International Journal of Signal Processing, Image Processing and Pattern Recognition
Vol.7, No.1 (2014), pp.225-236
http://dx.doi.org/10.14257/ijsip.2014.7.1.21

Face Recognition using the most Representative Sift Images

Issam Dagher, Nour El Sallak and Hani Hazim

University of Balamand
Department of Computer Engineering
dagheri@balamand.edu.lb

Abstract

In this paper, face recognition using the most representative SIFT images is presented. It is based on obtaining the SIFT (SCALE INVARIANT FEATURE TRANSFORM) features in different regions of each training image. Those regions were obtained using the K-means clustering algorithm applied on the key-points obtained from the SIFT algorithm. Based on these features, an algorithm which will get the most representative images of each face is presented. In the test phase, an unknown face image is recognized according to those representative images. In order to show its effectiveness this algorithm is compared to other SIFT algorithms and to the LDP algorithm for different databases.

Keywords: *Face recognition, SIFT, LDP, Clustering, matching*

1. Introduction

Facial expressions supply an important behavioral measure in order to study the features of the image [1]. Nowadays, automatic facial recognition systems have many applications. Many face recognition techniques exist [2]. Some of them are:

- Principal Component Analysis (PCA) [3, 6]. This algorithm decreases the large dimensionality of the data space. It also extracts the features present in the face images such features may or may not be directly related to face features such as eyes, nose, lips, and hair [4].
- Linear discriminant analysis (LDA) [5]. It extracts feature from a face image and reduces dimensions in pattern recognition. The training face images are mapped to the fisher-space for classification. In the classification phase, an input face is projected to the same fisher-space and classified by an appropriate classifier [8].
- The EBGM algorithm takes land marks on an image face for the most essential characteristics of a face image like eyes, nose and mouth. These land mark points are to be located on each image [9]. And based on the positions of the land marks the face is recognized.

Although the holistic feature can measure the entire characteristic of an image, it cannot avoid losing some details within an image. Many recent researches show that local features are more effective to describe the detailed and stable information of an image. Some of them are:

- Local binary pattern (LBP) [7]. The algorithm considers a 3x3 pixels window. Then the center is compared to each of its 8 neighbors (Equation 1). This will result in a binary representation of the new pixel formed.

ISSN: 2005-4254 IJSIP

And finally Substitute the original pixel with the new decimal one.

- Local Derivative Pattern (LDP) [7]. The power behind LDP is not only the high order derivative; *i.e.*, more features from far pixels; but is also the capability of varying directions.
- Scale Invariant Feature Transform (SIFT) proposed by D. Lowe [10]

We propose a method based on the SIFT for face recognition. We select the most representative images in order to decrease the size of data. We use the k-means algorithm to obtain stable sub-regions from training images and calculate the matching similarity of all equivalent region pairs.

2. Scale Invariant Feature Transform

Scale-invariant feature transform is a method that detects local features in images. The method was published by David G Lowe [10]. The SIFT algorithm could be split up into the following multiple parts:

2.1. Construction a Scale Space

The main objective of the scale space part is to get rid of unnecessary and false details from the image. This is done by using a Gaussian Blur filter .The process of scale space construction consists of generating progressively blurred out images with different sizes. SIFT use four octaves or scales which are made by resizing the original image to half size each time.

Blurring (1) is simply the convolution of the Gaussian operator and the original image.

$$L(x,y,\sigma) = G(x,y,\sigma) * I(x,y) \qquad (1)$$

Where the Gaussian Operator is given by:

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

x and y are the location coordinates, and $\boldsymbol{\sigma}$ is the scale parameter.

