ISSN:0975-9646

Comparison Study of Algorithms Used for Feature Extraction in Facial Recognition.

Shreyas N Raj[1], Prof.Vijayalakshmi Niar[2]

[1]PG Scholar, [2]Assistant Professor
Department of Computer Science Christ University,
Bengaluru, India

***Abstract*— This work does a comparative study on the algorithms used for Feature extraction in Facial recognition. The algorithms are run through a common data set and the percentage of features extracted are recorded. The data recorded is converted to graphs to get a better understanding of the outcome. The data set designed for this testing is created on the basic problems that are faced in the field of facial recognition. The algorithms are executed on these for the sole reason of finding out which of them is more reliable to be used in the system in current scenarios. Also, the algorithms are chosen in such a manner that they comprise of both the latest technology as well as the old one, which makes it explicit compared to the work existing in this field. The data recorded from the execution is converted to graphs, which gives the reader a better understanding of the outcome rather than going through tables of data.**

Keywords*— Feature Extraction; Speeded Up Robust Features (SURF); Histogram of Oriented Gradients (HOG); Local Binary Patterns (LBP).

I. INTRODUCTION

The application in the field of biometrics is booming currently. It is considered safer to have a biometric access these days for secure access rather than having the old mechanisms of having passwords, PIN, or key as a password, as these can be duplicated or stolen easily. The various types of biometrics in use these days are finger prints, iris scanners, palm scanners, face recognition etc. The security obtainable with face recognition systems are comparatively better with other biometric because they can be faked or tampered with. That's the prime reason why the facial biometric system is far more superior. Face recognition is utilized more when compared with other biometrics, with the sole reason that the user need not intervene with the system [1].

Face recognition is playing an important role in various areas and has a wide range of implementation. It has various applications of pattern matching, which draws a lot of attention. Face recognition is a process that includes three stages namely, face detection, feature extraction and face recognition [1].

Fig 1 shows the basic steps in processing a face, which includes detection, alignment, extracting features and matching. For each stage there are various algorithms that can be applied.

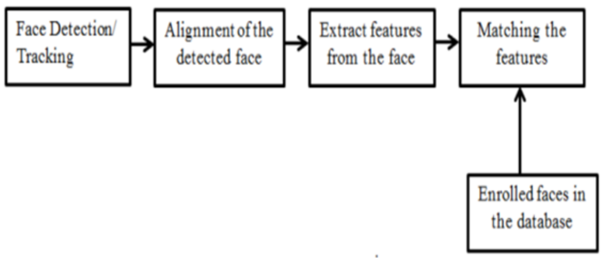

Fig 1: Stages in face recognition

Face detection is the first step to be carried out in recognition of faces. It is a process of detecting the human faces from a cluttered image. There are various algorithms specifically designed for the purpose of detection of faces and viola Jones face detection being the most commonly used [1]. The detected face need to be aligned for further processing.

Feature extraction is a crucial step in the process of face recognition. Feature extraction is a special form of dimensionality reduction. The main goal of feature extraction is to obtain the most relevant information from the original data and represent that information in a lower dimensionality space. When the input data to an algorithm is too large to be processed and it is suspected to be redundant, then the input data will be transformed into a reduced representation set of features. Transforming the input data into the set of features is called feature extraction [2]. It is this step that finds out points on the face, which is considered as data for the system. The data is measured by the system to understand and differentiate between faces of different people.

As faces have the most complicated data that can vary with respect to various conditions, to achieve good accuracy rate in recognition of face is a difficult task. The recognition rate on face fluctuates during different lighting condition, face orientation, expression, and aging effect being some of the factors. These factors deteriorate the recognition rate of face recognition system. As technology emerges, the application of face recognition has improved and there is various research conducted in this field that help in overcoming all these factors to a great extent.

Face recognition system can be used to simply find a face or try to compare the face detected and find the match. The latter is applied in biometric systems or security. It is a mixture of features that are extracted and compared with and can allow access if there is a match.

The application in security is quite complicated as the person’s face might mostly not be visible, hence, detecting the face might get complicated. Also the drawback is that we need to have a well-defined database that can help us compare and find a match.

