**FACE RECOGNITION WITH VOICE APPLICATION**

**D MADHUSUDAN1**

Assistant Professor, Department of Electronics and Communication Engineering,

St. Peter’s Engineering College, Hyderabad, Telangana, India.

Email:madhus7001@gmail.com1

**ABSTRACT:**

Facial recognition may be a non-intra-operative method of biometric identification and useful for different applications. The important time implementation of the algorithm with adequate accuracy is required, with hardware timing into consideration. This research deals with the implementation of machine learning algorithm for real time facial image recognition. The face recognition software uses algorithms to match a digital image captured through a camera, to the stored face print so on virtually authenticate an individual's identity. The Haar-Cascade method was one among the primary methods developed for face recognition. The HOG (Histogram of Oriented Gradients) method has worked very effectively for visual perception and thus suitable for face recognition also. Both the methods are compared with Eigen feature-based face recognition algorithm. The face recognition model is implemented to detect and recognize faces in real-time by means of Raspberry Pi and Pi camera for the user defined database additionally to the available databases. After recognizing the face, we'll interface the machine learning algorithm with the voice by using the libraries (GTTS, Playsound). In this way it'll detect and recognize and can answer the person.

**KEYWORDS:** Haar-Cascade, Trainer, Histogram of Oriented Gradients (HOG), Support Vector Machine (SVM), Voice Recognition, Google text to Speech (GTTS)

1. **INTRODUCTION**

Face recognition is that the task of identifying an already detected object as a known or unknown face. Often the matter of face recognition is confused with the matter of face detection Face Recognition on the opposite hand is to make a decision if the "face" is someone known, or unknown, using for this purpose a database of faces so as to validate this input face. Face detection may be a computer vision problem that involves finding faces in photos. It is a trivial problem for humans to unravel and has been solved reasonably well by classical feature-based techniques, like the cascade classifier. One example is that the Multi-task Cascade Convolutional Neural Network or MTCNN for brief. As researcher interest in face recognition continued, many various algorithms were developed, three of which are well studied in face recognition literature.



**Figure 1: Integration of Face Recognition with Voice Application**

The various functional blocks of the project in this system are listed below:

* **USB camera –**It is an electronic component used to capture the image from the user.
* **Frame/Image –** It is a stored file that is taken as a pre-defined input from the usb camera to detect faces of the user.
* **Detect Faces –** Identify and checks whether the image given, actually has a face of a person or not.
* **Compute 128-d face embeddings -**To extract the 128-d feature vectors called embeddings that quantify each face in an image[1]
* **Compute 128d vector –** To calculate the 128-d feature vectors for all the data samples in the data base.
* **Recognize face –**Verifying data samples from the data set and identify whether the image is present in the data set or not
* **Message –** Voice message which is given as output after recognizing a face
* **Talk with me –** An application developed to interact with a person
* **Echo Dot –** An electronic speaker embedded with artificial intelligence
1. **LITERATURE WORK**

The author expressed two main approaches for the face- recognition process the two being geometrical and photometrical. The geometrical is based on the features of the image that is taken as input whereas the photometrical is based on the view-based analysis. Although there were many algorithms introduced only three are very well studies and researched, the three being Principal Components Analysis (PCA), Linear Discriminant Analysis (LDA), and Elastic Bunch Graph Matching (EBGM).[2] The author expressed that there are four basic techniques that are used in the face detection process that is feature based, appearance based, knowledge based, template matching. In the first method i.e. feature based it depends on the structural positions and the images of the facial and non-facial regions present. In the second method i.e the appearance based we find that the image is detectable only by the process of statistical analysis and machine learning algorithms. In the third method i.e. the knowledge based we find that the rules are according to the human knowledge such as whether the image has two eyes, one nose etc. In the fourth i.e the template matching we find that the method used a pre-defined set of rules or templates and then compare the given image with the predefined templates. [3] In geometric based technique, the local features include mouth, eyes, eyebrows, and nose is at first extracted and computed from face images, which is used for classification. The best of geometric-based methods is Active Appearance graph Models (AAM) [4]. The photometric based technique has different recognition algorithms that are applied to the various methods available to users by using the photo-face database. Two different techniques inside are retrieved by the albedo image and the 3-D facial geometry indicated by the depth of the normal field [5]. In this author explains how to use OpenCV to conduct face recognition**.** To build a face recognition software, firstly face detection is performed and then extraction of face embeddings is carried out from each and every face that is used in deep learning. After that the model trains a face recognition reproduction on the embeddings, and then finally identify the faces in the images with OpenCV [6].

