



FACE RECOGNATION USING LOCAL BINARY PATTERN (LBP)

A Report for the Evaluation 3 of Project 2

Submitted by

VIBHOR CHOUDHARY

(1613101817/16SCSE101830)

*in partial fulfillment for the award of the
degree of*

Bachelor of Technology

IN

Computer Science and Engineering

SCHOOL OF COMPUTING SCIENCE AND ENGINEERING

Under the Supervision of

Mr. S. KALIDASS

ASSISTANT PROFESSOR

MAY-2020

TABLE OF CONTENTS

CHAPTER NO.	TITLE	PAGE NO.
1.	Abstract	3
2.	Introduction	4
3.	Existing System	5
4.	Proposed system	6
5.	Implementation or architecture diagrams	9
6.	Output / Result / Screenshot	14
7.	Conclusion/Future Enhancement	15
8.	References	16

Abstract:-

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. In our research work mainly consists of three parts, namely face representation, feature extraction and 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. In our research 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. 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

Introduction:-

Biometrics

Biometrics is used in the process of authentication of a person by verifying or identifying that a user requesting a network resource is who he, she, or it claims to be, and vice versa. It uses the property that a human trait associated with a person itself like structure of finger, face details etc. By comparing the existing data with the incoming data we can verify the identity of a particular person . There are many types of biometric system like fingerprint recognition, face detection and recognition, iris recognition etc., these traits are used for human identification in surveillance system, criminal identification. Advantages of using these traits for identification are that they cannot be forgotten or lost. These are unique features of a human being which is being used widely .

Face Recognition

Face is a complex multidimensional structure and needs good computing techniques for recognition. The face is our primary and first focus of attention in social life playing an important role in identity of individual. We can recognize a number of faces learned throughout our lifespan and identify that faces at a glance even after years. There may be variations in faces due to aging and distractions like beard, glasses or change of hairstyles. Face recognition is an integral part of biometrics. In biometrics basic traits of human is matched to the existing data and depending on result of matching identification of a human being is traced. Facial features are extracted and implemented through algorithms which are efficient and some modifications are done to improve the existing algorithm models. Computers that detect and recognize faces could be applied to a wide variety of practical applications including criminal identification, security systems, identity verification etc. Face detection and recognition is used in many places nowadays, in websites hosting images and social networking sites. Face recognition and detection can be achieved using technologies related to computer science.

Features extracted from a face are processed and compared with similarly processed faces present in the database. If a face is recognized it is known or the system may show a similar face existing in database else it is unknown. In surveillance system if a unknown face appears more than one time then it is stored in database for further recognition. These steps are very useful in criminal identification. In general, face recognition techniques can be divided into two groups based on the face representation they use appearance-based, which uses holistic texture features and is applied to either whole-face or specific regions in a face image and feature-based, which uses geometric facial features (mouth, eyes, brows, cheeks etc), and geometric relationships between them.

Existing System:-

Face recognition is the challenge of classifying whose face is in an input image. This is different than face detection where the challenge is determining if there is a face in the input image. With face recognition, we need an existing database of faces. Given a new image of a face, we need to report the person's name.

The success of LBP has continued since 2011. A large number of new LBP variants have been proposed, including, for example, the Median Robust Extended Local Binary Pattern (MRELBP) operator (Liu et al. 2016a). An extensive experimental evaluation of different LBP and deep texture descriptors is presented in (Liu et al. 2016b). The robustness of texture operators against different classification challenges is considered, including changes in rotation, scale, illumination, viewpoint, number of classes, different types of image degradation, and computational complexity. The best overall performance is obtained for MRELBP when distinctiveness, robustness and computational complexity are all taken into consideration. After this study, a taxonomy of LBP variants is presented together with extensive experimental evaluation (Liu et al. 2017). LBP can be seen as a simple Bag-of Words (BoW)

descriptor, in which each LBP pattern corresponds to a word. A comprehensive survey of BoW and deep CNN texture representations is presented in (Liu et al. 2019).

Proposed System:-

Local binary pattern (LBP) is a simple yet very efficient texture operator which labels the pixels of an image by thresholding the neighborhood of each pixel and considers the result as a binary number. Due to its discriminative power and computational simplicity, LBP texture operator has become a popular approach in various applications. It can be seen as a unifying approach to the traditionally divergent statistical and structural models of texture analysis. Perhaps the most important property of the LBP operator in real-world applications is its robustness to monotonic gray-scale changes caused, for example, by illumination variations. Another important property is its computational simplicity, which makes it possible to analyze images in challenging real-time settings.

