2-2015

Robust Textural Features for Real Time Face Recognition

Chen Cui
University of Dayton

Vijayan K. Asari
University of Dayton, vasari1@udayton.edu

Andrew D. Braun
University of Dayton

Robust textural features for real time face recognition

Chen Cui, Vijayan K. Asari and Andrew D. Braun
University of Dayton, Ohio, USA

ABSTRACT

Automatic face recognition in real life environment is challenged by various issues such as the object motion, lighting conditions, poses and expressions. In this paper, we present the development of a system based on a refined Enhanced Local Binary Pattern (ELBP) feature set and a Support Vector Machine (SVM) classifier to perform face recognition in a real life environment. Instead of counting the number of 1's in ELBP, we use the 8-bit code of the thresholded data as per the ELBP rule, and then binarize the image with a predefined threshold value, removing the small connections on the binarized image. The proposed system is currently trained with several people's face images obtained from video sequences captured by a surveillance camera. One test set contains the disjoint images of the trained people's faces to test the accuracy and the second test set contains the images of non-trained people's faces to test the percentage of the false positives. The recognition rate among 570 images of 9 trained faces is around 94%, and the false positive rate with 2600 images of 34 non-trained faces is around 1%. Research work is progressing for the recognition of partially occluded faces as well. An appropriate weighting strategy will be applied to the different parts of the face area to achieve a better performance.

Keywords: Face recognition, feature extraction, real time processing, enhanced local binary pattern, support vector machine

1 INTRODUCTION

Face recognition algorithms worked with different face databases to test and compare the recognition results. However, most of the face databases, such as Yale[1] , CMU PIE,[2] JAFFE,[3] etc., create an environment which control most of the challenged issues, like object motion, lighting, poses, etc., to be in the same condition, and change one or two issues to test. In this paper, we record several sample videos with real life conditions to work with the proposed system which applies a refined Enhanced Local Binary Patterns (ELBP)[4] for feature extraction and Support Vector Machine (SVM)[5] for classification.

The ELBP is able to represent textural features of a face image in different lighting conditions. Instead of comparing the intensity value of every neighborhood pixel with the center pixel's intensity value directly as in Local Binary Patterns (LBP)[6–9] , the ELBP description compares the total positive distance and the total absolute distance between the neighborhood pixel and the center pixel. It then counts the number of ones in the 8-bit binary code of the thresholded neighborhood values to be the representation of the pixel under consideration. The range of the intensity value is reduced from 0∼255 to 0∼8, if a 3 by 3 neighborhood is assumed. However, the counting strategy of ELBP may merge the details of the neighborhood textural information. For instance, the final labels of 11000000 and 00000011, obtained from the ELBP rules before counting, are both 2. To retain the precise features of the input face images, we modify the ELBP algorithm by replacing the counting procedure with converting the thresholded image to a binary image. The proposed system is currently trained with several people's face images obtained from the video sequences captured by an input camera in real life conditions. It is tested with two image sets: one contains the disjoint face images of the trained faces to determine the recognition rate; the other one includes face images of several non-trained faces to investigate the percentage of the false positives caused by the non-trained ones. Single SVM creates high dimensional margins to classify the high dimensionality data into two groups. So the proposed system has multiple single SVM structures to achieve the multi-class SVM classification.

Further author information: (Send correspondence to Chen Cui)
Chen Cui: E-mail: cuic01@udayton.edu
Dr. Vijayan K. Asari: E-mail: vasari1@udayton.edu

Imaging and Multimedia Analytics in a Web and Mobile World 2015, edited by Qian Lin, Jan P. Allebach, Zhigang Fan, Proc. of SPIE-IS&T Electronic Imaging, Vol. 9408, 940806
 · doi: 10.1117/12.2083400

The main contributions of this paper are as the following:

- We propose a novel system for face recognition in real life conditions.
- We improve the ELBP algorithm to obtain a more stable feature set for the classification system.
- We increase the recognition rate, while decreasing the false positive rate.

2 RELATED WORKS

In this section, we will introduce a general structure of a face recognition system. Since we improve the ELBP algorithm for face feature extraction, some well-known feature extraction algorithms will be reviewed. A brief introduction of the proposed face recognition system will also be described.

2.1 Face Recognition System

Figure 1 shows the main parts of a face recognition system.