For all of these applications, feature extraction should give accurate results and should be efficient enough to extract as many as features that can be extracted. There are many algorithms that are designed that helps us in extracting features. These algorithms are divided in two categories:

- Hand engineered feature extraction methods (e.g. SURF, HOG, GIST, Scale Invariant Feature Transform(SIFT), LBP).
- Learn features that are discriminative in the given context (i.e. Sparse Coding, Auto Encoders, Restricted Boltzmann Machines, Principal Component Analysis(PCA), Independent Component Analysis (ICA), K-means).

Facial feature extraction is an important part in the Facial recognition system because, it is based on these features the system can recognise a face. It helps in picking on feature points that define a persons’ face and differentiate it from others. The common features that the system gathers are length of the jaw line, spacing of eyes, nose, mouth and ears from one another, any unusual markings such as a mole, scar or any kind of deformity found within the area of the face.

This work focuses on hand engineered feature extraction methods (e.g. SIFT, SURF, VLAD, HOG, GIST, LBP). A comparative study is performed on HOG, SURF and LBP on a given data set to find which of these algorithms or methods are most efficient.

1.1*Speeded Up Robust Features (SURF)*

SURF is used not only for feature detector but descriptor as well. It is used mainly for object recognition, image registration, classification and 3D reconstruction. SURF was built on another feature extraction algorithm, Scale invariant feature transform (SIFT), which was one of first algorithms used in the late 90’s. As, SURF authors’ claim, it is several times faster in fetching the results than SIFT and also very robust in nature.

SURF uses an integer approximation of the determinant of Hessian blob detector to detect the intersect points, making it capable of computing 3 integer operations using a precomputed integral image. The feature descriptor used in SURF is based on the sum of the Haar wavelet response around the point of interest, which makes it proficient to be computed with the aid of internal image.

SURF descriptors have been used to locate and recognize objects, people or faces, to reconstruct 3D scenes, to track objects and to extract points of interest.

Initially the image is transformed into coordinates. After which the SURF uses the multi-resolution pyramid technique to copy the original image with Pyramidal Gaussian or Laplacian to obtain the image with the same size but with reduced bandwidth, due to which we obtain a special blurring effect on the original image, known as Scale space and ensures that the points of interest are scale variant.

The algorithm has three main parts:

1. Interest point detection
2. Local neighbourhood description
3. Matching.

The final step, Matching is mostly done when the system is required to give access or identify anyone from a database.

1.2 *Histogram of Oriented Gradients (HOG)*

Introduced in the year 2005 by Navneet Dalal and Trigg, the histogram of oriented gradients (HOG) is another feature descriptor utilized in computer vision and image processing for the purpose of abject detection. The technique counts occurrences of gradient orientation in localized portions of the paper. This method is parallel to edge orientation histogram. Scale-invariant transforms descriptors and shape contexts. For improved accuracy the image is computed on a compact of grid of uniformly spaced cells and uses overlapping local contrast normalization. The image is further divided into small regions called cells, which are interconnected. HOG directions are compiled for each pixel within these cells. The local histogram is contrast-normalized by computing the measure of intensity across a larger region of the image, called a block, and later using these values to normalize all the cells within the block for an improved accuracy.

This normalization yields better invariance for the changes obtained on illumination and shadowing.

HOG was initially tested on the MIT data set, which comprised of 509 training set and 200 datasets, which consisted mainly of photographs of people with front face and back face. It gave promising results which made it popular and one of the, most efficient algorithms for feature extraction for human faces and objects as well.

HOG consist of the following steps:

1. Gradient computation
2. Orientation binning
3. Descriptor blocks
4. Block normalization
5. SVM classifier
6. Neural Network Classifier

1.3 *Local Binary Patterns (LBP)*

One of the oldest algorithm, invented in the year 1994, is one of the predominant algorithms used for simple feature extraction. It is also used with the HOG algorithm mentioned above to improve the performance of HOG on certain datasets. LBP is a type of visual descriptor used for classification in computer vision. LBP is the particular case of the Texture Spectrum model. It has been found to be a powerful feature for texture classification. LBP was first proposed as a gray level invariant texture primitive. LBP operator describes each pixel by its relative gray level to its neighboring pixels, e.g., if the gray level of the neighboring pixel is higher or equal, the value is set to one, otherwise to zero [3].