2.2. Laplacian of Gaussian Calculation

They are approximated by calculating the difference (DOG) between two nearby scales (Figure 1).

$$D(x,y,\sigma) = L(x,y,k_i\sigma) - L(x,y,k_j\sigma) \qquad (3)$$

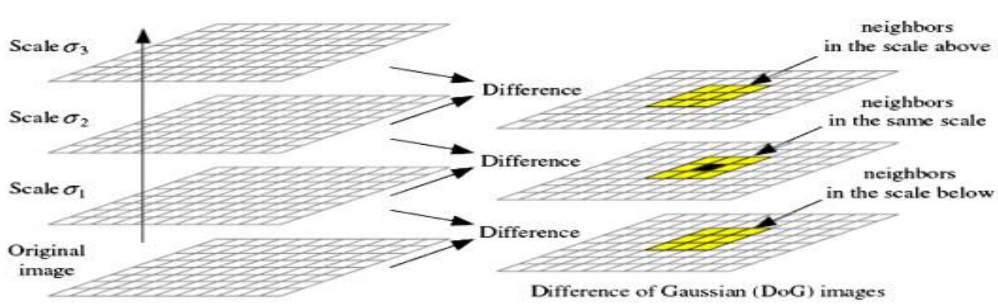

Figure 1. Difference of Gaussian Formation

2.3. Finding Key-points

Key-points are produced through the following 2 processes:

First locating maxima and minima: To detect the local maxima and minima of DOG, each point is compared with its 8 neighbors at the same scale, and its 9 neighbors up and down one scale. If the point is the greatest or the least of all 26 neighbors, it is marked as key-point.

Then finding sub-pixel maxima/minima: These key-points could be estimated using Taylor Series expansion.

2.4. Eliminating Edges and Low Contrast Regions

Low contrast regions are removed by checking their intensities and comparing it to a threshold. If the pixel of DoG image is less than a certain value it will be rejected.

After that, the goal is to remove edges, to find the corners and eliminating the flat regions. To do those, two gradients should be calculated at each key-point so 3 cases could happen (note that these gradients are perpendicular to each other):

1- For flat regions the gradients are both small.

2- For edges, one of the two gradients will be big

3- For corners, both gradients are big.

So when the both gradients are big, the point is considered as a key-point, and eliminated in the other two cases.

2.5. Assigning an Orientation to the Key-points

To assign an orientation to a key-point, the gradient directions and magnitudes should be calculated around this key-point and the dominant orientation in that region is assigned to the key-point .The size for the assigned orientation depends on its scale. Equations (4) and (5) are the gradient magnitude and the gradient orientation respectively.

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

$$\theta(x,y) = tan^{-1}(\frac{L(x,y+1) - L(x,y-1)}{L(x+1,y) - L(x-1,y)} \quad (5)$$

After calculating the gradient magnitude and orientation for all pixels around the key-point a histogram is created. This histogram is broken into 36 bins, each bin contains 10 degrees.

2.6. SIFT Features Generation

To avoid any illumination and orientation issues, each key-point is assigned a 128 dimensional vector. To do this the following steps should be done:

- A 16*16 window around the key-point is selected.
- This window is divided into sixteen 4*4 window.
- For each 4*4 window, the magnitude and orientation are calculated and a histogram is made of the results.

- This histogram is divided into 8 bins and the amount of orientation added to the bin depends to the gradient magnitude (using Gaussian weighting function).Finally each key-point is represented by 4*8*8 = 128 number.

Now each image is represented by a certain number of key-points, and each key-point is a vector of 128 components.

3. Training Phase

Our training algorithm consists of 2 stages:

1- Forming the k regions of each training image. Where each region is characterized by a set of SIFT features.
2- Obtaining the most representative images of each face.

3.1. The k-regions Formation

It can be summarized by the following flowchart:

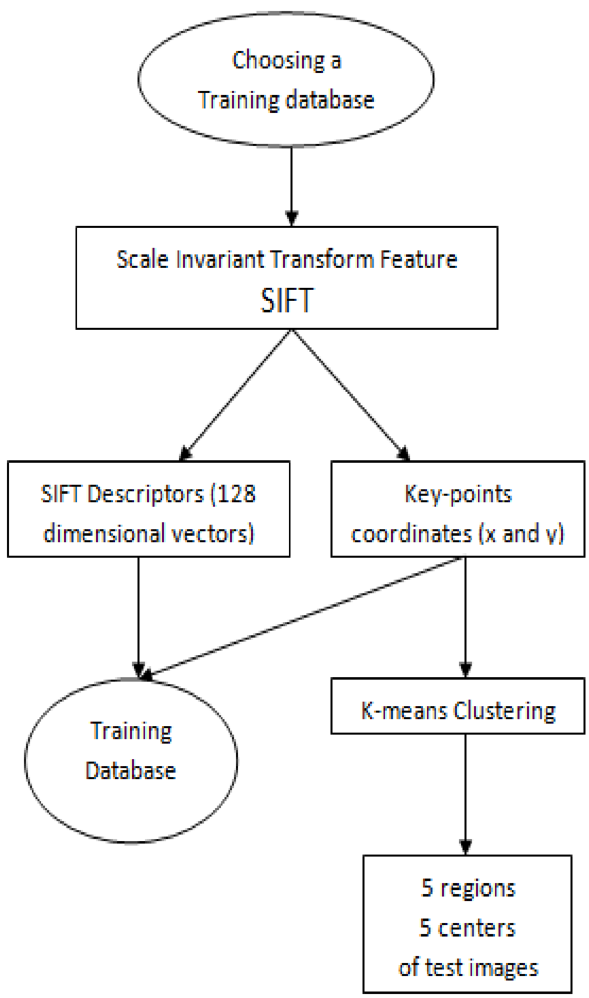

Figure 2. The k-regions Formation

It consists of the following steps:

1- Apply the SIFT algorithm to each training image. This will give a set of 128 dimensional key-points and their x-y coordinates.

2- Apply the k-means algorithm to the x-y coordinates. This will give the k-regions where each region is characterized by set of 128 vectors.

3.2. Obtaining the most Representative Images

It can be summarized by the following flowchart:

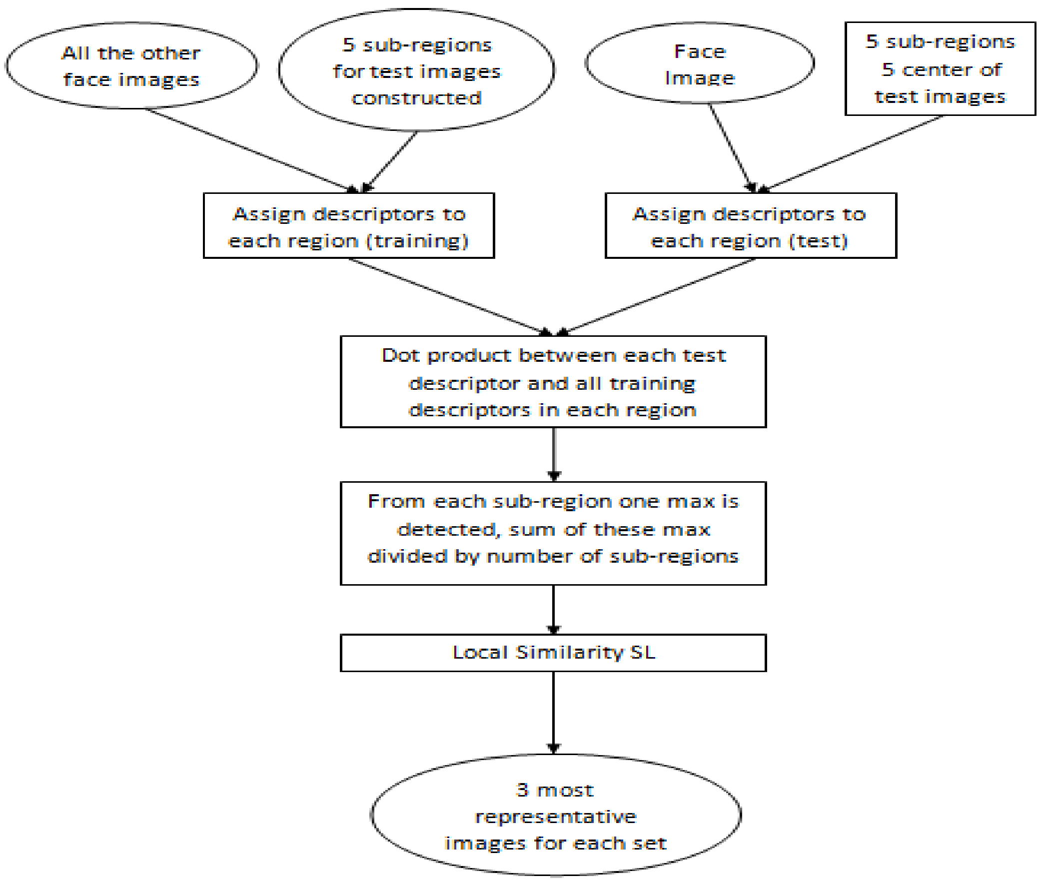

Figure 3. Getting the M representative Images using k=5 Regions

A For each face with its images 1,..., N . Do the following steps:

1- do the following steps for image I and the other images j=1,…,N ($j \neq i$)

2- Dot product between the descriptors of image i (in each region)and all other descriptors of image j (on same region)

3- For each region the maximum dot product is selected

4- Sum the value of those dot products

5- Find the local similarity SL

Find the most representative images which have the maximum values of SL. A face image is represented by $(m_1, m_2, \ldots, m_k)$features scattered in k sub-regions and denoted by $I = \left(f_1^1, \ldots, f_1^{m_1}, f_2^1, \ldots, f_2^{m_2}, \ldots, f_k^1, \ldots, f_k^{m_k}\right)$

It should be noted that SL and d are given by the following formulas:

$$S_L(I_l, I_r) = \frac{1}{k}\sum_{i=1}^{k}\left(\max\left(d(f_{li}^x, f_{ri}^y\right) \times w_i\right) \quad (6)$$

$$d = \frac{\langle f_1, f_2 \rangle}{\| f_1 \| \cdot \| f_2 \|} \quad (7)$$

d denotes the similarity between two SIFT features

4. Test Phase

After the training phase, every face is characterized by M images. The test algorithm is shown in the following flow chart:

It should be noted that the global similarity measure SG is given by:

$$S_G(I_t, I_r) = \frac{match(I_t, I_r)}{\sum_{i=1}^{k} m_{ri}} \quad (8)$$

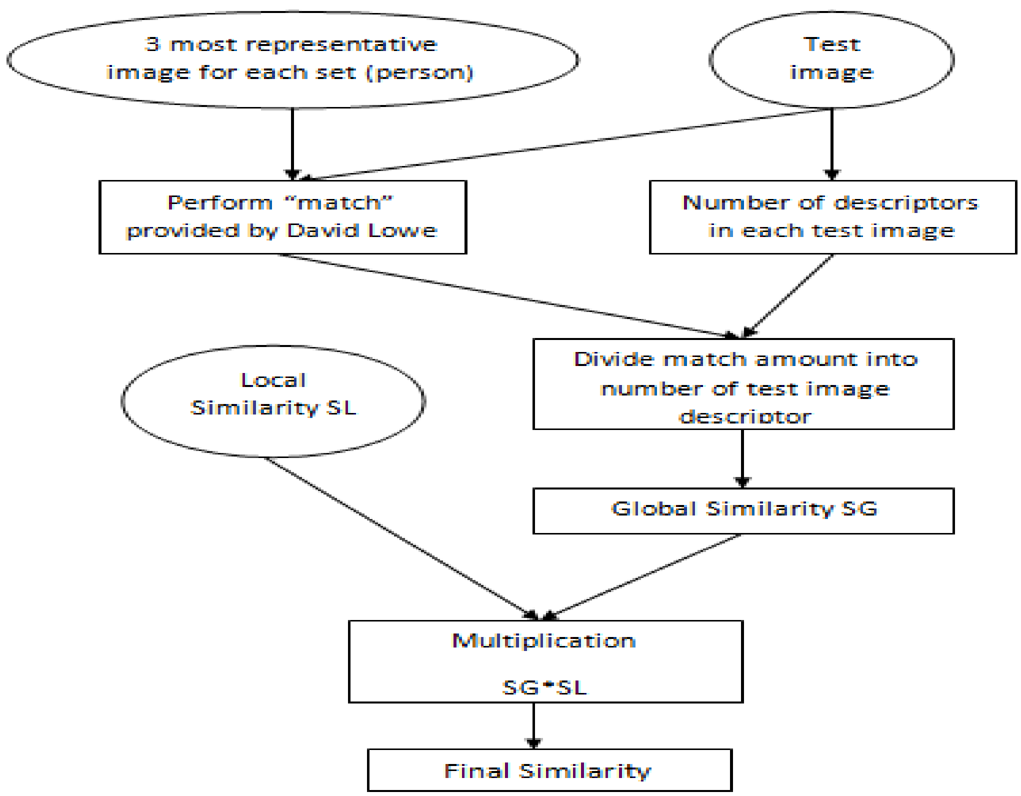

Figure 4. Recognizing a Test Image using M=3 Representative Images

It consists of the following steps:

1- Perform Match provided by David Lowe [10] on the test image and each of the training images.

2- The number of match key-points should be divided by the number of test image key-points.

3- The global similarity SG is obtained and presented by (8).

4- The Final Similarity S is obtained by multiplying the global similarity SG by the local similarity SL :

$$S = SG \times SL \qquad (9)$$

5. Experimental Results

Four popular face databases were used to demonstrate the effectiveness of the proposed algorithm. The ORL [11] contains a set of faces taken between April 1992 and April 1994 at the Olivetti Research Laboratory in Cambridge. It contains 40 distinct persons with 10 images per person. The images are taken at different time instances, with varying lighting conditions, facial expressions and facial details (glasses/no-glasses). All persons are in the up-right, frontal position, with tolerance for some side movement. The UMIST [12] taken from the University of Manchester Institute of Science and Technology. It is a multi-view database, consisting of 575 images of 20 people, each covering a wide range of poses from profile to frontal views.

The Yale [13] taken from the Yale Center for Computational Vision and Control. It consists of images from 15 different people, using 11 images from each person, for a total of 165 images. The images contain variations with following total expressions or configurations: center-light, with glasses, happy, left-light, without glasses, normal, right-light, sad, sleepy, surprised, and wink. And the BIOID database [14]. The dataset consists of 1521 gray level images with a resolution of 384x286 pixel. Each one shows the frontal view of a face of one out of 23 different test persons.

Each image in the ORL database is scaled into (92 × 112), in the UMIST Database is scaled into (112 × 92), the Yale Database is cropped and scaled into (126 × 152) and the BIOID is cropped and scaled to (128 x 95). To start the face recognition experiments, each one of the four databases is randomly partitioned into 60% training set and 40% test set with no overlap between the two. 10 different partitions were made.

Table 1 compares the average percentage recognition results of the following techniques:

- **LDP:** LDP [7] with different orders and different directions.
- **Aly:** SIFT matching by Aly [15].
- **Lenc-Kral:** SIFT matching by Lenc and Kral [16].
- **MR**: Our most representative algorithm using 5 representative images and 5 regions.

Table 1. Results for the 4 Databases

ORL		LDP	SIFT Aly	SIFT Lenc-Kral	MR
Order	Direction	Percentage	Percentage	Percentage	Percentage
1	0	75	70.5	77.6	87.6
	1	70			
	2	77.5			
	3	75			
2	0	70			
	1	72.5			
	2	75			
	3	72.5			
3	0	85			
	1	82.5			
	2	85			
	3	80			
4	0	65			
	1	62.5			
	2	65			
	3	65			

UMIST		LDP	SIFT Aly	SIFT Lenc-Kral	MR
Order	Direction	Percentage	Percentage	Percentage	Percentage
1	0	50.8	60.9	69.4	86.6
	1	50			
	2	52.2			
	3	51.2			
2	0	71.8			
	1	52.5			
	2	72.5			
	3	56.7			
3	0	80			
	1	77.5			
	2	79.2			
	3	80.2			
4	0	71.2			
	1	70			
	2	71.3			
	3	71.2			

YALE		**LDP**	**SIFT Aly**	**SIFT Lenc-Kral**	**MR**
Order	Direction	Percentage	Percentage	Percentage	Percentage
1	0	55.7	61.1	74.2	85.3
	1	55.2			
	2	57.6			
	3	53.3			
2	0	66.2			
	1	64			
	2	70			
	3	72.3			
3	0	75.5			
	1	72			
	2	75.5			
	3	70			
4	0	65			
	1	62.5			
	2	65			
	3	65			