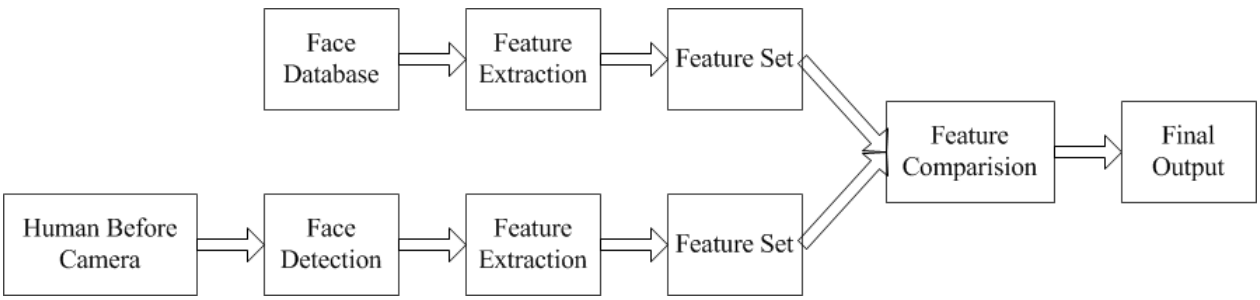


Figure 1. A face recognition system

First, the system is trained with a feature set obtained from an existing face database by some feature extraction algorithms. Then, a camera is set up for image capture. A face detection algorithm will work on the captured image to locate the particular face area. After the detection, the same feature extraction algorithm will be applied to process the detected face area. Finally, the two feature sets, one from training and the other one from the camera, will be compared to obtain the classification results.

2.2 Face Feature Extraction

Since we concentrate on the face feature extraction part, we will review some well-know feature extraction algorithms in this section. Also, these algorithms will be compared with the proposed algorithm in Experimental Results (Section 4). For the face detection, we use Haar-Cascade[10–12] face detection to tackle.

- Local Binary Patterns (LBP)

  LBP describes a pixel value by calculating the relationship between the pixel and its related neighbor pixels, obtaining a less lighting-affected feature set. This model was proposed in 1990[7,8] , first described in 1994[9,13] . Figure 2 shows the main idea of LBP.

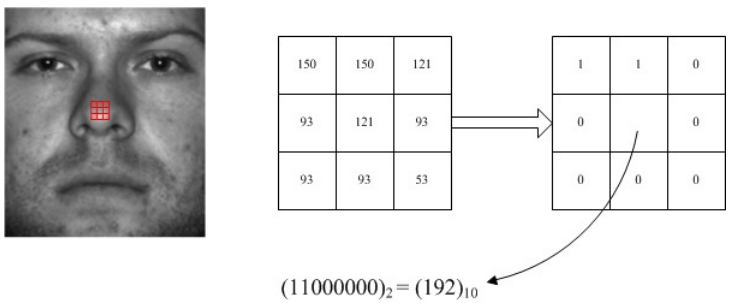


Figure 2. LBP descriptor

The face image in Figure 2 comes from Yale Database B.[14] Consider the center pixel of the red block needs to be computed. First, the differences between the center pixel and its neighbor pixels are calculated. Then, if the difference is larger than zero, a "1" will be assigned to that neighbor pixel, otherwise, a "0" will be assigned. The new pixel value will be represented by the 8-bit binary code taken from all the neighbor pixels. Figure 3 shows an example after the process.

Figure 3. The comparison between the original image and the LBP image

Even though LBP reduces the effect of lighting, the outcome features may not be obvious in an extreme dark/bright environment.

- Histogram of Oriented Gradients (HoG)[15]

  HoG is widely used for object detection in image processing.[16,17] It calculates the magnitude and the orientation of the gradient of each sub-block in an image. The main steps of HoG are described as the following:

  - Compute centered horizontal and vertical gradients of each pixel
  - Compute the magnitude and orientation from the gradients
  - Divide the image into several equal cells
  - Assign the gradients into 9 bins based on their orientations
  - Normalize the histogram of each cell with its three nearest neighbors (four cells form a block)
  - Concatenate all the histograms from all the blocks to present the entire image

  Figure 4 shows an example of the HoG descriptor. The front view human body image is from the MIT pedestrian database.[18]

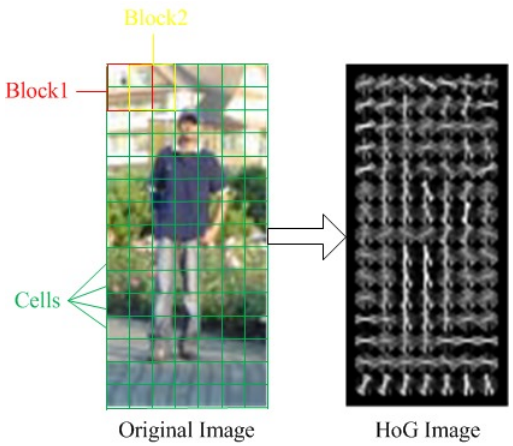


Figure 4. An example of the HoG descriptor

The green lines divide the entire image into 105 equal cells, and there are 9 bins whose angles range from 0 to 180 degrees in each cell. The normalization is done in each big block which is constituted by 4 green cells, i.e. the red and yellow block in the image. After the normalization, the entire image will be represented by concatenating all the histograms from the green cells, as the HoG image shown in Figure 4.

HoG also has been used in face recognition,[19] however, it requires high quality face images to be involved. The details of the face features cannot be obtained with the low quality images since the face features are more complex than objects.

- Sobel Operator for Edge Detection[20, 21]

  Sobel operator is a 2-D spatial gradient measurement of an image to enhance areas with high spatial frequency corresponding to the edges.[22] There are two kernels of the Sobel operator, and the second kernel is the 90 degree's rotation of the first one. Figure 5 shows the 3 by 3 kernels of the operator.

