

Designing of Face Recognition System

Vijay Kumar Sharma

School of Electronics & Communication Engineering, Shri Mata Vaishno Devi University, Katra, India, 182320

Abstract: Face Recognition is the most popular and trending technology in the present era. It is an effective way to provide vision to a machine for better interaction with humans. The way of living will be reflected if machines can read our faces. The face recognition system will move the world in a new dimension. It will be beneficial in many ways to find the identity and security. In this paper, a face recognition system is proposed for advanced applications such as access and security, payments, criminal identifications etc. The process of identification will be based on face recognition which is further divided into three steps: detection of face, extractions of the features and classification, and real time recognition. Detection of face is recognized as the essential step of our system. It is used to extract a face in a frame, which is based on the Viola-Jones object detection algorithm that uses AdaBoost classifier with Haar and LBP features. Local Binary Patterns (LBP) is utilized to extract the unique features of the face like eyes, nose, and mouth in the feature extraction phase. The facial image is correlated with the images available in the database for the classification. The system is implemented in Python using OpenCV library. Kivy is used to create a user interface and also to build executables for different platforms.

Keywords— Face recognition; Local binary pattern (LBP); LBPH; OpenCV; Python; Kivy.

I. INTRODUCTION

Human face can be used as an identity for numbers of applications because of its uniqueness. Human faces are different for different persons basis of various parameters. The uniqueness and measurement of the different parameters help us to recognize the persons. Uniqueness of the face and different parameters separates the humans from the machines.

The face recognition is independent of user's physical interaction. It has very precise measurement and permits for high deployment and authentication. It is so easy that can use available resources to simplify the problems. In the 1960s, the first semi-automatic systems to recognize the face features like location and dimension of eye, nose, ear etc. were carried out. In 1970, Goldstein and Hartmann used 21 specific subjective markers such as hair color, lip thickness to automate the recognition. In 1990s more advance face recognition algorithms were developed, to extract the image information such as Eigenfaces (1991) which is a principal component analysis (PCA) method that uses significant features to represent the difference between facial images. Fisherfaces (1997) which is a linear discriminant analysis (LDA) method for dimensionality reduction. It works better than PCA, where Eigenfaces and Fisherfaces have a similar

approach. The efficient way of utilization of LBP provides decision pattern basis of binary numbers by observing the image pixels and considering the threshold level [1]. The obtained decision patterns in forms of binary values are get engaged in the histogram that depicts the measurement of the image. Histograms of the images should be comparing to match their similarities. In this system, LBPH is using which is one the easiest face recognition algorithms. Due to its computational simplicity and robustness to monotonic gray scale transformations. It is available in the OpenCV library, which are implementing in Python.

The organization of the paper is given as: in section II the LBP is explained. The paradigm of the system is presented in section III. The methodology and result analysis are discussed in section IV and section V conclude the paper.

II. LOCAL BINARY PATTERN

Face recognition detects unique facial features and proportions to accurately identify users, there exist several methods to perform face recognition. One of the easiest face recognition method is the Local Binary Pattern (LBP) method which was announced by Ojala et al in 1996 [3]. LBP is texture recognition algorithm [3], which describes shape and texture of a digital image. Local Binary Patterns try to divide an image into set of unit pixels in three rows and three columns where the center pixel is known as threshold which was surrounded by eight neighbor pixels. Now comparing the center pixel intensity with the neighbor pixels and gives the results accordingly by observing the center pixel intensity and threshold level. Logic '1' shows the greater value of center pixel intensity if it is equal or greater than the threshold level. Hence, a binary representation is formed for each pixel.

Every change in the pixel is considered as the change in grayscale where input image is converted into LBP image which is basically binary pattern in the form of zeros and ones. Using LBP method in face recognition gives better results, in terms of speed and discriminative power. But this method has high noise level and illumination (lightening changes) [6].

OpenCV

The Intel's research center developed the open source computer vision (OpenCV). It is open source library for the computer vision. Now, it is continued by Itseez [7, 8]. OpenCV is developed for computer vision applications and to provide real time image processing. It is written in C++ but it is also comfortable with Python, Java, Android SDK, Ruby etc. The

license nature of OpenCV is BSD so; it is free for commercial and research use. It is compatible to most of the operating systems (OSs) and runs on all the OSs. Open CV library also consists of face recognition and detection algorithms such as Eigenfaces, Fisherfaces, LBPH, Haar cascades etc [9]. It is used to detect faces, customize the image quality, texture recognition in images, hand gesture recognition, and analyze the information from the system. The OpenCV library has more than 2500 advance algorithms. These algorithms can be used to facial recognition, detection and object identification.

III. PARADIGM OF THE SYSTEM

In this paper, the process of identification based on face recognition is split into major three stages: detection of face, data set generation and trainer, face recognition.

Face Detection: It is the first and very important stage of our system. Face detection is a method to extract the desired faces from a frame. It used the Viola-Jones object detection algorithm [2] which uses AdaBoost classifier with Haar and LBP features.

There are some similarities in human faces. These predictabilities can be matched using Haar Features. Haar-like features are defined as the difference in the acquired intensity of the two rectangular regions. The most common features of human faces are the eye and nose area and depending on the darkness and brightness of the same and compared with the cheeks.

Haar-Like features are evaluated. It is the collection of numbers of the face images taken from the different sources. It provides the numbers of features and their utilization by using the Adaboost algorithm. There are various types of the methods available to classify the detection of the face like color technique, Adaboost face detector, feature invariant technique etc.