The basic idea for developing the LBP operator was that two-dimensional surface textures can be described by two complementary measures: local spatial patterns and gray scale contrast. The original LBP operator (Ojala et al. 1996) forms labels for the image pixels by thresholding the 3 x 3 neighborhood of each pixel with the center value and considering the result as a binary number. The histogram of these $2^8 = 256$ different labels can then be used as a texture descriptor. This operator used jointly with a simple local contrast measure provided very good performance in unsupervised texture segmentation (Ojala and Pietikäinen 1999). After this, many related approaches have been developed for texture and color texture segmentation.

The LBP operator was extended to use neighborhoods of different sizes (Ojala et al. 2002). Using a circular neighborhood and bilinearly interpolating values at non-integer pixel coordinates allow any radius and number of pixels in the neighborhood. The gray scale variance of the local neighborhood can be used as the complementary contrast measure. In the following, the notation (P,R) will be used for pixel neighborhoods which means P sampling points on a circle of radius of R .

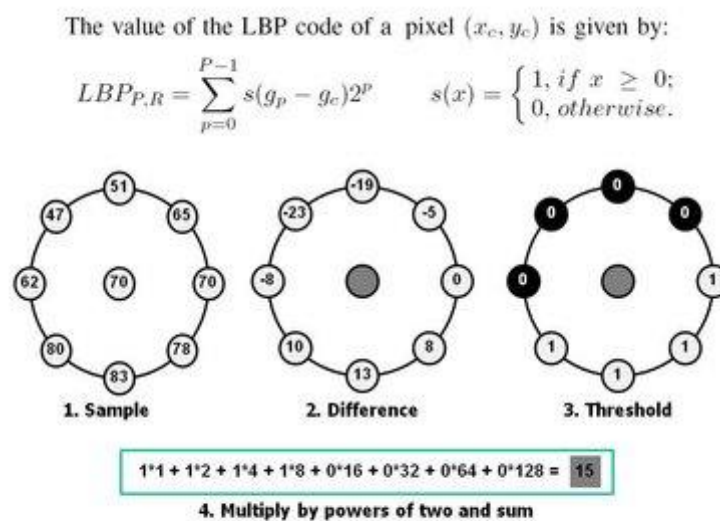


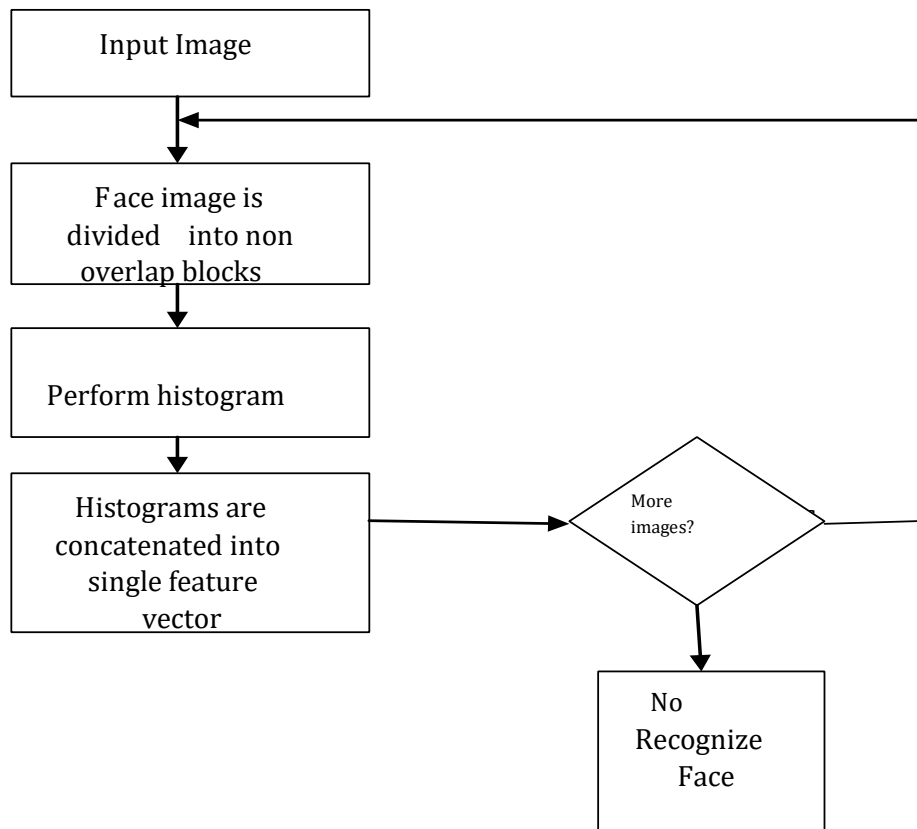
Figure 2: An example of LBP computation.