-1	0	+1
-2	0	+2
-1	0	+1

G_x

+1	+2	+1
0	0	0
-1	-2	+1

G_y

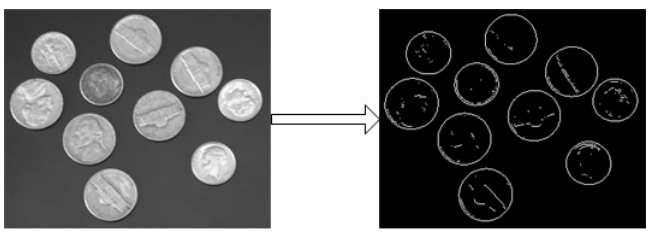
Figure 5. The convolution kernels of Sobel

The two kernels response the best to the vertical and horizontal edges of an image. They are applied separately on the image to obtain the gradient component in each direction (G_x and G_y). Then, the absolute magnitude ($|G|$) and the orientation of the gradient (θ) are calculated by Equation 1 and Equation 2.

$$|G| = \sqrt{{G_x}^2 + {G_y}^2}, \tag{1}$$

$$\theta = arctan(G_x/G_y). \tag{2}$$

Figure 6 shows an example after applying the Sobel operator on an image.



Original Image Sobel Image

Figure 6. An example of an image after being applied the Sobel operator

The strong edges are detected after being applied the Sobel operator, but some details are lost. Some other edge detection algorithms may maintain softer edges as well, i.e. Canny edge detector,[23–25] so it may cause problems if the input image has a complex background.

As we mentioned above, different features have different limitations so that we propose a face feature algorithm based on ELBP which is an enhanced version from the original LBP. The proposed method obtains more preferable features than the method reviewed. In next section, we will discuss more details of the proposed method, as well as the advantages.

3 METHODOLOGY

Figure 7 illustrates the frame work of the proposed system. In the training part, we apply the modified ELBP on the input images, training the SVM with the modified ELBP images to obtain the SVM structures. In the testing part, a camera captures images in real time, and Haar-Cascade face detection is applied to detect the face area. The detected face area is also processed by the modified ELBP descriptor. Finally, the processed face images are predicted by the training SVM structures.

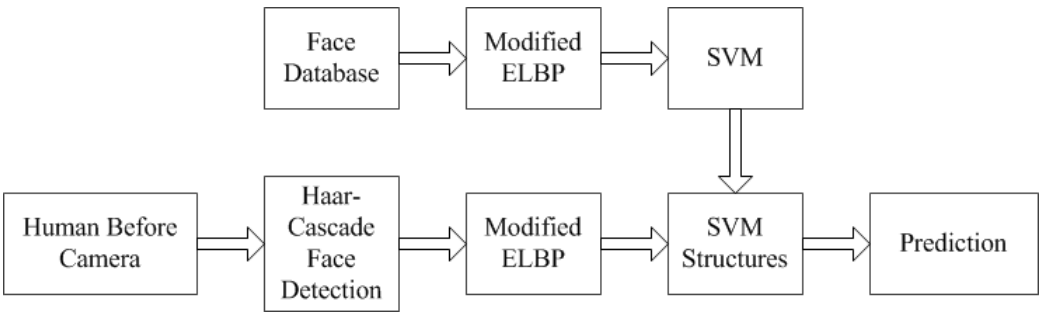


Figure 7. The framework of the proposed system

The proposed feature extraction method is a refined version of ELBP. It extracts more obvious face features than the former one. Before we step into the modified ELBP, a brief introduction will be discussed here. As we described in the Introduction and Related Works, ELBP not only considers the properties of the pixels in the neighborhood but also the relationship within the local sub-regions. However, the LBP based features are sensitive to noises, even smaller textural information in the RGB/gray-scale images will appear, such as the skin textures. Also, the face image will be noisy if the input signal of the source camera is affected.

3.1 ELBP

Figure 8 and Figure 9 illustrate the local sub-region based feature extraction of ELBP.

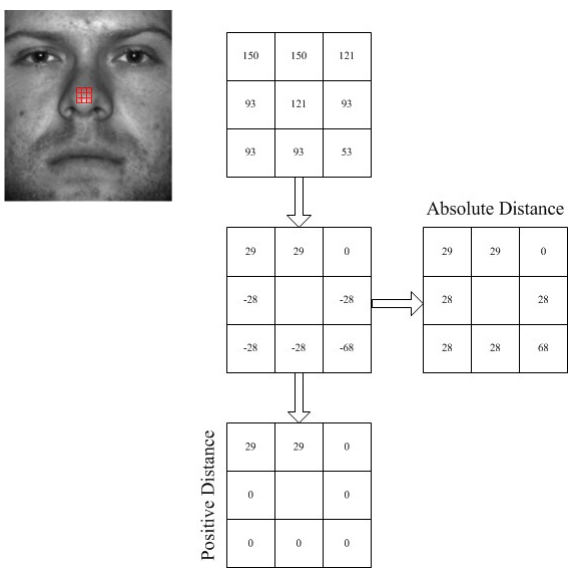


Figure 8. The illustration of ELBP (part 1)

