-based face recognition techniques found in the literature rely heavily on the so-called keypoint detector, which locates interest points in the given image that are ultimately used to compute the SIFT descriptors. While these descriptors are known to be among others (partially) invariant to illumination changes, the keypoint detector is not. Since varying illumination is one of the main issues affecting the performance of face recognition systems, the keypoint detector represents the main source of errors in face recognition systems relying on SIFT features. To overcome the presented shortcoming of SIFT-based methods, we present in this paper a novel face recognition technique that computes the SIFT descriptors at predefined (fixed) locations learned during the training stage. By doing so, it eliminates the need for keypoint detection on the test images and renders our approach more robust to illumination changes than related approaches from the literature. Experiments, performed on the Extended Yale B face database, show that the proposed technique compares favorably with several popular techniques from the literature in terms of performance.

**Key words:** SIFT, keypoint detector, SIFT descriptor, face recognition

## 1 Introduction

Face recognition is extensively used in a wide range of commercial and law enforcement applications. Over the past years many algorithms have been proposed for facial recognition systems. These algorithms include two basic aspects: holistic, e.g. PCA (Principal Component Analysis [1]) and LDA (Linear Discriminant Analysis [2]), and feature-based, e.g., Gabor- and Scale Invariant Feature Transform-based (or SIFT-based) methods [3], [4]. Holistic approaches use the entire face region for the task of feature extraction and, therefore, avoid difficulties in the detection of specific facial landmarks. Feature-based approaches, on the other hand, extract local features from specific feature points of the face. Generally, holistic approaches obtain better results on images captured in controlled conditions, while feature-based approaches exhibit robustness to variations caused by expression or pose changes.

One of the more recent additions to the group of feature-based face recognition techniques is the Scale Invariant Feature Transform (SIFT) proposed by Lowe in [4]. The SIFT technique and its corresponding SIFT features have many properties that make them suitable for matching different images of an object or a scene. The features are invariant to image scaling and rotation, (partial) occlusion and to a certain extent also to changes in illumination and 3D camera view point. The SIFT technique works by first detecting a number of interest points (called keypoints) in the given image and then computing local image descriptors at these keypoints. When performing recognition (or classification), each keypoint descriptor from the given image is matched independently against all descriptors extracted from the training images, and based on the outcome of the matching procedure, the image is assigned to a class featured in the database.

Event though the SIFT technique represent one of the state-of-the-art approaches to object detection/recognition, it has some deficiencies when applied to the problem of face recognition. Compared to general objects, there are less structures with high contrast or high-edge responses in facial images. Since keypoints along edges and low-contrast keypoints are removed by the original SIFT algorithm, interest points representing distinctive facial features can also be removed. Therefore, it is of paramount importance to properly adjust the thresholds governing the process of unstable keypoint removal, when applying the SIFT technique for the task of face recognition. In any case, the adjustment of the keypoint-removal-threshold represents a trial and error procedure that inevitably leads to suboptimal recognition performance.

Another thing to be considered, when using the SIFT technique for face recognition, are false matched keypoints. The majority of SIFT-based approaches employed for face recognition use different partitioning schemes to determine a number of subregions on the facial image and then compare the SIFT descriptors only between corresponding subregions. Due to the ”local” matching, wrong matches between spatially inconsistent SIFT descriptors are partially eliminated. However, variable illumination still has significant influence on the detection of keypoints, since the keypoint detector intrinsic to the SIFT technique is not invariant to illumination.

To overcome the presented shortcomings of the original SIFT technique (for face recognition), we propose in this paper a novel SIFT-based approach to face recognition, where the SIFT descriptors are computed at fixed predefined image locations learned during the training stage. By fixing the keypoints to predefined spatial locations, we eliminate the need for threshold optimization and face image partitioning, while the developed approach gains greater illumination invariance than other SIFT adaptations found in the literature.

The proposed method, called Fixed-keypoint-SIFT (FSIFT), was compared to several other approaches found in the literature. Experimental results obtained on the Extended Yale B face database show, that, under severe illumination conditions, consistently better results can be achieved with the proposed approach than with popular face recognition methods, such as PCA and LDA or other SIFT-based approaches from the literature.