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

- Divide the examined window into cells (e.g. 16x16 pixels for each cell or lesser).
- For each pixel in a cell, compare the pixel to each of its 8 neighbours (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 centre pixel's value is greater than the neighbours' 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 centre). 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.

Fig 2: LBP algorithm

II. Proposed System

This paper aims at comparing the algorithms SURF, HOG and LOG on the given data set to find which of these algorithms provides significant and faster results for all the photographs in the data set. To accomplish this, we will be running the algorithms in Mat lab on the data set we have prepared. The data set consists of photographs which are taken in different conditions such as varying lighting condition, partial faces, faces facing different directions and occluded faces, which are the most common problems that are faced in this system. Initially the data set starts off with a simple set of photographs that simple to identify and as the process carries on complexity of the photographs increases. The reason for choosing such a complex data set is to find which of these algorithms are fit to be used in system for ensuring that the system functions without any disturbances and can be reliable.

The photographs from the data sets are run using the algorithms one at a time and the percentage of the features extracted is recorded for each algorithm.

The algorithms are run against the same data set to find which of these algorithms are most efficient in recognizing the features from the photographs presented. The results are recorded and portrayed in the form of graphs.

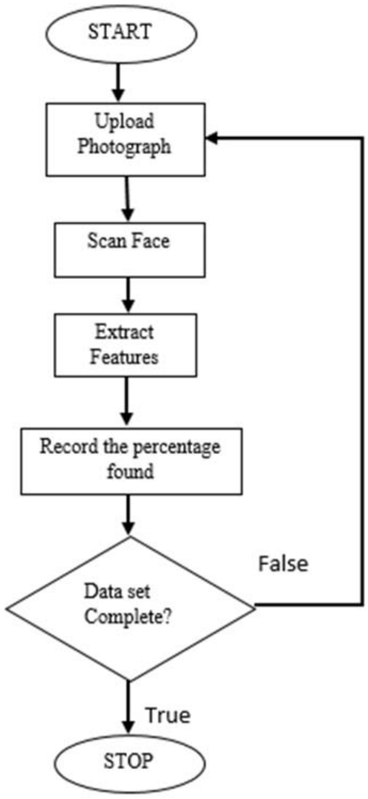

Fig 3. System Work Flow

III. Results

The results recorded after the algorithms are checked on the data set, are recorded and represented in the form of graphs as shown.

In the graphs below, the x-axis in the graph represents the percentage of features extracted from a given picture. The y-axis represents the number of the photograph in the data set.

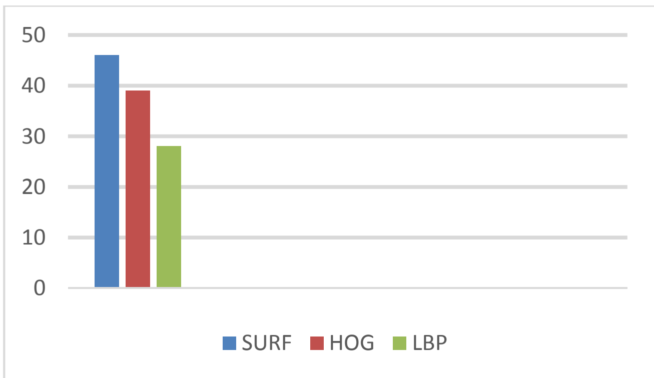

Fig 4. Performance Graph

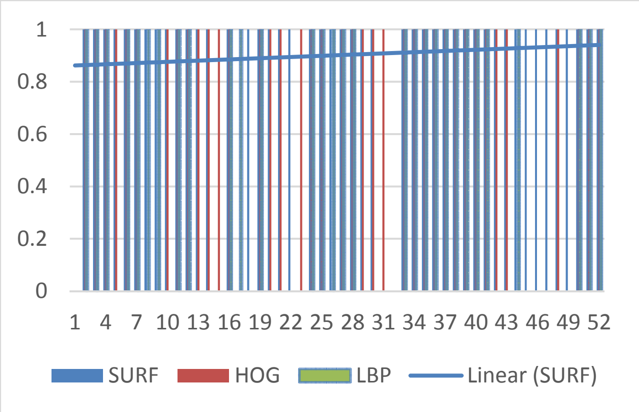

Fig 5. Linear Graph- SURF

From this linear graph, we can devise that the performance of the SURF algorithm increases eventually and doesn't stop deteriorating as and when the data set gets complicated.