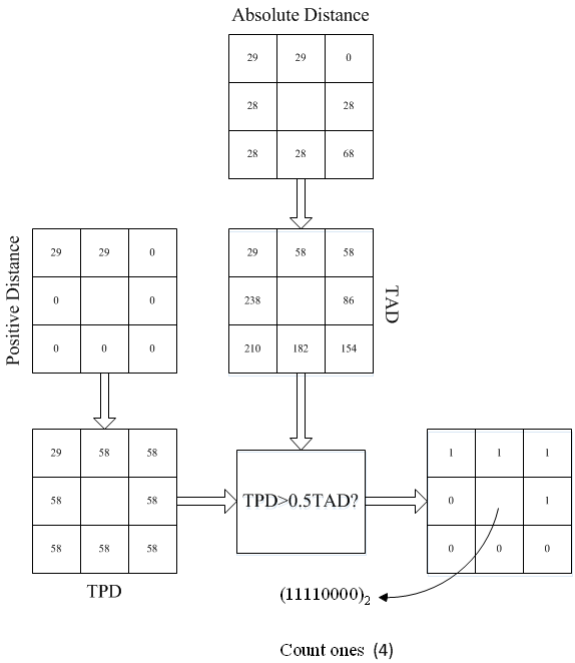


Figure 9. The illustration of ELBP (part 2)

First, the distances between the pixel and its related neighbor pixels need to be calculated. From the distances, the absolute and positive distances are obtained. Then, the accumulations of the absolute and positive distance (TAD and TPD) are operated separately. After TAD and TPD are obtained, a comparison between them is taken to calculate the final 8-bit binary code. Finally, the number of 1's need to be counted to be the new representation. Equation 3 to Equation 6 show the derivation of ELBP.

$$TAD_q = \sum_{i=1}^{q} |g_i - g_c|, \tag{3}$$

$$TPD_q = \sum_{g_k \geq g_c} (g_k - g_c), \tag{4}$$

$$s(TPD_q - \frac{TAD_q}{2}) = \begin{cases} 1 & \text{if } TPD_q - \frac{TAD_q}{2} > 0 \\ 0 & \text{if } TPD_q - \frac{TAD_q}{2} \leq 0 \end{cases}, \tag{5}$$

$$ELBP(x_c, y_c) = \sum_{q=0}^{8-1} s(TPD_q - \frac{TAD_q}{2}), \tag{6}$$

where $q = 1, 2, 3, ..., 8$, q is the number of neighbor pixels taken into consideration, k is less than or equal to q, x_c and y_c are the coordinates of the processed pixel, and $ELBP(x_c, y_c)$ is the new pixel value. Figure 10 shows the comparison among the original, LBP and ELBP images.

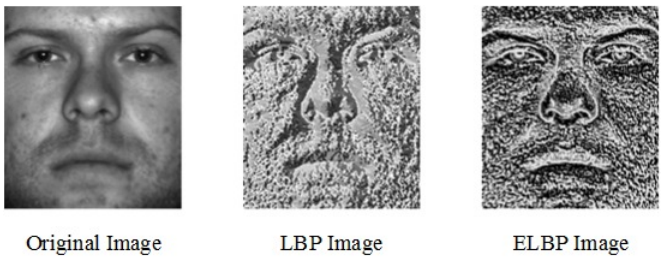


Figure 10. The comparison among the original, LBP and ELBP images

From Figure 10, the ELBP descriptor generates a clearer feature than LBP, and the range of the intensity level is highly reduced, from 0∼58 (uniform patterns[26]) to 0∼8. Even though the obtained features are improved, the noises from the skin texture still exist. The general idea of the proposed method is to lower the effect of the skin texture and speed up for the real time processing.

3.2 Modified ELBP

The modified ELBP followed most of the calculation steps from ELBP, and only removed the counting strategy. In other words, before the modification, the range of the intensity level is from 0 to 255. Figure 11 is the flow chart of the modified ELBP.

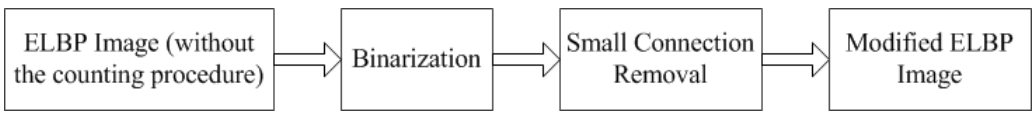


Figure 11. The flow chart of the modified ELBP

First, we convert the ELBP image to a binary image with 0.9 level threshold so that all the pixel values are either '1' or '0'. Figure 12 shows an example of the conversion.

