Assignment 2 & 3 : Face Recognition using LBP

Team Name : The Salvator Brothers
School of Engineering and Applied Science, Ahmedabad University

Computer Vision - CSP502, Winter 2021

1st Kirtan Kalaria
AUL202005
Ahmedabad University
Ahmedabad, India
kirtan.k@ahduni.edu.in

2nd Manav Vagrecha
AU1841022
Ahmedabad University
Ahmedabad, India
manavkumar.v@ahduni.edu.in

CONTENTS

***Abstract*—The face area is first divided into small regions from which Local Binary Patterns (LBP), histograms are extracted and concatenated into a single feature vector. This feature vector forms an efficient representation of the face and is used to measure similarities between images.**

***Index Terms*—Local Binary Pattern (LBP), Feature Extraction, Classification, Pattern recognition, histogram, feature vector**

I. INTRODUCTION

The face of a human being conveys a lot of information about identity and emotional state of the person. Face recognition is an interesting and challenging problem, and impacts important applications in many areas such as identification for law enforcement, authentication for banking and security system access, and personal identification among others. Here, we mainly categorize the problem into three parts such as

- Face Representation
- Feature Extraction
- Classification

Face representation represents how to model a face and determines the successive algorithms of detection and recognition. The most useful and unique features of the face image are extracted in the feature extraction phase. In the classification the face image is compared with the images from the database. Following are few face recognition algorithms :

- LBPH (Local Binary Pattern Histograms)
- HoG (Histogram of Oriented Gradients)
- SIFT (Scale-Invariant Feature Transform)
- SURF (Speeded-Up Robust Features)
- Eigenfaces
- Fisherfaces

In our work, we empirically evaluate face recognition which considers both shape and texture information to represent face images based on Local Binary Patterns for person-independent face recognition.

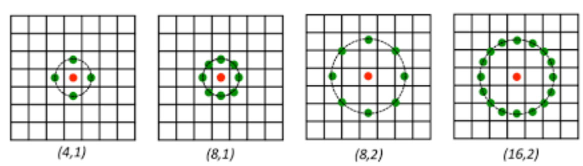

Fig. 1. Most popular LBP operators

Following are the general parameters used in LBPH method:

- **Radius:** the radius is used to build the circular local binary pattern and represents the radius around the central pixel. It is usually set to 1.
- **Neighbors:** the number of sample points to build the circular local binary pattern. Keep in mind: the more sample points you include, the higher the computational cost. It is usually set to 8.
- **Grid X:** the number of cells in the horizontal direction. The more cells, the finer the grid, the higher the dimensionality of the resulting feature vector. It is usually set to 8.

- **Grid Y:** the number of cells in the vertical direction. The more cells, the finer the grid, the higher the dimensionality of the resulting feature vector. It is usually set to 8.

Due to radius and neighbours, it is oftenly called as Circular LBP. Fig: 1 shows the different types of LBP operators used in different scenarios to define their specific neighbourhood as required

II. Motivation and Background

Today in this information era, data is secured by passwords, encryption keys, fingerprints, and many other modes. The human face plays an important role in our social interaction and in conveying people's identity. Biometric face recognition technology has received significant attention in the past several years due to its potential for applications in both law enforcement and non-law enforcement agencies. As compared with other biometrics systems using fingerprint, palm print, and iris, face recognition has distinct advantages because of its non-contact process. Images can be captured from a distance without touching the person and a face can be extracted from that image. The identification does not require interacting with the person. In addition, recognized face images can be recorded and archival can later help to identify the person(s).

III. Detailed Mathematical analysis & Process Flow

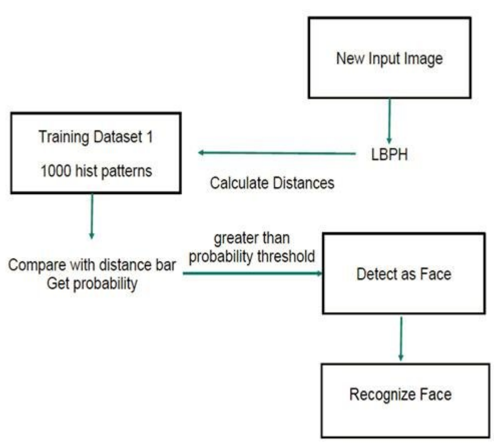

Fig. 2. FlowChart of Face Recognition using LBPH method

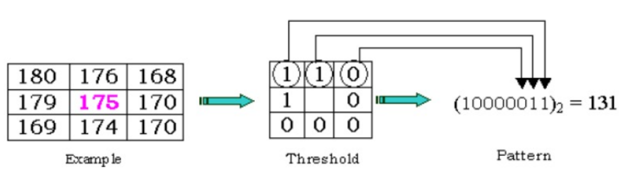

Fig. 3. Demonstration of LBP method

As shown in the Fig: 3, for any pixel C, the neighbouring pixels are compared with it and if the neighbouring pixel N is greater than the pixel C, then output will be 1, or 0 otherwise. That is,

$$l(i) = \begin{cases} 1; & p(i) - p(j) \geq 0 \\ 0; & p(i) - p(j) < 0 \end{cases}$$

