International Journal for Research in Engineering Application & Management (IJREAM)
ISSN : 2454-9150 Vol-07, Issue-07, OCT 2021

Using Median-LBPH Algorithm for Real-Time Face Recognition System

Lohith Raj S N*, Student, BIT, Bangalore, India, rajlohith2@gmail.com
Karthik J Bharadwaj*, Student, BIT, Bangalore, India, bharadwajkarthik7@gmail.com
Harsha M V, Student, BIT, Bangalore, India, harshavana@gmail.com

Abstract - **The LBPH algorithm is used ubiquitously for Face Recognition applications in modern times because of its simplicity of implementation, despite providing high accuracy and less computation time. However, in conditions of varied illumination, face expression and angles at which face images are captured, its confidence is decreased. We propose a slightly modified algorithm that considers the median of the neighbourhood pixels rather than the pixel itself to overcome this issue. This algorithm is called Median-LBPH. The grey value of every pixel is replaced by the median of all the neighbourhood pixel values. Then the features are extracted, and a histogram representing the original image is saved in the model. This model, in turn, can be used to compare with histograms obtained from the faces in real-time footage to find a potential match. This algorithm is used in an end-to-end face recognition system, a web application prototype for Law Enforcement Agencies to maintain a central criminal database shared and accessed across various departments. A live surveillance system is added as part of this novel application so that whenever an already registered criminal appears live on surveillance cameras, a notification will be received, and personnel appropriate Law Enforcement authorities will receive e-mail and text messages through a secured channel.**

Keywords — Face Recognition, Median-Local Binary Pattern Histogram (MLBPH), Haar Cascade, Adaboost, Neighbourhood Median

I. Introduction

Apart from the intuition of the police, the Law Enforcement Agencies require help from modern technological innovations to combat crime and stay ahead of the perpetrators. There are many biometric methods in the present day that help the police to solve crimes. The most reliable ones are Fingerprints, DNA matching and Face recognition. State-of-art face recognition systems are used in authentication systems, authorization, verification of identities, most important live surveillance. It is one of the alacritous fields in research on pattern recognition and computer vision. These methods are not entirely reliable and hence bestows further enhancements, especially in applications where incorrectness in prediction is unacceptable.

Face recognition is now being used in a wide range of day-to-day applications, including tagging in Facebook and Instagram pictures, face unlock on mobile phones and as a tool for authorization of employee in offices. We can notice that these algorithms' usage ranges from layman day-to-day utilization to confidentiality intensive security applications.

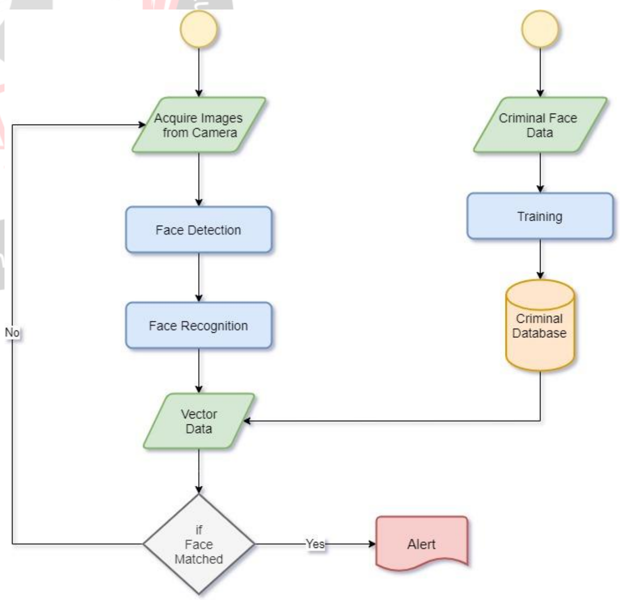

Fig.1 Flowchart of The Real-Time Surveillance System

Different types of face recognition algorithms have been proposed in the past, including Linear Discriminant Analysis (LDA), Deep Convolution Network algorithm, Sparse Coding (SC) algorithm and Histograms of Oriented Gradients (HOG) algorithm. These algorithms are undergoing rapid improvements with every passing day. OpenCV python library provides ready-made

 DOI : 10.35291/2454-9150.2021.0553

implementation of three algorithms - Eigenfaces, Fischerfaces and Local Binary Pattern Histogram. LBPH provides a much more reliable face recognition method as it can recognize images and side faces, making it robust and flexible.