Another extension to the original operator is the definition of so-called uniform patterns, which can be used to reduce the length of the feature vector and implement a simple rotation-invariant descriptor. This extension was inspired by the fact that some binary patterns occur more commonly in texture images than others. A local binary pattern is called uniform if the binary pattern contains at most two bitwise transitions from 0 to 1 or vice versa when the bit

pattern is traversed circularly. For example, the patterns 00000000 (0 transitions), 01110000 (2 transitions) and 11001111 (2 transitions) are uniform whereas the patterns 11001001 (4 transitions) and 01010010 (6 transitions) are not. In the computation of the LBP labels, uniform patterns are used so that there is a separate label for each uniform pattern and all the non-uniform patterns are labeled with a single label. For example, when using (8,R) neighborhood, there are a total of 256 patterns, 58 of which are uniform, which yields in 59 different labels.

Ojala et al. (2002) noticed in their experiments with texture images that uniform patterns account for a little less than 90% of all patterns when using the (8,1) neighborhood and for around 70% in the (16,2) neighborhood. Each bin (LBP code) can be regarded as a micro-texton. Local primitives which are codified by these bins include different types of curved edges, spots, flat areas etc.

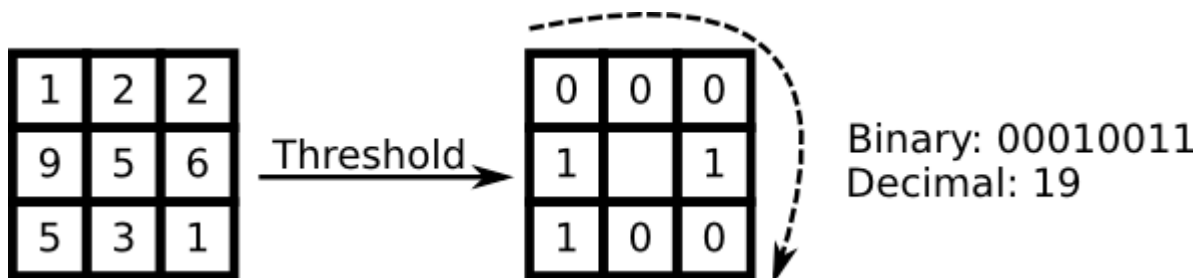
Implementation or architectural diagrams :-



So in order to get good recognition rates you'll need at least 8(+1) images for each person and the Fisherfaces method doesn't really help here.

So some research concentrated on extracting local features from images. The idea is to not look at the whole image as a high-dimensional vector, but describe only local features of an object. The features you extract this way will have a low-dimensionality implicitly. A fine idea! But you'll soon observe the image representation we are given doesn't only suffer from illumination variations. Think of things like scale, translation or rotation in images - your local description has to be at least a bit robust against those things. The Local Binary Patterns

methodology has its roots in 2D texture analysis. The basic idea of Local Binary Patterns is to summarize the local structure in an image by comparing each pixel with its neighborhood. Take a pixel as center and threshold its neighbors against. If the intensity of the center pixel is greater-equal its neighbor, then denote it with 1 and 0 if not. You'll end up with a binary number for each pixel, just like 11001111. So with 8 surrounding pixels you'll end up with 2^8 possible combinations, called Local Binary Patterns or sometimes referred to as LBP codes. The first LBP operator described in literature actually used a fixed 3 x 3 neighborhood just like this:



ALGORITHMIC DESCRIPTION

A more formal description of the LBP operator can be given as:

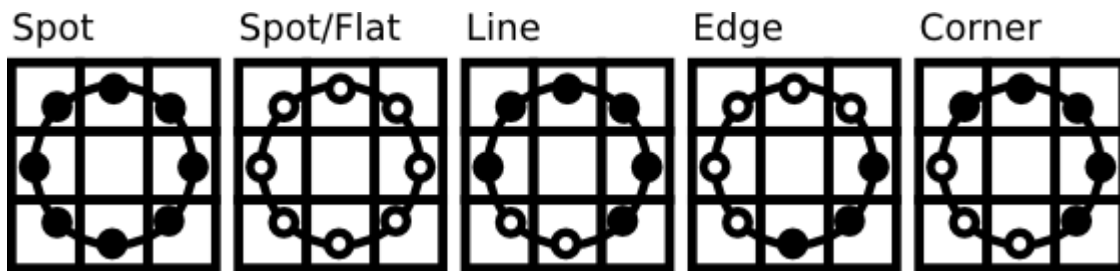
$$\text{LBP}(x_c, y_c) = \sum_{p=0}^{P-1} 2^p s(i_p - i_c)$$

