FACIAL RECOGNITION BASED ON THE LOCAL BINARY PATTERNS MECHANISM

Alexandru Blanda[1]

ABSTRACT

This work presents a method of facial recognition, based on Local Binary Models. The idea of using this algorithm was induced by the fact that human faces can be seen as a composition of microstructures, which can be easily described by this algorithm. I will discuss about the process of facial recognition at a theoretical level and the implementation of an application, written in MATLAB, which will help me run some experiments on a number of images, using a standard database.

KEYWORDS: Local Binary Patterns (LBP), classification, characteristics extraction, facial recognition, histograms.

1. INTRODUCTION

General Presentation of LBP

Most of the facial recognition algorithms are designed for working with images containing well aligned faces, good luminosity and frontal position. Yet, in many available applications, these conditions are not always possible. For example, we remind of surveillance systems, human-robot interaction or automatic labeling.

Here, using local aspect descriptors like Gabor jets, SURF, SIFT, HOG and the histograms of LBP, became more frequent. These algorithms are more efficient against occlusions, the variation of face expressions and the variation of face position than the traditional holistic algorithms. [1]

Facial Recognition

For the facial recognition process, there are more methods of extracting the most useful characteristics from the processed images. One of these methods is the method of Local Binary Patterns. In the last years, this algorithm presented a high interest because of the calculation simplicity, its stability against position variation, grey tones changes, luminosity, but mostly due to its mode of codification and its ability in describing the texture of an image. Thus, it started to be used in various image analysis applications such is the field of facial recognition. However, the algorithm has some limitations regarding noise sensibility and lack of rotation invariance, leading to many extensions and proposals in the purpose of removing these unwanted factors. [4]

Base principles of LBP

Using the automatic facial recognition based on Local Binary Patterns, the description of a digital image texture and form becomes possible. This method starts by dividing an image in multiple regions, from which the characteristics of the image are to be extracted.

[1] Engineer , blanda.alexandru@gmail.com, ”Politehnica” University of Bucharest, 060042 Bucharest, Romania

The characteristics stand in binary patterns that describe the surroundings of the pixels from the regions. After obtaining the characteristics, they are to be concatenated in a single histogram, standing as a representation of the whole image.

Later, the image can be compared with the images from the database, measuring the similarities (or the differences) between histograms. According to many studies, automatic recognition based on LBP gives very good results, regarding the speed and the performance.

Due to the way of describing the texture and the form of the image, this method successfully confronts with factors that could affect the quality of the image: various conditions of luminosity face expressions, image rotation or aging. [3]

2. CALCULATION MODE. LBP CODING

The algorithm based on LBP is, in fact, a binary code of an image, defined by pixels, that offers information about the local surroundings of the pixel. LBP operator works with all eight neighbors of a pixel, using the value of the central pixel as a threshold. If a nearby pixel has a bigger value than the central pixel (or the same value of grey), then the value in binary will be equal to 1; if the pixel has a smaller value, then its binary value will be 0. The LBP code for the central pixel obtained by concatenation of the eight values of 0 and 1 as we can see in the following figure: [2]

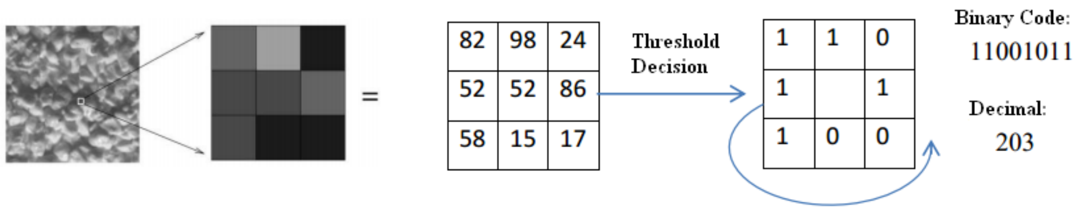

Figure 1. LBP Coding

3. THE LBP HISTOGRAMS. HISTOGRAM CALCULATION

The original Local Binary Pattern operator labels the pixels of an image through a threshold decision in a surrounding of 3x3: each neighbor point is compared with the central value and, considering the result, a binary number is created. This number stands for the Code of the Local Binary Pattern. Then the operator extended in order to be used in different sizes of the radius, being redefined for the representation of the most important microstructures presented in the uniform local binary patterns. [2]

Histogram of these patterns forms a vector, which is in fact the representation of the image texture. These histograms can be used in measuring the similarity between the images, calculating the distance between them.