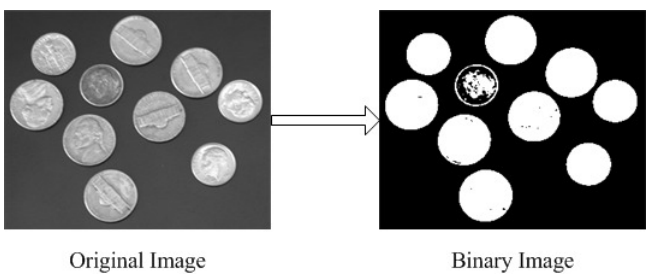


Figure 12. An example of image binarization

The binarization[27] for the grayscale images may lose some details of the features, however, it helps to reduce noises from the ELBP images.

After the binarization, we remove the small connections[28–30] in the image. This is because the big connections are caused by the strong edges in the face area, such as the contours of eyes, nose and mouth, whereas the small connections are mostly caused by the skin textures. Some noises from the skin textures will be removed when the small connections are cleared. In our case, we delete the area whose pixel connections are less than 10. Figure 13 illustrates this process.

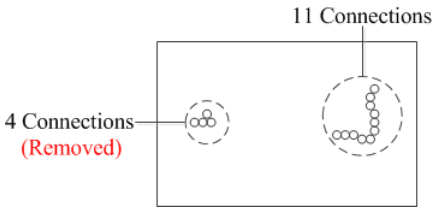


Figure 13. Illustration of the connections that need to be removed

Now, the final representation from the modified ELBP descriptor is obtained. Figure 14 indicates the differences among the original ELBP image, the image after binarization and the modified ELBP image.

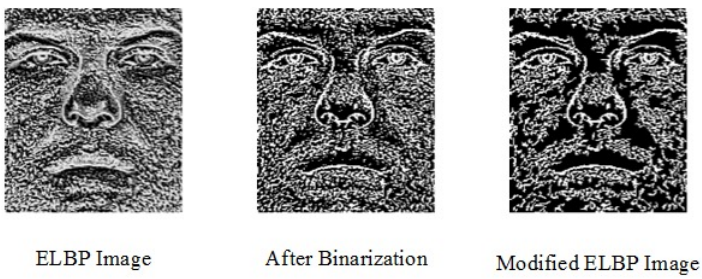


Figure 14. The comparison among the original ELBP image, the image after binarization and the modified ELBP image

Compared with the ELBP image, the modified one reduces some skin textures and enhances the edges. Figure 15 shows the comparison between the modified ELBP images and the Sobel Operators reviewed in Related Works (Section 2). Even though we changed the threshold of the Sobel Operators to try to obtain the best image, the outcome results are worse than our proposed method. In next section, more comparisons will indicate the effectiveness.

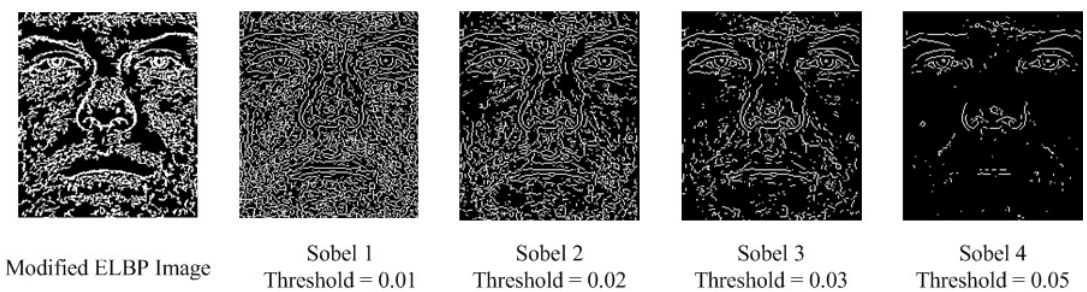


Figure 15. The comparison between the modified ELBP image and the Sobel Operator processing image

4 EXPERIMENTAL RESULTS

In order to test the effectiveness of the proposed system, we compared the performance with the algorithms mentioned in Section 2. We recorded several videos in real life conditions to do both training and testing. Nine people's face images obtained from the videos to be the training data, and the face images obtained from the other disjoint videos to be the testing data. The testing data includes the 9 trained faces and 34 non-trained faces. The video record environment is shown in Figure 16. The camera is faced to the entrance, and people enter the main door, then walk into the laboratory. There are around 20 images per person to train the system, and these images are selected manually from the obtained face images. For instance, there are 75 face images obtained from video 1 but only 18 images are selected to train the system. Figure 17 shows a group of examples of the obtained face images(this database is a private one which is created by the authors, and the face images shown here are also from one of the authors). It is obviously that the sizes of the obtained face images are different, as well as

the lighting conditions. Because of the movement or the position of the focal point, some of the images may be blurred. These difficulties may cause more false positives or false negatives. In order to increase the recognition rate, we apply a tracking strategy to the proposed system. Consider there are two face images extracted from two neighbor frames in a same video sequence separately, if the distance from the coordinates of the two images is in a range(usually the length of the face image), these two images will be considered from the same person. Then, if 20 percent images from the same person are classified to the same group, we will assign the following image of that person to the classified group. All the other algorithms are also applied the same tracking strategy. Besides the 9 trained classes, we create one more class for the non-trained faces. If a face image is not assigned to any of the 9 trained classes, it will be assigned to the 10th class – "Unknown".