Median LBPH, a new and improved implementation of the Local Binary Pattern Histogram, increases robustness in diversifying illumination intensities and changing expression of the captured face. It takes into account the median of the neighbourhood pixels rather than the pixel itself. The grey value of every pixel is replaced by the median of all the neighbourhood pixel values. Then the features are extracted, and a histogram representing the original image is saved in our face database. A new dataset was created for this implementation, which involved recording face data of 100 different people (60 images of each person captured at different angles). Whenever a face is detected in the footage, a histogram is constructed for the image and compared with the histograms in the face database. In this face recognition system, we consider the matches valid only if the algorithm returns a match with a confidence greater than a set threshold (=60%).

II. FACE DETECTION AND RECOGNITION

It is pretty easy for humans to recognize faces, but many variables affect face recognition accuracy and identification for computers, such as illumination, image resolution, etc. In this research, we use Haar Cascade to detect and extract faces from video surveillance and the Local Binary Pattern Histogram Algorithm to ascertain the identity of the extracted faces.

A. Extraction of Faces – Haar Cascade

Haar Cascade algorithm, sometimes referred to as Viola-Jones Face Detection Technique, is one of the oldest ubiquitous face detection algorithms. It is an Object Detection Algorithm that is used to identify faces in real-time video footage. The algorithm is based mainly on edge or line detection features introduced by Viola and Jones in their inquiry "Rapid Object Detection using a Boosted Cascade of Simple Features," published in 2001. We have designed the algorithm to capture images at a constant rate (1 shot every 2 seconds) from the live video, detects and picks faces from each of the pictures.

Viola and Jones introduced the idea of Haar features, as shown in the picture. The unique features make it easy to identify the lines and edges in the image and distinguish between areas where there is an abrupt change in pixel intensity. The algorithm will calculate the value for each pixel in the image and identify boundaries that divide darker areas from the lighter areas of the picture.

Fig.2 An illustration of extraction of face image using Haar Cascade Algorithm

If the pixels have a value close to 1, it represents the dark areas of the image, and if it is close to 0, it means the lighter side. Each of these points will determine the significant parts of the picture. The algorithm mainly focuses on finding the sum of all pixels on the lighter side of the image and the sum on the darker side of the image and compute their difference. This difference is called the Haar Value. If an edge separates an embodiment's dark and light sides, the haar value will be closer to 1, else 0. The algorithm must traverse the entire image to detect all of its edges, giving us Haar Features.

A new idea called the Integral image is used to tackle tedious calculations involved in identifying Haar Features. The integral image is calculated from the original image by computing each pixel in the Integral image as a sum of all pixels lying in its left and top in the original image. The bottom-right pixel in the Integral image is the sum of all pixels in the original image.

Only four integral additions are required to detect any Haar feature, thereby drastically decreasing the time complexity. Therefore, the number of sums to be computed does not depend on the number of pixels.

To get Haar Value, we calculate,

$$\text{Haar Value} = \frac{\text{Sum of pixels in dark area}}{\text{Number of pixels in the dark area}} - \frac{\text{Sum of pixels in light area}}{\text{Number of pixels in the light area}}$$

The amount of the features detected at this point will be very high, some of which are not relevant to the facial features. Here, the algorithm requires additional support, a Feature Selection technique that identifies better-performing features and eliminates unnecessary ones. This technique is called Adaboost, where all the features are applied to the images separately to create Weak Learners. These Learners are designed in such a way that they incorrectly classify only a minimum number of images. These Learners perform merely better than a random guess. These features will be significantly reduced if this technique is deployed.