Once the Local Binary Pattern of every pixel is calculated, the vector associated to the image can be built. For an efficient representation of the face, the image is at first divided in kxk regions. In figure no 2, an image containing a face is divided in 64 regions. For each region, there is built a histogram with all labels. It means that every stack of a histogram represents a pattern and contains the number of apparitions of the pattern in the region. The associated vector is built after this by concatenating the regional histogram into a global histogram.[3]

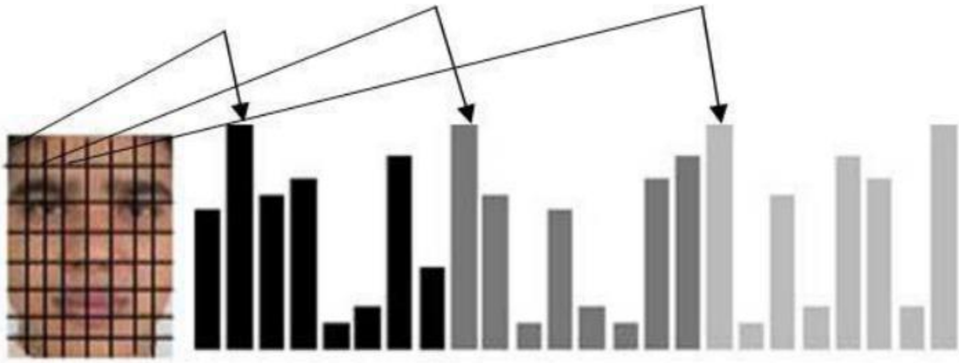

Figure 2. Region Histograms

For every region, all the non-uniform patterns (with more than two transitions) are marked with a unique label. This means that every regional histogram contains P(P-1)+3 stacks; P(P-1) stacks for patterns with two transitions; 2 stacks for patterns with zero transitions; 1 stack for all non-uniform patterns. Overall, the vector associated to an image contains k x k(P (P-1)+3) stacks. Therefore, for an image divided in 64 regions and 8 poll points on the circle, the associated vector has a dimension of 2891 stacks. [2]

In order to compare two face images, a test image (S) and a model image (M), must be measured the difference between the characteristic vectors. This can be done through various methods of measuring the dissimilarities between histograms:

- Histogram intersection:

$$D(S,M)=\sum_{j=1}^{k^2}(\sum_{i=1}^{P(P-1)+3}\min(S_{i,j},M_{i,j}))$$

- Log-likelihood statistic:

$$L(S,M)=\sum_{j=1}^{k^2}(\sum_{i=1}^{P(P-1)+3}S_{i,j}\log M_{i,j})$$

- Chi square statistic:

$$X^2(S,M)=\sum_{j=1}^{k^2}(\sum_{i=1}^{P(P-1)+3}\frac{(S_{i,j}-M_{i,j})^2}{(S_{i,j}-M_{i,j})})$$

In these equations $S_{i,j}$ and $M_{i,j}$ stand for the dimensions of a stack I, from the region j (the number of apparitions of the pattern L(i) in the region j).

4. IMPLEMENTATION OF THE APPLICATION

The following figure represents the general architecture of the implemented facial recognition application:

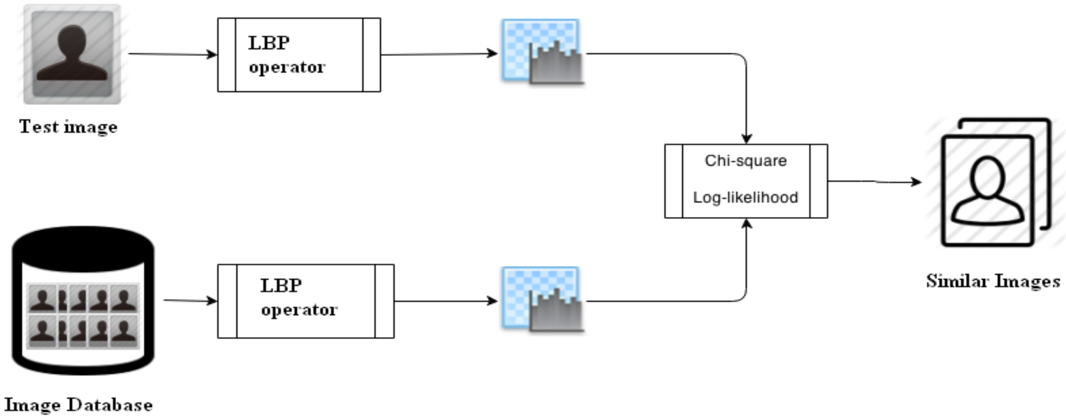

Figure 3. Facial Recognition Application

The main functionalities implemented are the following:

- LBP operator calculation with radius and variable points number;
- Determination of the similarities between images using all the following methods:
  - Histogram intersection
  - Log-likelihood statistic
  - Chi-square
- Different shares for different zones of the face.

The application developed in MATLAB consists in a main program, which appeals to a number of two base functions in the process of facial recognition: improved LBP and detect_image. The first function is the one that applies the proper LBP algorithm, with basic math formulas, and the second realizes the detection of the image and the two images histograms comparison through Chi-square function (or equivalent formulas).

The application structure is the following:

- Reading of the image/images from the database
- Appealing to the function improve LBP
- Storing in a .mat type file of the characteristic vectors that leading to the LBP histogram construction for every image from the database
- Reading the test image, applying the LBP algorithm again
- Appealing to the function detect_image. Comparison of two histograms of the images with the chi-square function and selecting the first n smallest differences between the two images
- Displaying the results

Finally, the program will display the results, regarding the requested parameters. In the following figure, we selected the first 3 smallest differences between the images, in a 2x2 picture. The image on the position (2,1) is the test image, and the others: (2,2), (2,3), (2,4) are the images found in the database. As we can see, the first selection, looking at the smallest differences between the images, is represented by the image (2,2), because it is identical with the test image, so we have 100% similarity. The success rate in this case is 100%, because the program recognized all the images of the selected person.

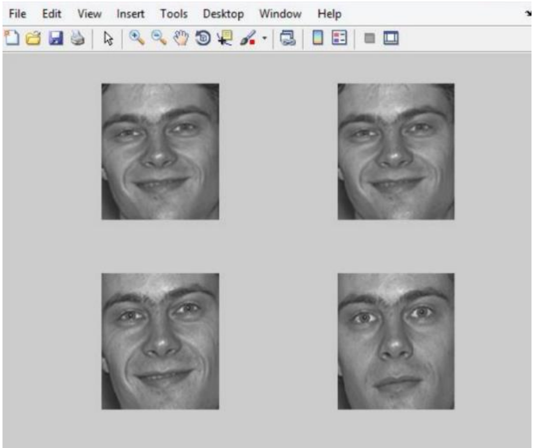

Figure 4. Program Interface and result

5. RESULTS, PERFORMANCES AND OPTIMIZATIONS OF THE LBP ALGORITHM

After testing the application on a set of images, we noticed differences regarding its performances. Some images were successfully processed, but in other images, the face was not correctly detected.

Testing images that contain different facial expressions

We tested images of different persons, in order to notice the success rate of the algorithm when there are different facial expressions. Such a test can be observed in the following figures:

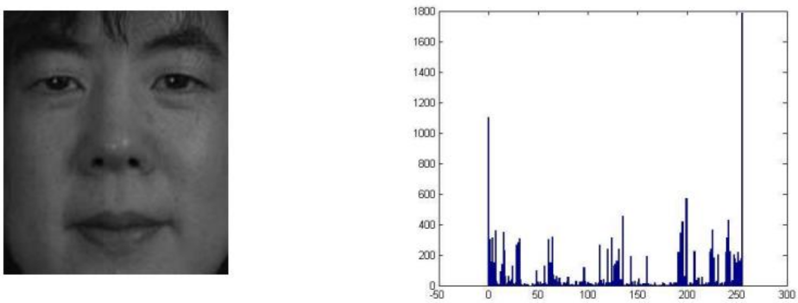

Figure 5. Neutral lest image

The previous figure represents the test image used for identifying similar images from the database. The result is displayed in a 1x2 picture, where (1,1) is the test image and (1,2) the recognized image after applying the LBP algorithm, as we can see in the following figure:

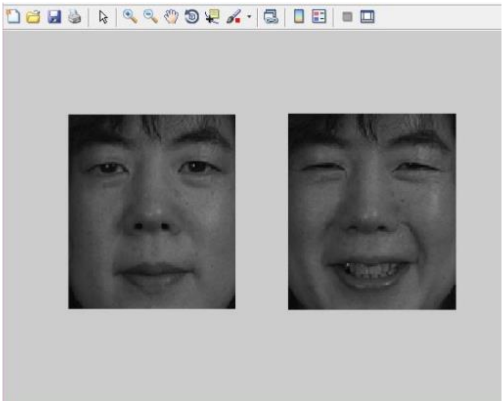

Figure 6. Correct Recognition even in various facial expressions

After performing several tests, the success rate in similar images detection based on a test image exceeds 67%. This result can also be noticed in the following figures, where the first image from every matrix represents the starting test image, and the rest of the image are found by the application.

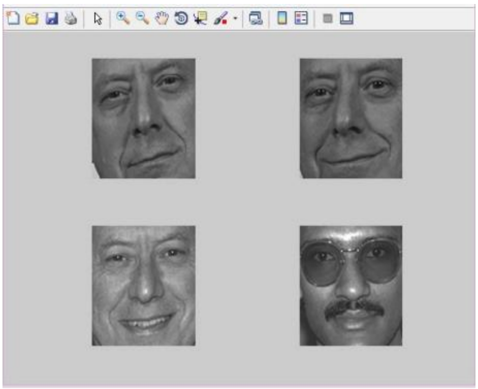

Figure 7. Some wrong results at the end of a good result

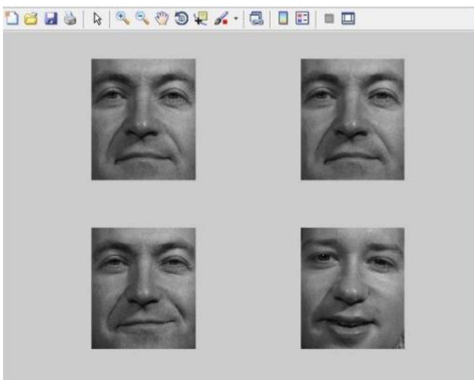

Figure 8. Correct results

Testing images with different facial accessories

In the purpose of studying the performance of the algorithm in cases where facial accessories may appear and vary, we performed a number of tests for images with persons that may wear glasses, beard etc. In the following figure, we can observe such a test. We chose that the results must be displayed in a 1x2 picture; test image is (1,1) and the result is (1,2). For this type of tests and for the using database, the success rate of the application was 100%.

Figure 9. Example of recognition when wearing facial accessories

Results using shares for different face zones

We tried to realize an optimization of the application, introducing different shares. In the characteristic vectors similarity calculation, for different face zones (eyes, mouth). The results are quite promising. In some instances, the share method offered better results than the classic method, but in other cases, the results were identical. The calculation method using shares did not yet offer worse results in none of the tests we performed.

Comparative results between the similarity calculation formulas

The only formula that offered acceptable results is the Chi-square formula. This can be noticed in the following figures. Therefore, in Fig. 10 we can observe the results of a test using the histograms intersection formula, and in Fig. 11, the results of a test performed using the log-likelihood statistic formula.