## 2 The Scale-invariant Feature Transform

This section reviews the basics of the SIFT algorithm, which according to [4] consists of four computational stages: *(i)* scale-space extrema detection, *(ii)* removal of unreliable keypoints, *(iii)* orientation assignment, and *(iv)* keypoint descriptor calculation.

### 2.1 Scale-space extrema detection

In the first stage, interest points called keypoints, are identified in the scale-space by looking for image locations that represent maxima or minima of the difference-of-Gaussian function. The scale space of an image is defined as a function $L(x, y, \sigma)$, that is produced from the convolution of a variable-scale Gaussian, $G(x, y, \sigma)$, with the input image, $I(x, y)$:

$$L(x, y, \sigma) = G(x, y, \sigma) * I(x, y), \tag{1}$$

with

$$G(x, y, \sigma) = \frac{1}{2\pi\sigma^2} e^{-(x^2+y^2)/2\sigma^2}, \tag{2}$$

where $\sigma$ denotes the standard deviation of the Gaussian $G(x, y, \sigma)$.

The difference-of-Gaussian function $D(x, y, \sigma)$ can be computed from the difference of Gaussians of two scales that are separated by a factor $k$:

$$D(x, y, \sigma) = (G(x, y, k\sigma) - G(x, y, \sigma)) * I(x, y) = L(x, y, k\sigma) - L(x, y, \sigma) \tag{3}$$

Local maxima and minima of $D(x, y, \sigma)$ are computed based on the comparison of the sample point and its eight neighbors in the current image as well as the nine neighbors in the scale above and below. If the pixel represents a local maximum or minimum, it is selected as a candidate keypoint.

### 2.2 Removal of unreliable keypoints

The final keypoints are selected based on measures of their stability. During this stage low contrast points (sensitive to noise) and poorly localized points along edges (unstable) are discarded. Two criteria are used for the detection of unreliable keypoints. The first criterion evaluates the value of $|D(x, y, \sigma)|$ at each candidate keypoint. If the value is below some threshold, which means that the structure has low contrast, the keypoint is removed. The second criterion evaluates the ratio of principal curvatures of each candidate keypoint to search for poorly defined peaks in the Difference-of-Gaussian function. For keypoints with high edge responses, the principal curvature across the edge will be much larger than the principal curvature along it. Hence, to remove unstable edge keypoints based on the second criterion, the ratio of principal curvatures of each candidate keypoint is checked. If the ratio is below some threshold, the keypoint is kept, otherwise it is removed.

### 2.3 Orientation assignment

An orientation is assigned to each keypoint by building a histogram of gradient orientations $\theta(x, y)$ weighted by the gradient magnitudes $m(x, y)$ from the keypoint's neighborhood:

$$m(x,y) = \sqrt{(L(x+1,y) - L(x-1,y))^2 + (L(x,y+1) - L(x,y-1))^2}, \quad (4)$$

$$\theta(x,y) = \tanh\left(L(x,y+1) - L(x,y-1))/(L(x+1,y) - L(x-1,y)\right), \quad (5)$$

where $L$ is a Gaussian smoothed image with a closest scale to that of a keypoint. By assigning a consistent orientation to each keypoint, the keypoint descriptor can be represented relative to this orientation and, therefore, invariance to image rotation is achieved.

### 2.4 Keypoint descriptor calculation

The keypoint descriptor is created by first computing the gradient magnitude and orientation at each image point of the 16×16 keypoint neighborhood (left side of Fig. 1). This neighborhood is weighted by a Gaussian window and then accumulated into orientation histograms summarizing the contents over subregions of the neighborhood of size $4 \times 4$ (see the right side of Fig. 1), with the length of each arrow in Fig. 1(right) corresponding to the sum of the gradient magnitudes near that direction within the region [4]. Each histogram contains 8 bins, therefore each keypoint descriptor features $4 \times 4 \times 8 = 128$ elements. The coordinates of the descriptor and the gradient orientations are rotated relative to the keypoint orientation to achieve orientation invariance and the descriptor is normalized to enhance invariance to changes in illumination.

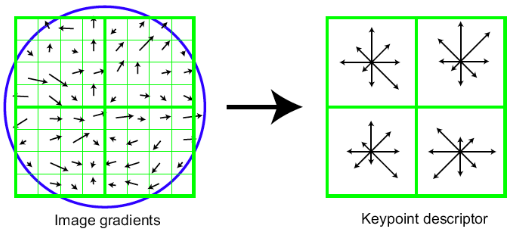


**Fig. 1.** In this figure the $2 \times 2$ subregions are computed from an $8 \times 8$ neighborhood, whereas in the experiments we use a $16 \times 16$ neighborhood and subregions of size $4 \times 4$.