After getting thresholding for every selected neighbours, we will select the point at the topmost Left corner and write all the points in the defined order and then we will calculate the decimal value of the specific binary sequence obtained. This will give a decimal value for that specific pixel C at the center based on the neighbourhoods. We will perform this task for each and every pixel and as a result a different figure will be obtained which is basically the output of the LBP operator.

$$LBP_{N,R}(x_c, y_c) = \sum_{n=1}^{N-1} s(g_n - g_c)2^i$$

This method uses the signs of Haar — a set of elementary combinations of dark and bright areas. As shown in the figure below, signs are divided into three types

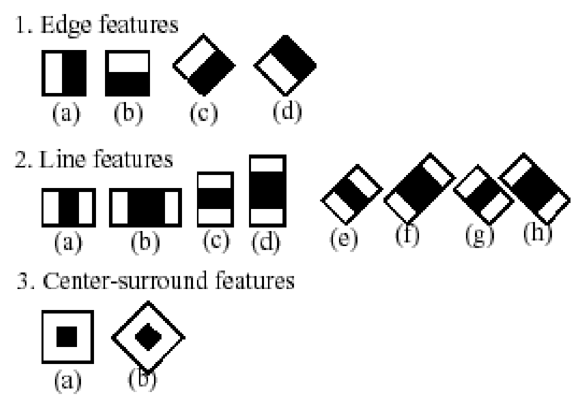

Fig. 4. Haar-Like Features

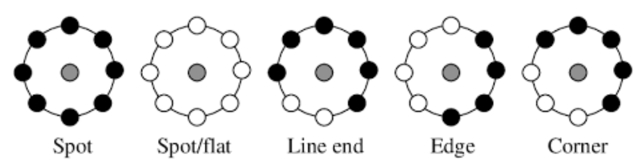

Fig. 5. Textural Features in Pixel Representation

Now, we will disintegrate the obtained Output image into different blocks. Now, we will calculate the histogram of the features for each specific blocks and we will finally concatenate all the histograms into one which will indicate the feature vector of the whole image. This histogram effectively has a description of the face on three different levels of locality: the LBP labels for the histogram contain information about the patterns on a pixel-level, the labels are summed over a small region to produce information on a regional level

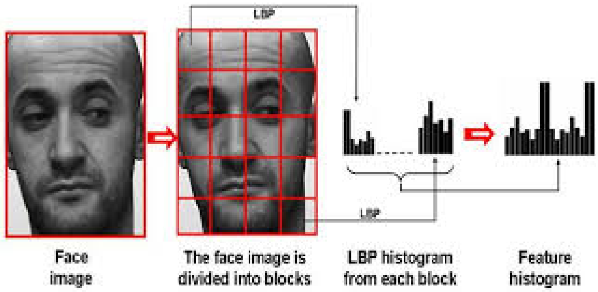

Fig. 6. Feature Extraction using Histogram

and the regional histograms are concatenated to build a global description of the face.

The nearest-neighbor classifier is used to match the new image with the trained template, which is computed by one of these formulas :

- Histogram Intersection

$$D(S, M) = \sum_i min(S_i, M_i)$$

- Log-Likelihood Statistic

$$L(S, M) = -\sum_i min(S_i, M_i)$$

- Chi-squared (χ^2) Statistic

$$\chi^2(S, M) = \sum_i \frac{(S_i - M_i)^2}{(S_i + M_i)}$$

- Euclidean Distance

$$U(S, M) = \sqrt{\sum_i (S_i^2 - M_i^2)}$$

- Ada Boost

$$\epsilon_m = \frac{\sum_{j=1}^{N} (w_j^{(i)}) I(S_i * M_j \neq T_j)}{\sum_{j=1}^{N} (w_j^{(i)})}$$

IV. Comparison with other Algorithms

A. Time Complexity

- General order of time complexity is: HOG<LBP<SURF<SIFT

B. Space complexity

- It's space complexity is the one of the lowest.
- Its order is more or less similar to that of time complexity.

C. Sensitivity to noise

- LBP is significantly sensitive to noise.
- A change of a noisy pixel changes the binary pattern. Multiple noisy points in a block disturbs the histogram significantly.
- Smoothing can be used prior to LBP to address this issue.