BIOID		**LDP**	**SIFT Aly**	**SIFT Lenc-Kral**	**MR**
Order	Direction	Percentage	Percentage	Percentage	Percentage
1	0	54.2	65.7	74.3	86.5
	1	60.1			
	2	62.3			
	3	61.6			
2	0	59.1			
	1	60.7			
	2	62.3			
	3	67.1			
3	0	75.8			
	1	72.3			
	2	75.4			
	3	80.7			
4	0	57.1			
	1	59.7			
	2	60.3			
	3	62.1			

6. Conclusion

In this paper, face recognition using SIFT most representative images is presented. It is based on applying the K-means algorithm on the key-points obtained from the SIFT algorithm; thus dividing each image into different regions.

Our MR (most representative images) using 5 representative images and 5 regions is compared to the LDP with different directions (it should be noted that the LDP gives better performance than the LBP). Our MR is also compared to other SIFT matching algorithms. It gave the best recognition results.

References

[1] S. Ganesh Bashyal and K. Venayagamoorthy, “Recognition of facial expressions using Gabor wavelets and learning vector quantization”, Engineering Applications of Artificial Intelligence, vol. 21, no. 7, **(2008)** October, pp. 1056-1064.
[2] T. Ahonen, “Face description with local binary patterns: Application to face recognition”, IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 28, no. 12, **(2006)**, pp. 2037-2041.
[3] H. Moon, “Computational and performance aspects of PCA-based face-recognition algorithms”, Perception, vol. 30, no. 3, **(2001)**, pp. 303-321.
[4] D. L. Swets and J. Weng, “Using discriminant eigenfeatures for image retrieval”, IEEE Trans. Pattern Anal. Mach. Intell., vol. 18, no. 8, **(1996)**, pp. 831-836.
[5] L. Juwei, “Face recognition using LDA-based algorithms”, IEEE Transactions on Neural Networks, vol. 14, no. 1, **(2003)**, pp. 195-200.
[6] I. Dagher and R. Nachar, “Face recognition using IPCA-ICA algorithm”, IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 28, no. 6, **(2006)**, pp. 996-1000.
[7] Z. Baochang, “Local derivative pattern versus local binary pattern: face recognition with high order local pattern descriptor”, IEEE Transaction on Image Processing, vol. 19, no. 2, **(2010)**, pp. 827-832.
[8] K. Fukunaga, “Introduction to statistical pattern recognition”, Second ed., Academic Press.
[9] L. Wiskott, J. M. Fellous, N. Krüger and C. von der Malsburg, “Facerecognition by elastic bunch graph matching”, IEEE Trans. Pattern Anal. Mach. Intell., vol. 19, no. 7, **(1997)**, pp. 775-779.
[10] D. Lowe, “Distince image features from scale-invariant key-points”, Int. Journal of Computer Vision, vol. 60, no. 2, **(2004)**, pp. 91-110.
[11] ORL face database, website: http://www.cam-orl.co.uk/facedatabase.html, AT&T Laboratories Cambridge.
[12] UMIST face database, http://images.ee.umist.ac.uk/danny/database.html, Daniel Graham.
[13] Yalefacedatabase, www1.cs.columbia.edu/~belhumeur/pub/images/yalefaces/,ColubmbiaUniversity.
[14] Bioid face database, website: www.**bioid**.com/downloads/facedb/.
[15] M. Aly, “Face Recognition using SIFT Features”, Technical Report, Caltech, USA, **(2006)**.
[16] L. Lenc and P. Král, “Novel Matching Methods for Automatic Face Recognition using SIFT”, AIAI, **(2012)**, pp. 254-263.

Authors

Issam Dagher finished his MS in electrical engineering degree in 1994 from Florida International University, Miami, USA. He finished his Phd in 1997 from Univesity of Central Florida, Orlando USA. He is now an associate professor at the University of Balamand, Lebanon. His areas of interests are pattern recognition, neural networks, artificial intelligence, and computer vision. He published many papers on these topics.

Nour El Sallak and Hani Hazim finished their MS in electrical engineering degree in 2011 from University of Balamand. Their areas of interests are pattern recognition and image processing.