Face Recognition:

The process of face recognition is different from the detection of the face. In the process of detection of face, the system recognizes the face with an image and locates its position where as in face recognition the system recognizes specific people. Face recognition basically compares the input face within an image and visually search with all facial images in data set, find and identify specific faces within large image collections. Numbers of face recognition algorithms are available to recognize the face like Eigenfaces, Fisherfaces, LBP Histogram etc. are available where all these techniques have different type of approach to extract the image. In this system, LBPH method is used to perform face recognition. The Eigenfaces and Fisherfaces determine the most dominant features of the face in the training set and more time-consuming methods as compared to the LBPH, however LBPH figure out all the faces in training set individually. LBP is a texture recognition algorithm that is also used to represent faces [3, 10], which is one the easiest face recognition algorithms. Due to its computational simplicity and robustness to monotonic gray scale transformations. LBP is used to extract the exclusive structures of the face like eyes, nose, and mouth in the feature extraction phase. LBP is associated with the histogram of oriented gradients (HOG), to improve the

performance on same data sets [5, 11]. It is available in the OpenCV library, which are implementing in Python. Face recognition technology has many advantages such as (hygienic) just look no touching, convenient to use, it doesn't require the use of keys or RFID cards or passwords, (highly usability) everyone can use this technology as people's faces are stable it won't affect by weather like dry, wet or work conditions, accurate, secure etc.

Data Set and Trainer

In this process, face dataset of the user is created, in which 41 images of each user are taken and the attributes used are user ID and username. Which are trained into the dataset or database. The input data is synthesized and checked the available data in dataset [12]. If there are any similarities in them or they are matched to the ones in dataset, then they are recognized successfully.

IV. METHODOLOGY AND RESULT ANALYSIS

The process of identification of a person is divided into the following steps

- Face detection
- Feature extraction
- Classification

There are several facial extraction processes, but LBP is used here. The whole picture is divided into numbers of pieces and they have the different values of pixels. These pixel values are compared with the threshold level and provides the decisions. The shape, location, dimensions etc. of the pictures affected the analysis. The pieces of the picture are putted into a single histogram and basis on the nature of histogram, the decision is taken. When an input image is given, then its histogram is created. Which is compared with the histograms of the images present in the dataset. The image which is matched best is recognized. The flow diagram of the process is shown in figure 1.

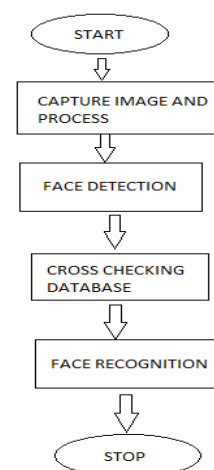


Figure 1. Flow diagram of the process

The first process to the system is to store the numbers of images in its database. This database is used to identify the related person. If camera captured any face, then it call to the database and check the measurements available in its database. The system is verified over 150 peoples and have an efficiency about 80 percent. It is tested on different environment and lighting conditions with different cameras and the results is near about same. The camera used here is Logitech C90 USB webcam.

V. CONCLUSION

In this paper, face detection and recognition using LBPH is proposed and it is implemented in Python environment by using OpenCV. Kivy is used to develop user interface and makes it executable on any platform.

To improve the calculation speed and efficiency, Haar like features are used in face detection and face recognition is done by using LBPH because of its computational simplicity and accuracy. The unique parameters of the face are observed and compared with the data available in the dataset.

REFERENCES

- [1] Matti Pietikäinen (2010) Local Binary Patterns. Scholarpedia, 5(3):9775.
- [2] Paul Viola, Michael J. Jones, Robust Real-Time Face Detection, International Journal of Computer Vision 57(2), 137–154, 2004.
- [3] T. Ojala, M. Pietikainen and D. Harwood, "A comparative study of texture measures with classification based on feature distributions" Pattern Recognition, vol. 29, 1996.
- [4] Ahonen, Timo, Abdenour Hadid, and Matti Pietikäinen. "Face recognition with local binary patterns" Computer vision-ECCV 2004 (2004): 469–481.
- [5] Xiaoyu Wang, Tony X. Han, Shuicheng Yan, "An HOG-LBP Human Detector with Partial Occlusion Handling", 2009 IEEE 12th International Conference on Computer Vision, 2009.
- [6] W. Zhao, R. Chellappa, P. J. Phillips, and A. Rosenfeld, "Face recognition: A literature survey" ACM Computing Surveys (CSUR), 35(4):399-458, 2003.
- [7] Pulli, Kari; Baksheev, Anatoly; Komyakov, Kirill; Eruhimov, Victor (1 April 2012). "Realtime Computer Vision with OpenCV". Queue. pp. 40:40–40:56. doi:10.1145/2181796.2206309.
- [8] Intel acquires Itseez: <https://opencv.org/intel-acquires-itseez.html>
- [9] Mita T, Kaneko T, Hori O. Joint haar-like features for face detection. In Tenth IEEE International Conference on Computer Vision (ICCV'05) Volume 1 2005 Oct 17 (Vol. 2, pp. 1619-1626). IEEE.
- [10] Xia, B., Sun, H. and Lu, B.L., 2008, June. Multi-view gender classification based on local Gabor binary mapping pattern and support vector machines. In 2008 IEEE International Joint Conference on Neural Networks (IEEE World Congress on Computational Intelligence) (pp. 3388-3395). IEEE.
- [11] Reddy, K.S., Kumar, V.V. and Reddy, B.E., 2015. Face Recognition Based on Texture Features using Local Ternary Patterns. *International Journal of Image, Graphics & Signal Processing*, 7(10).
- [12] Kanungo, P., Mohanty, B.P. and Mohanty, K.S., 2017, December. A simple and fast automatic face colour identification for reducing the search space. In 2017 2nd International Conference on Man and Machine Interfacing (MAMI) (pp. 1-6). IEEE.