, with (x_c, y_c) as central pixel with intensity i_c ; and i_n being the intensity of the the neighbor

pixel. s is the sign function defined as:

$$s(x) = \begin{cases} 1 & \text{if } x \geq 0 \\ 0 & \text{else} \end{cases} \quad (1)$$

This description enables you to capture very fine grained details in images. In fact the authors were able to compete with state of the art results for texture classification. Soon after the operator was published it was noted, that a fixed neighborhood fails to encode details differing in scale. So the operator was extended to use a variable neighborhood . The idea is to align an arbitrary number of neighbors on a circle with a variable radius, which enables to capture the following neighborhoods:



For a given Point (x_c, y_c) the position of the neighbor (x_p, y_p) , $p \in P$ can be calculated by:

$$\begin{aligned} x_p &= x_c + R \cos\left(\frac{2\pi p}{P}\right) \\ y_p &= y_c - R \sin\left(\frac{2\pi p}{P}\right) \end{aligned}$$

Where R is the radius of the circle and P is the number of sample points. The operator is an extension to the original LBP codes, so it's sometimes called Extended LBP (also referred to as Circular LBP) . If a points coordinate on the circle doesn't correspond to image coordinates, the point get's interpolated. Computer science has a bunch of clever interpolation schemes, the OpenCV implementation does a bilinear interpolation:

$$f(x, y) \approx \begin{bmatrix} 1-x & x \end{bmatrix} \begin{bmatrix} f(0,0) & f(0,1) \\ f(1,0) & f(1,1) \end{bmatrix} \begin{bmatrix} 1-y \\ y \end{bmatrix}.$$

IMPLEMENTATION OF LBP

The LBP feature vector, in its simplest form, is created in the following manner:

- Divide the examined window into cells (e.g. 16x16 pixels for each cell).
- For each pixel in a cell, compare the pixel to each of its 8 neighbors (on its left-top, left-middle, left-bottom, right-top, etc.). Follow the pixels along a circle, i.e. clockwise or counter-clockwise.
- Where the center pixel's value is greater than the neighbor's value, write "0". Otherwise, write "1". This gives an 8-digit binary number (which is usually converted to decimal for convenience).
- Compute the histogram, over the cell, of the frequency of each "number" occurring (i.e., each combination of which pixels are smaller and which are greater than the center). This histogram can be seen as a 256-dimensional feature vector.
- Optionally normalize the histogram.
- Concatenate (normalized) histograms of all cells. This gives a feature vector for the entire window.

The feature vector can now be processed using the Support vector machine, extreme learning machines, or some other machine learning algorithm to classify images. Such classifiers can be used for face recognition or texture analysis.

A useful extension to the original operator is the so-called uniform pattern, which can be used to reduce the length of the feature vector and implement a simple rotation invariant descriptor. This idea is motivated by the fact that some binary patterns occur more commonly in texture images than others. A local binary pattern is called uniform if the binary pattern contains at most two 0-1 or 1-0 transitions. For example, 00010000 (2 transitions) is a uniform pattern, but 01010100 (6 transitions) is not. In the computation of the LBP histogram, the histogram has a separate bin for every uniform pattern, and all non-uniform patterns are assigned to a single bin. Using uniform patterns, the length of the feature vector for a single cell reduces from 256 to 59. The 58 uniform binary patterns correspond to the integers 0, 1, 2, 3, 4, 6, 7, 8, 12, 14, 15, 16, 24, 28, 30, 31, 32, 48, 56, 60, 62, 63, 64, 96, 112, 120, 124, 126, 127, 128, 129, 131, 135, 143, 159, 191, 192, 193, 195, 199, 207, 223, 224, 225, 227, 231, 239, 240, 241, 243, 247, 248, 249, 251, 252, 253, 254 and 255.

Output / Result / Screenshot:-

A colored face image is transformed to grey scale image as grey scale images.