Once the faces are extracted from the original image, they are fed to the Local Binary Pattern Histogram Algorithm, where an already trained model identifies the input faces.

B. Recognition of Faces obtained from Haar Cascade – Median Local Binary Pattern Match Algorithm

Even though many deep learning networks perform the process of face detection, LBP, which was first described in 1994, stills hold its fort being simple, efficient, yet accurate. It is a powerful texture operator that labels the pixels of an image by juxtaposing it with the pixel's neighbourhood and putting it as a binary digit.

 DOI : 10.35291/2454-9150.2021.0553

ISSN : 2454-9150 Vol-07, Issue-07, OCT 2021

By amalgamating this LBP operator with histograms, we can depict face images with a simple data vector. Perhaps the most crucial reason for its omnipresence in real-world applications is that even if illumination variations cause grayscale changes, it performs adroitly. Another valuable property is the simplicity of its computations, making it feasible to analyse images in challenging real-time settings.

LBPH makes use of 4 parameters to achieve the purpose:

1. Grid X: This parameter defines the number of cells in the horizontal direction, which we choose as per requirement. Greater is the number of cells selected; finer will be the grid, thereby increasing the dimensionality of the resulting vector. It is usually set to 8.

2. Grid Y: This parameter is similar to Grid X but represents the number of cells in the vertical direction.

3. Radius: This parameter is generally used to build the circular local binary pattern and represents the radius around the central pixel. It is usually set to 1.

4. Neighbours: This parameter identifies the number of points to be selected to construct the circular binary pattern. Higher is the number of neighbours chosen, the higher would be the complexity of the resulting computation. It is usually set to 8.

We used a sample dataset of 100 people, compiled by recording 60 face images in different angles and illumination intensities. Their pictures were registered manually using Haar Cascade to train the algorithm. The algorithm associates each person and their photos with a unique ID later used to designate a match, if any.

To begin with, the LBPH constructs an Intermediate image, which renders the original image in a sophisticated way by highlighting various facial characteristics. The algorithm uses an idea of a sliding window which is based on the parameter's radius and neighbours.

To better differentiate different faces, the image is first converted into grayscale. This step is performed both while recording the face data and when we are trying to identify a detected face—sliding windows of dimensions of 3x3 traverse through the entire image where the profile has to be determined. Within this 3x3 matrix, we take the central pixel and compare it with the neighbouring ones. For each of these neighbours, we set a value comparing it with the central pixel, which acts as a threshold; The pixel is set to 1 if it exceeds the median value, and it is set to 0 if it is less than that.

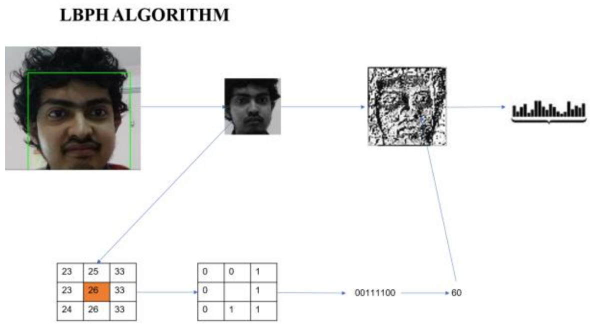

Fig.3 An illustration of constructing a histogram using MLBPH

But in the case of Median LBPH, there is a slight deviation from the above step; We take the median of all the values in the matrix and assign it to the central value before the comparison occurs. The designation of binary values to the pixels in the matrix would be a better representation and thus give higher accuracy. The resultant matrix after this step would only hold binary values. These values are then concatenated to obtain a new binary value. The consequent binary value will then be converted into a decimal value and supersedes the central value of the matrix. This process is repeated for the entire image, and binary values are computed. This image now obtained is a better representation of the intended attributes.

The algorithm then computes a histogram representing each region based on the pixel intensity of all points in the area. Then, the histograms of different regions are concatenated to give a much bigger histogram. The resulting histogram represents the characteristics of the image original image.