### 2.5 Matching

When using the SIFT algorithm for object recognition, each keypoint descriptor extracted from the query (or test) image is matched independently to the database of descriptors extracted from all training images. The best match for each descriptor is found by identifying its nearest neighbor (closest descriptor) in the database of keypoint descriptors from the training images. Generally, many features from a test image do not have any correct match in the training database, because they were either not detected in the training image or they arose from background clutter. To discard keypoints whose descriptors do not have any good match in the training database, a subsequent threshold is used, which rejects matches that are too ambiguous. If the distance ratio between the closest neighbor and the second-closest neighbor, (i.e., the closest neighbor that is known to come from a different object than the first) is below some threshold, than the match is kept, otherwise the match is rejected and the keypoint is removed. The object in the database with the largest number of matching points is considered the matched object, and is used for the classification of the object in the test image.

## 3 SIFT-based Face Recognition

Over the past few years there have been some studies (from the early studies, e.g., [5], [6] to more recent ones, such as [12]) assessing the feasibility of the SIFT approach for face recognition. The progress of the SIFT technique for face recognition can be summarized as follows:

One of the first attempts to use the SIFT algorithm for face recognition was presented in [5]. The algorithm used here, differs from original SIFT algorithm in the implementation of the matching stage. Each SIFT descriptor in the test image is matched with every descriptor in each training image. Matching is done using a distance based criterion. A descriptor from the test image is said to match a descriptor from the training image, if the distance between the 2 descriptors is less than a specific fraction of the distance to the next nearest descriptor. The problem with this method is that it is very time consuming. Matching between two images has a computational complexity of $\mathcal{O}(n^2)$, where $n$ is the average number of SIFT descriptors in each image.

In [6], the original SIFT algorithm is rendered more robust by following one of two strategies that aim at imposing local constraints on the matching procedure: the first matches only SIFT descriptors extracted from image-windows corresponding to the mouth and the two eyes, while the second relies on grid-based matching, Local matching, i.e. within a grid or a cluster, constrains the SIFT features to match features from nearby areas only. Local matching also reduces the computational complexity linearly. The computational complexity required for matching a pair of images by a local method is $\mathcal{O}(n^2/s)$, where $s$ is the number of grids or clusters. As seen from Fig. 2, where the basic SIFT algorithm from [4] was used to match the SIFT descriptors, there are some keypoints matched, that do not represent the same characteristic of the face. Although we would expect the distance between such keypoints to be high, since they correspond to different regions of the faces, this is clearly not the case. Therefore better results are achieved, if certain subsets of SIFT keypoints are used for matching and only (spatially) corresponding subsets of SIFT descriptors are matched (as is [6] and later in [7], [9], [10] and [11]).

Both local and global information for face recognition are used in [7]. Instead of using a grid based approach, the SIFT features are clustered into 5 clusters

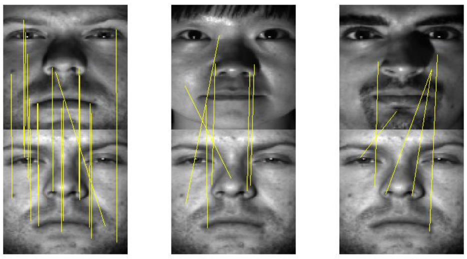


**Fig. 2.** Match results for one of the test images (bottom image) with a set of training faces (top) using the basic SIFT algorithm.

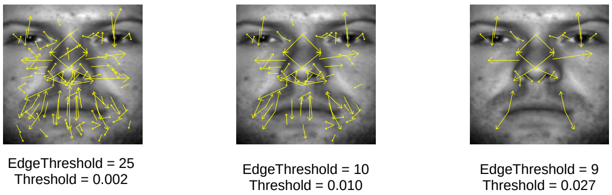
using kmeans clustering (2 clusters for the eyes, one for the nose, and 2 clusters at the edges of the mouth). Only the SIFT descriptors between two corresponding clusters are matched. This ensures that matching is done locally. As a global matching criterion, the total number of descriptor matches (as in [4]) is used.