D. Drawback of algorithm

- Although basic form LBP is computationally less expensive, its efficiency is impacted by storing an array of histograms or appending them to form a single big histogram.
- These drawbacks can be addressed by using it in combination with other techniques.
- Focused specific feature extraction is more computationally efficient.
- It sometimes has poor detection and faulty recognition in low illumination or darker skin colors.

V. Results

A. Data Information

The ORL face database

This directory contains a set of faces taken between April 1992 and April 1994 at the Olivetti Research Laboratory in Cambridge, UK.

There are 10 different images of 40 distinct subjects. For some of the subjects, the images were taken at different times, varying lighting slightly, facial expressions (open/closed eyes, smiling/non-smiling) and facial details (glasses/no-glasses). All the images are taken against a dark homogeneous background and the subjects are in up-right, frontal position (with tolerance for some side movement).

The files are in PGM format and can be conveniently viewed using the 'xv' program. The size of each image is 92x112, 8-bit grey levels. The images are organised in 40 directories (one for each subject) named as:

sX

where X indicates the subject number (between 1 and 40). In each directory there are 10 different images of the selected subject named as:

Y.pgm

where Y indicates which image for the specific subject (between 1 and 10).

When using these images, please give credit to Olivetti Research Laboratory. A convenient reference is the face recognition work which uses some of these images: F. Samaria and A. Harter "Parameterisation of a stochastic model for human face identification" 2nd IEEE Workshop on Applications of Computer Vision December 1994, Sarasota (Florida). The paper is available via anonymous ftp from quince.cam-orl.co.uk and is stored in pub/users/fs/IEEE_workshop.ps.Z

B. Plot

C. Explanation of Code and Results

We began by defining the functions to first get the LBP texture/image and then using it, the histogram for an image by appending individual histograms from each block of the