Each image from the training dataset is represented by its histogram. We perform this process again while we try to recognize the face input, which is received from the Haar cascade. The newly obtained histogram is matched with all the histograms obtained from the training dataset. The ID of the image which is closest to the input is returned as a match. To determine the proximity of the histograms, we make use of Euclidian distance. The algorithm also returns the confidence based on the Euclidean distance. The lower is the confidence, the better the match. A threshold can be set on confidence to determine if a person has been identified with sufficient certainty.

III. IMPLEMENTATION

The project developed is end-to-end criminal data management and live recognition surveillance. The description of various functionalities can better be explained by splitting the system into two modules.

A. Registration

This module includes adding a new criminal to the already existing data set and re-train the algorithm. When the criminal does not already exist in the central database, the personnel perform registration. After the criminal's data is entered during registration, the page provides

 DOI : 10.35291/2454-9150.2021.0553

functionality to record the face data of the criminal and train the algorithm so that a histogram can be created and registered under the convict's name. Whenever he appears on live surveillance, and there's a match, a notification will be sent to the authorities.

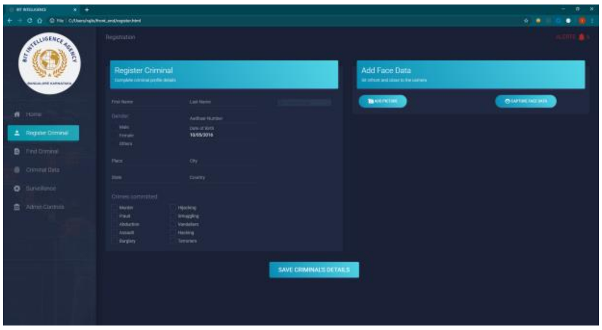

Fig.4 Registration Page

B. Surveillance and Identification from footage

This module covers two functionalities - recognition of a criminal from live surveillance and recognition from an already recorded video footage., The application forks a separate Thread for every camera that the system monitors. The reason for creating individual Threads is to constantly keep track of the surveillance footage from each camera by extracting faces from the real-time footage of the cameras every three seconds. The algorithm builds the histogram for the extracted faces and compares them with the existing histograms to find a match. The module also provides functionality to recognize faces from old or already recorded footage. Whenever a potential match is found, an alert is sent to authorities through email and a text message. The authorized personnel can upload footage to run a check on the database for potential matches.

Fig.5 Page illustrating a sample surveillance timeline

IV. RESULTS

Using face data of up to 100 different people taking 60 pictures for each person in different angles and brightness, we conducted various experiments based on the different illumination angles or the inclination angle at which a person appears in the surveillance. The results are as follows:

1) This table plots distance against different views of the person from the capturing point. As we can expect, there is a commiserate decrease in the model's ability to recognize or detect faces as the distance increases.

TABLE I
TABLE INDICATING SUCCESS AND FAILURE AT DIFFERENT ANGLES AND ILLUMINATION FOR ONE PARTICULAR CASE (D: DETECTED, R: RECOGNIZED)

	FRONT VIEW		SIDE VIEW		DOWNWARD VIEW	
Distance(in meters)	D	R	D	R	D	R
1	Yes	Yes	Yes	Yes	Yes	Yes
1(50% brightness)	Yes	Yes	Yes	Yes	Yes	Yes
2	Yes	Yes	Yes	Yes	Yes	Yes
2(50% brightness)	Yes	Yes	Yes	Yes	Yes	No
3	Yes	Yes	Yes	No	Yes	No
3(50% brightness)	Yes	Yes	Yes	No	Yes	No
4	Yes	Yes	Yes	No	No	No
4(50% brightness)	Yes	Yes	Yes	No	No	No
5	Yes	Yes	No	No	No	No
5(50% brightness)	Yes	No	No	No	No	No

Fig. 7 Frames from 3m distance

Fig.8 Frames from 1m distance

2) We also compared the results of the existing LBPH

 DOI : 10.35291/2454-9150.2021.0553

algorithm with our Median LBPH algorithm. These results point out that there is a sufficient increase in MLBPH. The experiment was carried out for 250 frames and was repeated 10 times and their average was recorded as shown in table 2.