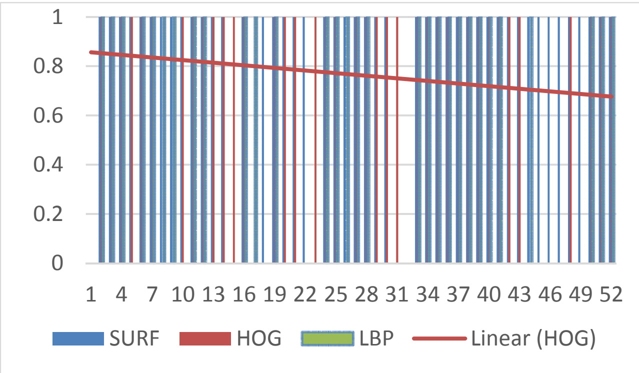

Fig 6. Linear Graph- HOG

From this linear graph, we can spot that the performance of the HOG algorithm declines as the data set gets complicated. But can perform better on certain type of data set.

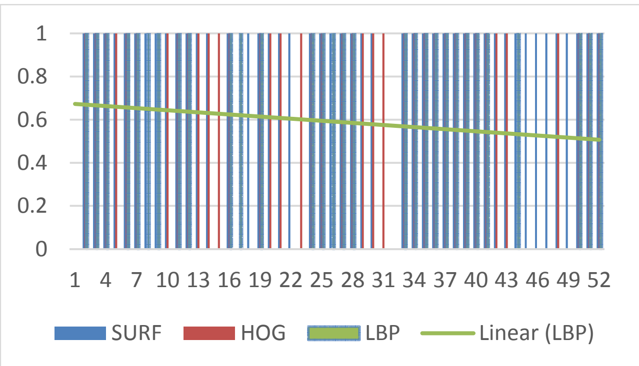

Fig 7. Linear Graph- LBP

From this linear graph, we can determine that the performance of the LBP algorithm starts off at a lower rate of feature extraction in comparison to the other two algorithms. Also, as the data set gets complicated, resulting in very less percentage of features getting extracted.

IV. CONCLUSION

From the above data (Figure 4) we can incur that SURF algorithm performs better than the other algorithms on the complete data set. Performance of this algorithm doesn't drop even in conditions such as low light photographs and photographs where only partial faces are found. The results are compounded faster, which improves the quality of the system it is used in.

In scenarios where SURF could extract up to 90% (0.9 in Fig 5), the other algorithms could gain much less.

Followed by the SURF algorithm is the HOG algorithm. As we can find from Figure 6, the graph declines gradually as the data set becomes complex. It can still extract the features, but not as much in comparison to SURF algorithm.

The last is the oldest algorithm, LBP which runs well on photographs where the face is mainly posing front. It gradually declines as the data set gets complicated and the algorithm cannot extract the features completely.

REFERENCES

[1] Erald Vucini, Muhittin Gokmen and Eduard Groller ," Face Recognition under Varying Illumination", International Journal of Computer Science and Information Technologies, Vol. 5 (4) , 2014, 2174-5278.

[2] Faizan Ahmad, Aaima Najam and Zeeshan Ahmed, " Image-based Face Detection and Recognition: "State of the Art" IEEE-11329

[3] Gaurav Kumar, Pradeep Kumar Bhatia, "A Detailed Review of Feature Extraction in Image Processing Systems"

[4] Geng Du*, Fei Su, Anni Cai." Face recognition using SURF Features"

[5] Ligang Zhang, Vinod Chandran," Discovering the Best Feature Extraction and Selection Algorithms for Spontaneous Facial Expression Recognition".

[6] Sakrapee Paisitkriangkrai, Chunhua Shen, Jian Zhang,"Face Detection with Effective Feature Extraction"

[7] Vijayalakshmi, A., and Pethuru Raj. "An Efficient Method to Recognize Human Faces From Video Sequences with Occlusionl." World of Computer Science and Information Technology Journal (WCSIT) 5.2 (2015): 28-33.

[8] Mr. Wenyi Zhao, Arvindh Krishnaswamy," Discriminant analysis of principal components for face recognition".