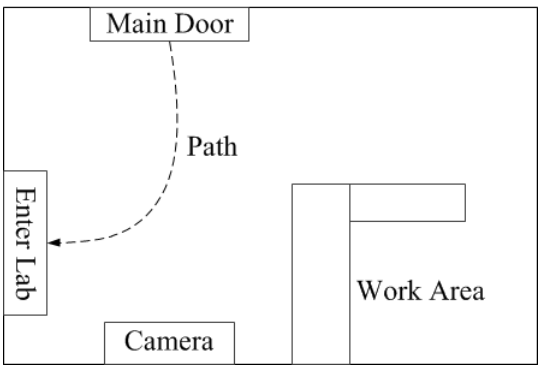


Figure 16. Illustration of the data collection environment

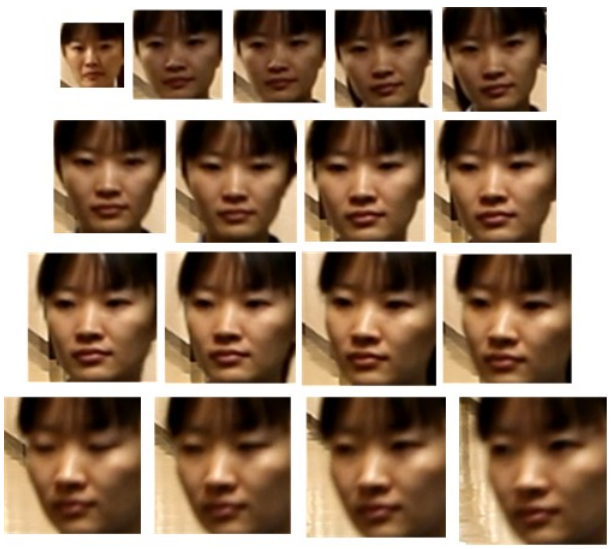


Figure 17. A group of examples obtained from the recorded videos

Table 1 and Table 2 indicate the number of face images per person to train, the number of the testing images per person of each trained person (the training and testing images are totally disjoint), the number of the correct recognized images with different algorithms and the recognition rate. From Table 2, the average recognition rate from LBP is the highest, and the proposed MELBP is the third highest. However, the face recognition not only considers the recognition rate but also the false positive rate. Therefor using only the recognition rate does not represent the final result.

Person	Training Images	Testing Images	Correct Recognized images (Rate)		
			LBP	HoG	Sobel
1	19	68	68(100%)	68(100%)	0(0%)
2	18	72	72(100%)	72(100%)	34(47%)
3	18	42	42(100%)	42(100%)	35(83%)
4	19	63	63(100%)	63(100%)	42(67%)
5	20	64	64(100%)	64(100%)	56(88%)
6	19	75	75(100%)	66(88%)	0(0%)
7	18	93	93(100%)	93(100%)	0((0%))
8	17	38	38(100%)	38(100%)	35(92%)
9	18	60	60(100%)	60(100%)	27(45%)
Average	18.44	63.89	63.89(100%)	62.89(98%)	25.44(40%)

Table 1. Recognition results among different algorithms (part 1)

Person	Training Images	Testing Images	Correct Recognized images (Rate)	
			ELBP	**MELBP**
1	19	68	67(99%)	55(81%)
2	18	72	72(100%)	72(100%)
3	18	42	40(95%)	41(98%)
4	19	63	63(100%)	63(100%)
5	20	64	59(92%)	64(100%)
6	19	75	75(100%)	75(100%)
7	18	93	14(15%)	73(78%)
8	17	38	38(100%)	38(100%)
9	18	60	60(100%)	60(100%)
Average	18.44	63.89	54.22(85%)	60.22(94%)

Table 2. Recognition results among different algorithms (part 2)

Table 3 and Table 4 show the false recognition with 34 non-trained people's face images. These 34 people are different from the 9 people mentioned above. Even though LBP and HoG obtain high recognition results, the false positive rates are too high. The face recognition system is applied to real life conditions to help send out warnings if person of interest appears. However, LBP and HoG may cause too many false warnings. Our test is limited by the number of people so that the results may be worse if more and more people are involved. Sobel operator obtains the lowest average false positive rate but the recognition rate is only 40%. Compare ELBP with the proposed MELBP, the average false positive rates are 0.3% and 1.1%, whereas, the average recognition rates are 85% and 94%. Although the average false positive rate rises a little bit, the average recognition rate is improved much more. The two tests with the trained and non-trained people's face images show the effectiveness of the proposed system.