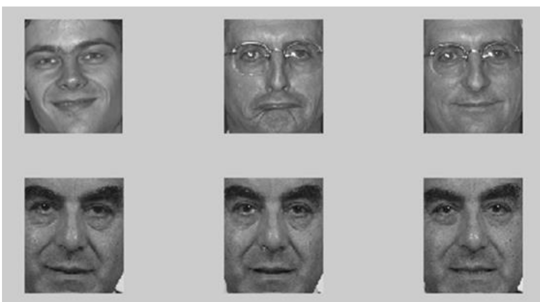

Figure 10. Recognition using Histogram Intersection

Figure 11. Recognition using Log-Likelihood

6. CONCLUSIONS

The LBP operator was used in different applications, like image texture classification, image reconstruction etc.

The performed implementation has a success rate of 70% in different testing scenarios. We can consider, regarding the experimented results, that the method is more efficient when a bigger radius is used in the calculation of LBP operator and the fact that Chi-square is the most efficient formula in order to estimate the image similarity.

We performed a number of tests for different images in the purpose of determining the performance of the algorithm based on Local Binary Patterns in conditions of luminosity, facial expressions or facial accessories variation. The experimental results demonstrates that this algorithm is quite efficient and stable, its success rate differing from one scenario to another.

7. FUTURE WORK

The quality of the recognition can be improved in multiple ways. One possibility is to try to rotate the image, in order to be aligned to the test images. This approach is used in text deskew, but it could also be implemented here, after selecting regions of interest (eyes, eyebrows, nose, mouth). These usually tend to be aligned horizontally and vertically, thus approaches from document deskew are very appropriate. [5]

For color images, the best option is to verify each channel individually, but if speed is required, an algorithm for decolorizing the image can improve the results over a simple conversion to greyscale, as it tries to retain as many important features as possible. [6]

Testing the image at multiple resolutions (the histogram regions thus map different sizes of the facial features) could eliminate some of the false positives (small local regions can be similar, but if the localities are larger, then the distinctions begin to appear). [7][8][9]

Other methods can be used also to improve the algorithm, but limitations occur because this is a pixel based system that compares images, rather than an interpretation algorithm that understands actual face features.

8. REFERENCES

[1] J. Ruiz-del-Solar, R. Verschae, M. Correa, “Recognition of faces in unconstrained environments: A comparative study,” EURASIP Journal on Advances in Signal Processing, vol. 2009, 2009

[2] A. V.Malkapurkar, R. Patil, S. Murarka, “A New Technique for LBP Method to Improve Face Recognition”, International Journal of Emerging Technology and Advanced Engineering, 2011

[3] A.Rahim, S. Azam, N. Hossain, R. Islam, “Face Recognition using Local Binary Patterns (LBP)”, Pabna University of Science and Technology, 56 Bangladesh, Global Journal of Computer Science and Technology Graphics & Vision, 2013

[4] Instituto de Optica (CSIC) Image Processing and Vision Modeling Group, LBP texture analysis, http://ftp.optica.csic.es/page49/styled-2/page60.html, accessed 10 Oct 2014

[5] D. Rosner, C.A. Boiangiu, M. Zaharescu, I. Bucur, "Image Skew Detection: A Comprehensive Study", Proceedings of IWoCPS-3, The Third International Workshop On Cyber Physical Systems, Bucharest, Romania, May 29-30, 2014, Pos 10

[6] A. Tigora, C. A. Boiangiu, “Image Color Reduction Using Iterative Refinement”, International Journal of Mathematical Models and Methods in Applied Sciences, Volume 8, 2014, pp. 203-207

[7] Adrian Enache, Costin-Anton Boiangiu, "A Pyramidal Scheme of Residue Hypercubes for Adaptive Video Streaming", International Journal of Computers and Communications, Volume 8, 2014, pp. 128-133

[8] Adrian Enache, Costin-Anton Boiangiu, "Adaptive Video Transmission Using Residue Octree Cubes", Computers, Volume 3, Number 2, 2014, pp. 58-68.

[9] Adrian Enache, Costin-Anton Boiangiu, “Adaptive video streaming using residue hypercubes,” in Proceedings of the 12th International Conference on Circuits, Systems, Electronics, Control & Signal Processing (CSECS '13), Budapest, Hungary, December 10-12, 2013, pp. 173-179.