In [8] SIFT features are extracted from the frontal and half left and right profiles. An augmented set of SIFT features is then formed from the fusion of features from the frontal and side profiles of an individual, after removing feature redundancy. SIFT feature sets from the database and query images are matched using the Euclidean distance and Point pattern matching techniques.

In [9] a Graph Matching Technique is employed on the SIFT descriptors to to deal with false pair assignment and reduce the number of SIFT features. In [10] SIFT features are ranked according to a discriminative criterion based on Fisher's Discriminant Analysis (similar as in [2]), so that the chosen features have the minimum within-class variation and maximum variation between classes. In [11] both global and local matching strategies are used. In order to reduce the identification errors, the Dempster-Shafer decision theory is applied to fuse the two matching techniques.

In [12] an approach called Keypoints-Preserving-SIFT (KPSIFT) is proposed. The KPSIFT approach keeps all the initial keypoints for SIFT descriptor calculation. This procedure greatly differs from the basic SIFT approach, where unreliable keypoints are removed as explained in section 2. However, this removal can eliminate some keypoints and discard potentially useful discriminative information for face recognition. With the basic SIFT procedure intrinsic properties of the face images have to be considered (recall that facial images contain only a few structures with high contrast or high-edge responses, which often leads to the removal of useful keypoints), when setting the threshold values governing the process of keypoint removal. As shown in [12], recognition rates improve when adjusting thresholds on low-contrast and edge keypoints in order to accept more keypoints. Fig. 3 shows three different adjustments of the (keypoint-removal) thresholds. Here, the threshold denoted as *EdgeThreshold* controls the removal of poorly localized keypoints along edges, while the threshold denoted as *Threshold* controls the removal of low contrast keypoints (see Section 2.2 for details).

The experiments in [12] show that the best recognition results are achieved with the thresholds resulting in the left image of Fig. 3.



EdgeThreshold = 25
Threshold = 0.002

EdgeThreshold = 10
Threshold = 0.010

EdgeThreshold = 9
Threshold = 0.027

**Fig. 3.** Keypoints detected in a sample face image with respect to the (keypoint-removal) threshold values: Result improving values (left), common values (middle), high-elimination threshold values (right).

While the presented techniques try to compensate the imperfections of the keypoint detector by imposing local matching constraints, by relaying on sub-windows of the images, by deploying graph-matching techniques, etc., we present in the remainder a simple procedure, which completely eliminates the need for the keypoint detector (in the test stage). With the proposed procedure, most shortcomings of the detector, such as susceptibility to illumination, influence of the (keypoint-removal) thresholds and false keypoint detections are solved.

## 4 The Fixed Keypoint SIFT Algorithm

### 4.1 Fixing the keypoints

Our method, the Fixed Keypoint SIFT Algorithm or FSIFT for short, is based on the supposition that each face was preliminary localized. Thus, each image consists only of a properly registered face region of a certain person.

We assume that for the training procedure only "good" quality images are available. This assumption is reasonable, since in most operating face recognition systems the enrollment stage and with it the acquisition of training images is supervised. During training we apply the original SIFT technique and its accompanying keypoint detector (with the (keypoint-removal) threshold adjusted - Fig. 3 left) to our training images and obtain a number of candidate keypoints for each image in the set of training images (first three images of Fig. 4). Next, we apply a clustering procedure to the set of candidate keypoints to obtain $k = 100$ centroids, which serve as the *fixed* keypoints for the computation of the SIFT descriptors. We can see in the fourth image of Fig. 4 that most of these centroids correspond to distinctive facial landmarks, such as the eyes, nose or the mouth.

Fig. 5 illustrates the advantages gained by the proposed approach. Here, the first image (from the left) depicts the keypoints locations found by original keypoint detector, while the second image presents the location of keypoints in the image of the same person captured in different illumination conditions. Not only the number of detected keypoints in the second image is smaller than in the first image, many of the keypoints are detected in different locations than

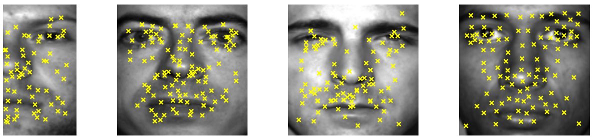


**Fig. 4.** Training procedure for learning the keypoint locations: sample images processed with the original keypoint detector (images one through three), the learned keypoint locations (fourth image).

in the first image and therefore a reduction of keypoint matches is expected. If SIFT descriptors are computed at fixed predefined locations (third and fourth image of Fig. 5) a greater robustness to illumination variations can be achieved.