1. **STUDY ON VARIOUS APPLICATIONS AND METHODOLOGY**

Face recognition systems identify people by their face images. Face recognition systems establishes the presence of an official person rather than just checking whether a legitimate identification (ID). The face recognition system directly compares the face images of the persons and doesn't use ID numbers to differentiate one from the others. When the highest two matched faces are highly similar to the query face image, manual review is required to create sure they're indeed different persons so on eliminate duplicates [7].

*Access Control*: In many of the access control applications, such as office access or computer logon, the scale of the group of people that require to be recognized is comparatively small. The face pictures also are caught under natural conditions, such as frontal faces and indoor illumination. The face recognition system of this application can do with high accuracy without much cooperation from user. The below are the instance.

Face recognition technology is deployed to observe continuously who is before of a computer terminal. It allows the user to leave the terminal without closing files and logging out. When the user leaves for a predetermined time, a pattern covers up the work and disables the mouse & keyboard. When the user comes back and is recognized, the figure clears and the previous session appears because it was left as it was. The other user who tries to logon without authorization is denied. [8]

*Security*: Now-a-days, security is a primary concern at airports and for airline personnel, department and passengers. Airport protection systems that use face recognition technology have been implemented at many airports round the world. The system is meant to alert the airport public safety officers whenever a personal matching the looks of a known terrorist/suspect enters the airport's security checkpoint. Anyone recognized by the system would have further investigative processes by the police. These days computer security has seen other application of face recognition technology. To deny someone else from altering files or making illegal transactions with the unknown when the authorized individual leaves the pc for a brief time, users are continuously authenticated, checking that the individual ahead of the pc screen or at a user is that the same authorized one that logged in. Image database investigations Searching image databases of licensed drivers benefit recipients, missing children, immigrants and police bookings.[9]

*General identity*: Verification, banking, e-commerce, national IDs, passports, employee IDs, identifying newborns, polling registration. [10]

*Surveillance*: Public places, security surveillance by face recognition system, other divisors all make the deployment of face recognition systems for giant scale surveillance a challenging task. Face orientations are subsequent are some examples of face-based surveillance. [11].

Start

Original Faces training set

|  |  |  |
| --- | --- | --- |
|  | E=eigenfaces (training Set) |  |

|  |  |  |
| --- | --- | --- |
|  | W=weight (E,training Set) |  |

Input unknown image X

|  |  |  |
| --- | --- | --- |
|  | Wx = weight(E,X) |  |

D=avg(distance(W,Wx))

 D<0?

End

Store X and Wx

X is not a face

X is a Face

Fig 2 Flow chart of eigenface-based algorithm [2]

There are three methods which can be used for the face recogniton.1) holistic matching methods, 2) Feature- based 3) Hybrid methods. [05]. In the holistic approach entire face is taken as an input. Eigen faces are a best example for the holistic matching methods [06]. Local features such as eyes and mouth and nose are extracted and their addresses i.e the locations are aligned into the structural classifier [06]. Hybrid is a combination of both holistic and also the feature extraction techniques. Any 3d image that is caught is actually an important ingredient used in the hybrid methods [06]. The classical face recognition algorithms are based on the external structure. The eigenface images are projected into a linear subspace [07]. These classical methods are a failure because they fail to represent when a large variation is found and an illumination expression occurs [07]. Artificial neural networks are also used in the face recognition process. They are different when compared to a classical method because they are used to solve a non-linear problem. In order to solve the pattern and identify a human face, a chaotic non-convergent network is used [08]. A non-negative sparse coding to learn facial features using normalized cross-correlation is applied. [08].