Person	Testing Images	False Recognized images (Rate)				
		LBP	HoG	Sobel	ELBP	**MELBP**
10	24	0(0%)	0(0%)	0(0%)	0(0%)	0(0%)
11	42	0(0%)	0(0%)	0(0%)	0(0%)	0(0%)
12	55	0(0%)	21(38%)	0(0%)	0(0%)	0(0%)
13	64	0(0%)	15(23%)	0(0%)	0(0%)	0(0%)
14	71	0(0%)	63(89%)	0(0%)	0(0%)	0(0%)
15	124	124(100%)	0(0%)	0(0%)	0(0%)	0(0%)
16	48	0(0%)	0(0%)	0(0%)	0(0%)	0(0%)
17	9	9(100%)	0(0%)	0(0%)	0(0%)	0(0%)

Table 3. False recognition results among different algorithms (part 1)

Person	Testing Images	False Recognized images (Rate)				
		LBP	HoG	Sobel	ELBP	**MELBP**
18	27	0(0%)	0(0%)	0(0%)	0(0%)	0(0%)
19	33	23(70%)	2(6%)	0(0%)	0(0%)	0(0%)
20	54	0(100%)	3(6%)	0(0%)	0(0%)	0(0%)
21	41	0(0%)	6(15%)	0(0%)	0(0%)	0(0%)
22	24	0(0%)	0(0%)	0(0%)	0(0%)	0(0%)
23	89	70(79%)	0(0%)	0(0%)	7(8%)	24(27%)
24	130	0(0%)	21(16%)	0(0%)	0(0%)	0(0%)
25	61	11(18%)	0(0%)	0(0%)	0(0%)	0(0%)
26	136	2(1%)	0(0%)	0(0%)	0(0%)	0(0%)
27	114	0(0%)	22(19%)	0(0%)	0(0%)	0(0%)
28	45	3(7%)	0(0%)	0(0%)	0(0%)	0(0%)
29	25	3(12%)	0(0%)	0(0%)	0(0%)	0(0%)
30	132	0(0%)	9(7%)	0(0%)	0(0%)	0(0%)
31	78	2(3%)	0(0%)	0(0%)	0(0%)	0(0%)
32	81	0(0%)	0(0%)	0(0%)	0(0%)	0(0%)
33	62	57(92%)	0(0%)	0(0%)	0(0%)	0(0%)
34	82	0(0%)	0(0%)	0(0%)	0(0%)	0(0%)
35	180	0(0%)	1(1%)	0(0%)	0(0%))	0(0%)
36	147	0(0%)	25(17%)	0(0%)	0(0%)	0(0%)
37	69	0(0%)	0(0%)	0(0%)	0(0%)	0(0%)
38	118	0(0%)	1(1%)	0(0%)	0(0%)	0(0%)
39	103	0(0%)	35(34%)	0(0%)	0(0%)	4(4%)
40	73	0(0%)	0(0%)	0(0%)	0(0%)	0(0%)
41	100	0(0%)	0(0%)	0(0%)	0(0%)	0(0%)
42	93	3(3%)	0(0%)	0(0%)	0(0%)	0(0%)
43	52	0(0%)	5(10%)	0(0%)	0(0%)	0(0%)
Average	76.06	9.03(11.9%)	6.74(8.9%)	0(0%)	0.21(0.3%)	0.82(1.1%)

Table 4. False recognition results among different algorithms (part 2)

5 CONCLUSIONS

In this paper, we proposed a face feature extraction algorithm based on the ELBP descriptor to process the real time face recognition in real life conditions. Several issues, such as lighting, expression variation, moving motion, etc., may appear in real life conditions at the same time. In order to obtain a stable feature, we refine the original ELBP features with the binarization and the small connection removal. The binarization of the ELBP image maximizes the differences between the strong edges on the face and the skin textures. Moreover, it minimizes the differences of the pixels which have similar intensity values in the grayscale image. The small connection removal reduces the noises created by the ELBP descriptor caused by the skin textures, so the face features, especially the contours of sensory organs, are more strengthened.

The comparisons among the LBP, ELBP, Sobel Operator and modified ELBP images show the effectiveness of the proposed method. Even though, the modified ELBP descriptor is not particular for the edge detection, it can strengthen the edges without losing too much information. However, the edge detector, Sobel Operator, causes problems when the threshold is small and losing too much information when the threshold is large. Although we include two more procedures to the original ELBP to improve the accuracy, the efficiency is not effected too much. The image binarization and small connection removal are low cost processing. The original ELBP needs around 0.0025 second to process, and the modified ELBP needs around 0.1067 second to process. These processing times are from MATLAB, and the differences will be much smaller if we process them in C++.

The research work is improving the face detection and feature extraction in extreme lighting conditions. A threshold needs to be applied to the image scene according to the variance. The modified ELBP can also be implemented for object detection or other recognition systems, but more experiments need to be conducted.

REFERENCES

[1] Bellhumer, P. N., Hespanha, J., and Kriegman, D., "Eigenfaces vs. fisherfaces: recognition using class specific linear projection," *IEEE Trans., Pattern Analysis and Machine Intelligence* **17**(7), 711–720 (1997).

[2] Sim, T., Baker, S., and Bsat, M., "The cmu pose, illumination, and expression (pie) database," in [*Proceedings of the Fifth IEEE International Conference on Automatic Face and Gesture Recognition*], IEEE Computer Society (2002).