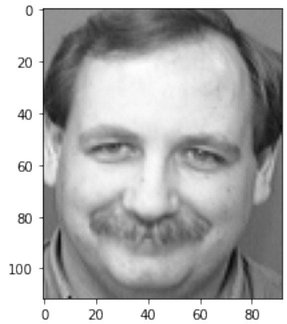

Fig. 7. Example Image for Recognition

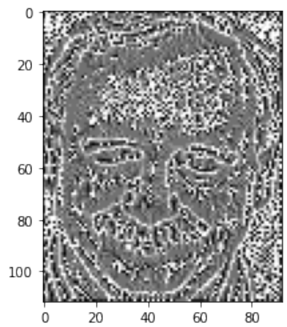

Fig. 8. LBP processed image

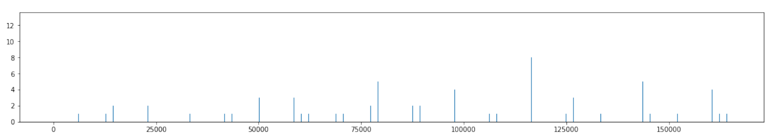

Fig. 9. Histogram of block wise complete image

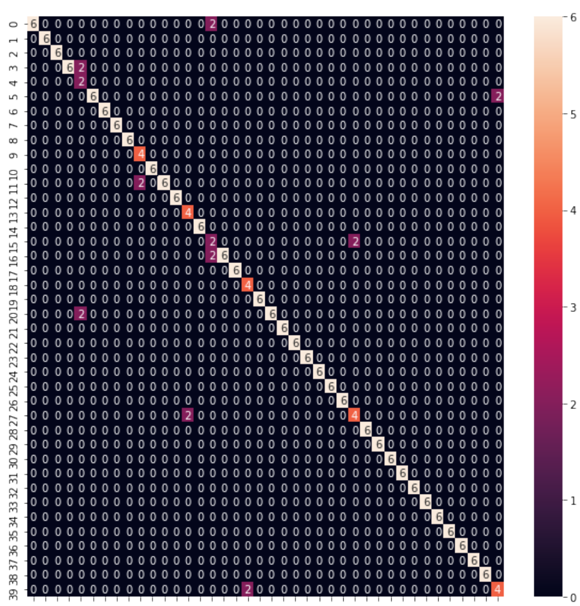

Fig. 10. Confusion matrix of the recognition of image and the actual name image

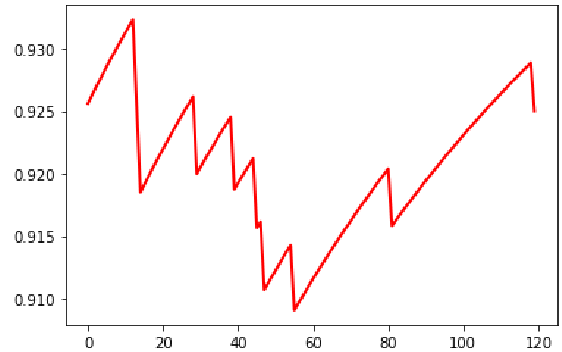

Fig. 11. Graph of Micro VS Prediction

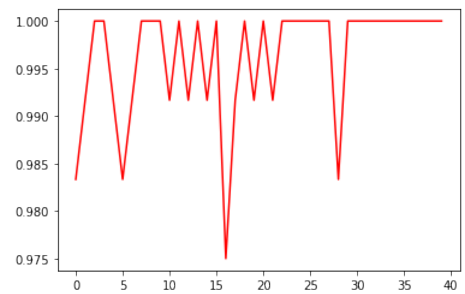

Fig. 12. Accuracy vs Class

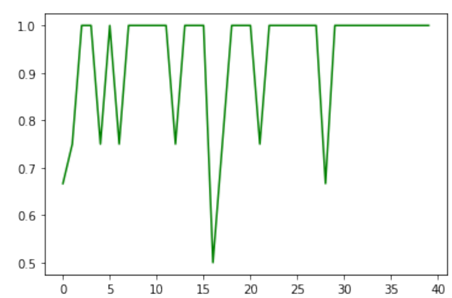

Fig. 13. Prediction vs Class

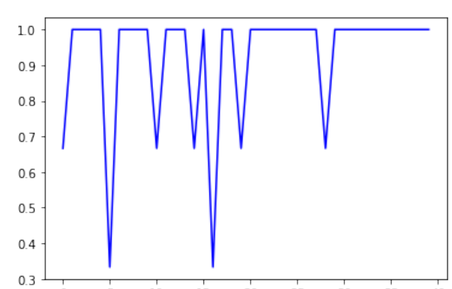

Fig. 14. Recall vs Class

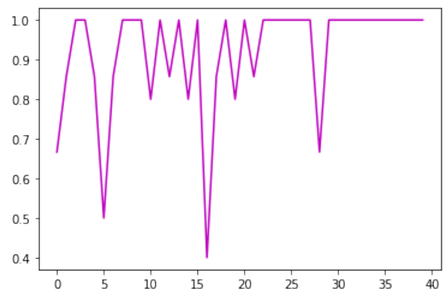

Fig. 15. F1 vs Class

image. Then we defined functions for and set up training and validation such that we have a (label, histogram) tuples for both the training and validation(testing) data, which can be used to make and test predictions. The role of raw images ends here. Next we also made a confusion matrix and with more clarity about the result interpretations, added a few more functions to add statistical performance measures. Then we make make the predictions for class labels and compare them with the actual class labels in the validation data. During the prediction process, we also calculate a cumulative accuracy (Micro F-1) and plot it to understand the variation of the accuracy. Since our validation data is ordered by class labels, we can get a visual sense of what data points lowered accuracy of prediction. We also plot a heat map for the confusion matrix to visualize it better.

VI. Comparison with others

We compared with other group who implemented LBP and we found out that the accuracy was more or less the same.

VII. Conclusion

In this work, we have implemented Face Recognition using simple Multi-Block LBPH method. It is seen that the algorithm is effective as it is less complex, more computationally beneficial and simpler than other algorithms. The resultant accuracy of this algorithm over our dataset of 400 images of 40 persons is 92.5% ie. from 120 test data, around 111 images we recognized perfectly and others were unrecognized.

References

[1] K. S. do Prado, "Face recognition: Understanding lbph algorithm," 03-Feb-2018. [Online]. Available: https://towardsdatascience.com/face-recognition-how-lbph-works-90ec258c3d6b# [Accessed: 20-Feb-2021].

[2] M. Pietikäinen, "Local Binary Patterns," Scholarpedia. [Online]. Available: http://www.scholarpedia.org/article/Local_Binary_Patterns. [Accessed: 20-Feb-2021].

[3] Zhang G., Huang X., Li S.Z., Wang Y., Wu X. (2004) Boosting Local Binary Pattern (LBP)-Based Face Recognition. In: Li S.Z., Lai J., Tan T., Feng G., Wang Y. (eds) Advances in Biometric Person Authentication. SINOBIOMETRICS 2004. Lecture Notes in Computer Science, vol 3338. Springer, Berlin, Heidelberg. https://doi.org/10.1007/978-3-540-30548-4_21

[4] B. Heisele, T. Poggio, and M. Pontil. Face detection in still gray images. A.I. Memo, (1687), 2000

[5] T.Mita, T. Kaneko, and O. Hori. Joint haar-like features for face detection. In ICCV, 2005.