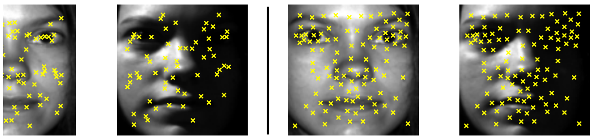


**Fig. 5.** SIFT keypoints detected on the differently illuminated images of the same person: by the original keypoint detector (first two images from the left), and by the proposed method (third and fourth image).

### 4.2 Matching

As the number of descriptors for each image is the same (it equals the number of centroids $k$), the sum of the Euclidean distances between equally located descriptors of the two images to be compared is used as the matching criterion. By doing so, computational complexity for matching between two images is also reduced to $\mathcal{O}(2k)$. Let us denote the sets of SIFT descriptors from the training images as $\mathcal{S}_j = \{S_{i,j}(x_i, y_i); i = 1, 2, ..., k\}$, where $j = 1, 2, ..., n$ denotes the training image index, $n$ stands for the total number of training images, $i$ represents the descriptor index, $k$ denotes the number of fixed keypoint locations (i.e., centroids), and $(x_i, y_i)$ denote the image location for the $i$-th SIFT descriptor. Let us further assume that the $n$ training images correspond to $N$ different classes (i.e., subjects) with corresponding class labels $\omega_1, \omega_2, ..., \omega_N$. Then, the matching procedure can formally be written as follows:

$$\delta_{SL_2}(\mathcal{S}_g, \mathcal{S}_t) = \min_j \delta_{SL_2}(\mathcal{S}_j, \mathcal{S}_t) \rightarrow \mathcal{S}_t \in \omega_g, \tag{6}$$

where $\mathcal{S}_t$ stands for the set of SIFT descriptor extracted from the test image at the $k$ predefined image locations, and the matching function is defined as $\delta_{SL_2}(\mathcal{S}_p, \mathcal{S}_r) = \sum_i \delta_{L_2}(S_{i,p}, S_{i,r})$.

The above expression postulates that a given test image is assigned to the class $\omega_g$, if the sum of the Euclidian distances between spatially corresponding descriptors of the test image and one of the training images of the $g$-th class is the smallest among the computed distances to all $n$ SIFT descriptor sets of the training images.

## 5 Experiments and Results

The experiments were done on the Extended Yale B (EYB) face database [15]. The database contains 38 subjects and each subject has approximately 64 frontal view images taken under different illuminations conditions. For the experiments the images were partitioned into five subsets. In the first image subset (S1 in the remainder), there are images captured in relatively good illumination conditions, while for the image subsets labeled S2 to S5, the lighting conditions get more extreme. S1 is used as the training set, while images in the other subsets are used as test images. It should be noted that the numbers in the brackets next to the subset label in Table 1 represent the number of images in each subset. All algorithms were implemented in Matlab relying partially on existing code available from [13] and [14]. The performance of the proposed approach was compared to the performance of some other face recognition techniques such as PCA, LDA, and to several different modifications of the SIFT algorithm. Table 1 presents the performance of the listed methods in form of rank one recognition rates for changeable illumination conditions. The recognition rates of PCA and LDA are shown in the first and second row, respectively. The original SIFT method is shown in the third row. The fourth row presents the results of the method from [7], which relies on clustering of the SIFT keypoints. With our method, denoted as FSIFT in the last row, better results are achieved in comparison

**Table 1.** Rank one recognition rates (in %) obtained on the EYB database.

| Method | S2 (456) | S3 (455) | S4 (488) | S5 (752) |
|---|---|---|---|---|
| PCA | 93.6 | 55.0 | 16.7 | 22.0 |
| LDA | 100 | 99.8 | 56.3 | 51. 0 |
| SIFT | 100 | 45.7 | 25.7 | 11.2 |
| SIFT_CLUSTER | 100 | 100 | 66.8 | 64.9 |
| **FSIFT** | **100** | **100** | **83.2** | **82.8** |