[3] Lyons, M., Akamatsu, S., Kamachi, M., and Gyoba, J., "Coding facial expressions with gabor wavelets," in [*Proceedings of the Third IEEE International Conference on Automatic Face and Gesture Recognition*], 200–205, IEEE Computer Society (1998).

[4] Cui, C. and Asari, V., "Adaptive weighted local textural features for illumination, expression, and occlusion invariant face recognition," in [*IS&T/SPIE International Conference on Electronic Imaging: Imaging and Multimedia Analytics in a Web and Mobile World*], (2014).

[5] Cortes, C. and Vapnik, V., "Support-vector networks," *Machine Learning* **20**, 273–297 (1995).

[6] Ahonen, T., Hadid, A., and Pietikinen, M., "Face description with local binary patterns: application to face recognition," *IEEE Trans, Pattern Analysis and Machine Intelligence* **28**, 2037–2041 (2006).

[7] He, D. and Wang, L., "Texture unit, texture spectrum, and texture analysis," *IEEE Trans, Geoscience and Remote Sensing* **28**, 509–512 (1990).

[8] Wang, L. and He, D., "Texture classification using texture spectrum," *Pattern Recognition* **23**, 905–910 (1990).

[9] Ojala, T., Pietikinen, M., and Harwood, D., "A comparative study of texture measures with classification based on feature distributions," *Pattern Recognition* **29**, 51–59 (1996).

[10] Viola, P. and Jones, M., "Rapid object detection using a boosted cascade of simple features," *Computer Vision and Pattern Recognition* **1**, 511–518 (2001).

[11] Papageorgiou, C. and Oren, M., "A general framework for object detection," *International Conference on Computer Vision* , 555–562 (1998).

[12] Lienhart, R. and Maydt, J., "An extended set of haar-like features for rapid object detection," *Image Processing International Conference* **1**, 900–903 (2002).

[13] Ojala, T., Pietikainen, M., and Harwood, D., "Performance evaluation of texture measures with classification based on kullback discrimination of distributions," *Pattern Recognition* **1**, 582–585 (1994).

[14] Georghiades, A. S., Belhumeur, P. N., and Kriegman, D. J., "From few to many: illumination cone models for face recognition under variable lighting and pose," *IEEE Trans., Pattern Analysis and Machine Intelligence* **23**(6), 643–660 (2001).

[15] Dalal, N. and Triggs, B., "Histograms of oriented gradients for human detection," *Computer Vision and Pattern Recognition* **1**, 886–893 (2005).

[16] N., D., B., T., and C., S., "Human detection using oriented histograms of flow and appearance," *Computer Vision* , 428–441 (2006).

[17] Suard, F., Rakotomamonjy, A., Bensrhair, A., and Broggi, A., "Pedestrian detection using infrared images and histograms of oriented gradients," *IEEE Intelligent Vehicles Symposium* , 206–212 (2006).

[18] Oren, M., Papageorgiou, C., Sinha, P., Osuna, E., and Poggio, T., "Pedestrian detection using wavelet templates," *Computer Vision and Pattern Recognition* , 193–199 (1997).

[19] Dniz, O., Bueno, G., Salido, J., and la Torre, F. D., "Face recognition using histograms of oriented gradients," in [*Pattern Recognition Letters*], **32**, 1598–1603 (2011).

[20] Kanopoulos, N., Vasanthavada, N., and Baker, R. L., "Design of an image edge detection filter using the sobel operator," *IEEE Journal, Solid-State Circuits* **23**, 358–367 (1988).

[21] Qu, Y.-D., Cui, C.-S., Chen, S.-B., and Li, J.-Q., "A fast subpixel edge detection method using sobel-zernike moments operator," *Image and Vision Computing* **23**(1), 11–17 (2005).

[22] Davies, E., "Machine vision: theory, algorithms, practicalities," (1997).

[23] Canny, J., "A computational approach to edge detection," *IEEE Trans, Pattern Analysis and Machine Intelligence* , 679–698 (1986).

[24] Harris, C. and Stephens, M., "A combined corner and edge detector," in [*Alvey vision conference*], **15**, 50 (1988).

[25] Ding, L. and Goshtasby, A., "On the canny edge detector," *Pattern Recognition* **34**, 721–725 (2001).

[26] Ojala, T., Pietika, M., and Ma, T., "Multiresolution gray-scale and rotation invariant texture classification with local binary patterns," *IEEE Trans, Pattern Analysis and Machine Intelligence* **24**, 971–987 (2002).

[27] Sauvola, J. and Pietikinen, M., "Adaptive document image binarization," *Pattern Recognition* **33**, 225–236 (2000).
[28] ERDdS, P. and R&WI, A., "On random graphs i," *Publ. Math. Debrecen* **6**, 290–297 (1959).
[29] Samet, H., "Connected component labeling using quadtrees," *Journal of the ACM* **28**(3), 487–501 (1981).
[30] Dillencourt, M., Samet, H., and Tamminen, M., "A general approach to connected-component labeling for arbitrary image representations," *Journal of the ACM* **39**(2), 253–280 (1992).