A colored face image

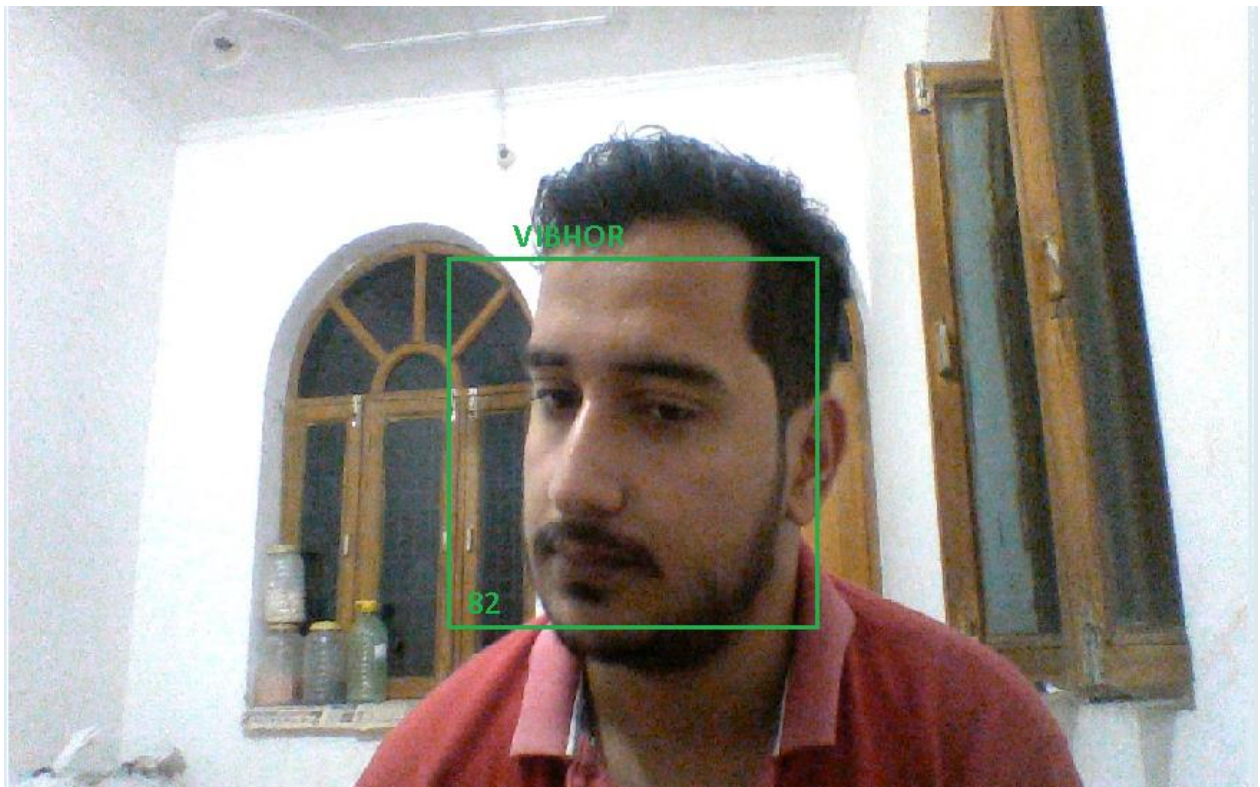
Grey scale face image

Training Set

There are 30 data sets are maintained for the single person.



FINAL OUTPUT



Conclusion/Future Enhancement:-

Since the face picture is composed of many minute patterns which can be efficiently identified by applying the local binary pattern operator. In this research paper, the more efficient facial recognition technology is described that was successfully applied to different analysis tasks, including face detection and recognition, Iris detection, fingerprint recognition, and problems related to expressions in face. This method will extract the most important feature from the given image to match the similarities between the different faces. Therefore this local binary pattern method will work best when compared to the other methods and also provides an efficient result.

References:-

- [1] Sarabjit Singh, Amritpal Kaur, Taqdir, "Face Recognition Technique using Local Binary Pattern Method", International Journal of Advanced Research in Computer and Communication Engineering Vol. 4, Issue 3, March 2015.

- [2] Roberto Bruunelli and Tomaso Poggio, "Face Recognition: Features versus Templates", IEEE Transactions on Pattern analysis and Machine intelligence Vol.15.No.10, October 1993.

- [3] Srinivasulu Asadi , Dr.Ch.D.V.Subba Rao , V.Saikrishna, A Comparative study of Face Recognition with Principal Component Analysis and Cross-Correlation Technique, International Journal of Computer Applications (0975 – 8887), Vol.10, No.8, November 2010.

- [4] Onsen Toygar, Adnan Acan, Face Recognition Using PCA, LDA and ICA Approaches on Colored Images, Journal of Electrical and Electronics Engineering Vol.3, no. 1, pp.735-743, 2003.