TABLE 2
TABLE COMPARING RESULTS FOR 250 CASES OF LBPH AND MLBPH TESTED SIMULTANEOUSLY

Algorithm	Correct	Wrong	Recognition %
LBPH	205	45	82%
MLBPH	216	34	86.4%

V. CONCLUSION AND FUTURE WORK

With the introduction of MLBPH, we have improved accuracy in recognising faces by around 5%. Face recognition has become a fascinating field for researchers. The motivation behind the enormous interest in the topic is the need to improve the accuracy of many real-time applications. The intricacy of the human face and the changes due to different effects make it more challenging to design and implement a powerful computational system for human face recognition.

A citizen database can be incorporated to make this a nation comprehensive law enforcement tool. Fingerprint functionality can be added to the application to make the search process more manageable. Fingerprints obtained from the crime scene can be run against the citizen database to determine the criminal. Using their photo from the citizen database, we can search for that criminal through surveillance.

REFERENCES

[1] M. Kaumalee Bogahawatte, Shalinda Adikari, Intelligent Criminal Identification System The 8th International Conference. April 2016.

[2] Banu Wirawan Yohanes, Reva Diaz Airlangga, Iwan Setyawan, Real Time Face Recognition Comparison using Fisherfaces and Local Binary Pattern. 2018 4th International Conference on Science and Technology.2018

[3] Hengliang Tang, Yanfeng Sun, Baocai Yin, Yun Ge, Face Recognition Based on Haar LBP Histogram. 2017 3rd International Conference on Advanced Computer Theory and Engineering.2017

[4] Aftab Ahmed, Farah deeba, Jiandong Guo, Fayaz Ali, LBPH Based Improved Face Recognition At Low Resolution. 2018 International Conference on Artificial Intelligence and Big Data. 2018

[5] Jyoti Nautiyal, Shivali Gahlot and Pawan Kumar Mishra, "An automated technique for criminal face identification using biometric approach", Conference on Advances in Communication and Control Systems 2013(CAC2S 2013)

[6] Unsang Park, Hyun-Cheol Choi, Anil.K Jain and Seong-Whan Lee, "Face tracking and recognition at a distance: A coaxial and concentric PTZ camera system", IEEE transactions on information forensics and security, vol.8, no.10, October 2013

[7] Yongqiang Li,ShangFei Wang, Yongping Zhao and Qiang Ji, "Simultaneous facial feature tracking and facial expression recognition", IEEE Transactions on image processing, vol.22, no.7, July 2013

[8] Reza Moradi Rad,Abdolrahaman Attar,Reza Ebrahimi atani, "A Robust Face Recognition method using edge based features", 2012 IEEE Symposium on Computers and Informatics. "PDCA12-70 data sheet," Opto Speed SA, Mezzovico, Switzerland.

[9] J. P. Bellhumeur, J.Hespanha, D. Knegman. Eigenfaces vs fisherfaces: Recognition using class specific linear projection IEEE Transactions on Pattern Analysis and Machine Intelligence. Special Issue on Face Recognition, 1997, 17 (1) :711-720.

[10] J. CHAO W L, DING J J, LIU J Z. Facial expression recognition based on improved local binary pattern and class-regularized locality preserving projection. Signal Processing, 2015, 117:1-10.

[11] J. HU Liqiao, QIU Runhe. Face recognition based on adaptive weighted HOG. Computer Engineering and Applications, 2017, 53(3): 164-168.

[12] J. Hai qiang LONG,Tai zhe TAN. Computer Simulation, 2017, 34(1):322-325.

[13] J. Yu yan JIANG, Ping LI, Qing WANG. Labelled LDA model based on shared background topic. Acta Electronica Sinica, 2015, 2013, (9): 1794-1799.

 DOI : 10.35291/2454-9150.2021.0553