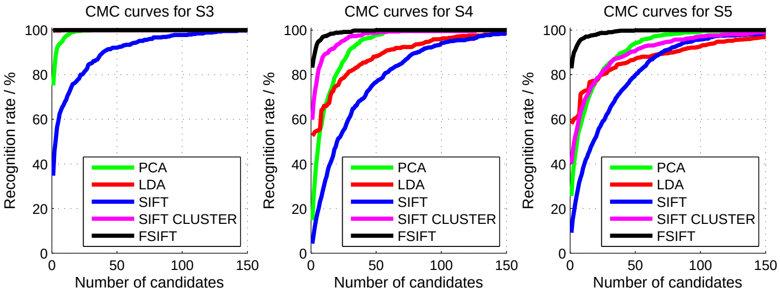


**Fig. 6.** Cumulative match curves for each subset.

with the recognition performance of the remaining techniques assessed in our experiments.

In Fig. 6, the results are presented as cumulative match curves (CMC) for subsets three through five. It should be noted that the CMCs are not shown for subset two, as all tested techniques achieve a perfect recognition rate of 100% for all ranks. From the results we can see that the FSIFT approach clearly outperformed all other techniques assessed in the comparison.

## 6 Conclusion and Future Work

In this paper an adaptation of the SIFT algorithm for face recognition was presented. Using the EYB database, we have shown that the performance of the proposed method is significantly better than the performance of popular techniques such as PCA or LDA and different SIFT-based recognition techniques from the literature. To be able to cope with possible pose variations, we plan to augment the proposed FSIFT technique with a pose detector and, consequently, extend it to a multi-pose version.

## References

1. Turk M., Pentland A.: Face Recognition Using Eigenfaces, Proc. IEEE Conference CVPR. pp. 586-591, 1991.
2. Etemad K., Chellappa R.: Discriminant Analysis for Recognition of Human Face Images, J. of the Opt. Society of America A, Vol. 14, No. 8, pp. 1724–1733, 1997.
3. Wiskott L., Fellous J.M., Kruger N., Malsburg C. von der: Face Recognition by Elastic Bunch Graph Matching, IEEE TPAMI, Vol. 19, No. 7, pp. 775–779, 1997.
4. Lowe D.G.: Distinctive Image Features From Scale-Invariant Keypoints, International Journal of Computer Vision, Vol. 60, pp. 91–110, 2004.
5. Aly M.: Face Recognition using SIFT Features, CNS/Bi/EE report 186, 2006.
6. Bicego M., Lagorio A., Grosso E., Tistarelli M.: On the Use of SIFT Features for Face Authentication, CVPR Workshop, pp. 35–35, 2006.
7. Jun Luo, Ma Y., Takikawa E., Lao S., Kawade M., Bao-Liang Lu: Person-Specific SIFT Features for Face Recognition, ICASSP, pp. 593–596, 2007.
8. Rattani A., Kisku D.R., Lagorio A., Tistarelli M.: Facial Template Synthesis based on SIFT Features, IEEE Workshop AIAT, pp. 69–73, 2007.
9. Kisku D.R., Rattani A., Grosso E., Tistarelli M.: Face Identification by SIFT-based Complete Graph Topology, IEEE Workshop AIAT, pp. 63–68, 2007.
10. Majumdar A., Ward R.K.: Discriminative SIFT Features for Face Recognition, Canadian Conference on Electrical and Computer Engineering, pp. 27–30, 2009.
11. Kisku D.R., Tistarelli M., Sing J.K., Gupta P.: Face Recognition by Fusion of Local and Global Matching Scores Using DS Theory: An Evaluation With Uni-Classifier and Multi-Classifier Paradigm, CVPR Workshop, pp. 60–65, 2009.
12. Geng C., Jiang X.: SIFT features for Face Recognition, IEEE Conference CSIT, pp. 598–602, 2009.
13. Lowe D.G.: Software for SIFT, `http://people.cs.ubc.ca/~lowe/keypoints/`
14. Vedaldi A.: MATLAB/C implementation of the SIFT detector and descriptor, `http://www.vlfeat.org/~vedaldi/code/sift.html`
15. Georghiades A.S., Belhumeur P.N., Kriegman D.J.: From Few to Many: Illumination Cone Models for Face Recognition Under Variable Lighting and Pose, IEEE TPAMI, Vol. 23, No. 6, pp. 643–660, 2001.