The Histogram of oriented Gradients is descriptors that are used in the optimization process and also in the computer vision. They are used in the pattern recognition are recognition of visual objects i.e faces. [09]

Fig 3: The Histogram of oriented Gradients [12**]**

HOG is reminiscent of edge orientation histogram, SIFT descriptor and shape context. They are calculated in dense cells that are normalized and overlapped with the local contrast histogram in the direction of the image gradient to improve the performance of the detector. Therefore, due to the distribution of local intensity gradients, this feature set is very effective for other types of shape- based objects (ie, face detection), even without any knowledge of the corresponding gradients. To extract the HOG descriptor, first calculate the appearance of the edge direction in the local neighborhood of the image. This means that the image is divided into small connected areas, called cells, and edge-direction histograms are calculated for each. According to whether the gradient is signed or unsigned, the channels of the histogram are distributed from (0 °-180°) or (0°-360°). In order to compensate for the light, the histogram count is normalized by accumulating a measure. The local histogram energy measurement of the connected area and then using the obtained result to normalize all cells in the block. Finally, the combination of these histograms represents the descriptor HOG. [09]

Speech recognition technology provides ways in which computer technology can be utilized to tasks which uses natural (human) languages or speech. Now we have entered in the era of computer science concerned with the interactions between computers and human (natural) languages. Voice recognition technology uses audio input for entering data. Speaking into a microphone and it produces the same result as typing words manually with a keyboard. Voice recognition software is developed with an internal database of recognizable words, sentences and phrases. The program matches the audio frequencies of speech with corresponding entries in the database. The study began in the 1936 with AT&T’s Bell Labs. Most research was funded and performed by Universities and the U.S. Government especially by the Military and DARPA. [10]

How does a computer convert spoken words or speech into the data which it can understand and then executes accordingly? In speech recognition technology a person has to speak in a microphone which will be the input. The electrical signal from the microphone is digitized by an analog-to-digital converter, and is stored in memory.[11] To determine the meaning of the input which in the form of voice, the computer matches the voice input with a sample voice format, or voice sample that has a known meaning to the computer. During the speech recognition a stream of audio samples are entered as input to the program. Before that, the speaker database must be ready and comparable feature must be computed from the input audio signal. The feature vectors are then matched to the contents of the database. This technique is similar to the traditional command inputs which are given from a keyboard. The program contains the input templates, and attempts to match these templates with the actual input provided to the computer. In fact, in the early 1990s, the best speech recognizers were giving a 15% error rate on a relatively small 20,000 word dictation task. Now though, that error percentage has dropped to as low as 1-2%, although this can vary greatly between speaker to speaker.[11]

The application of speech recognition is playing a major role in helping handicapped and disable people. In the last ten years, we can observe a lot of improvement in the performance of speech recognizing devices and current technology in relation to the needs of the disabled population. Blind people were among the people who have the most advantage from speech recognition computer technology. We use simple machines which read the words on early text-based computer screens and convert them into synthesized speech. It also takes the input from the commands spoken by the blind person which helps the blind person to operate the computer more easily and in an efficient manner. Not only for blind people, the speech recognition software is also useful for people who are having difficulty with using their hands.[12][13]

1. **WORKING METHODOLOGY**

The working methodology is discussed with a flowchart below.

|  |
| --- |
|  Collecting data sets from diff persons |
| Data Pre-processing |  |
|  |

5

Application Building

Build python code

Create data sets

Import libraries

Reading dataset

Filling missing values

Converting textual data to numerical data

Splitting the dataset

Convert numerical data into binary

Split data to train and test

Feature Scaling

Save model

1. **IMPLEMENTATION**



Fig 4: Flow chart for Face detection

1. **WORK FLOW**

**DATA SET CREATION**

* + The first task we are perform is DATA SET CREATION.
	+ This step is important for detecting the faces. During this step we crop the faces out of the image and extract features from it which are later used for recognizing the faces.
	+ The detect Multiscale function may well be a general function that detects objects. Since we are calling it on the face cascade, that’s what it detects.
	+ The first one is grayscale image. The second is that the multiplier.
	+ Since some faces is additionally closer to the camera, they'd appear bigger than the faces within the rear. the dimensions factor compensates for this.
	+ The detection algorithm uses a moving window to detect objects. Min Neighbors defines what number objects are detected near this one before it declares the face found.



Fig 5: Images in DATASET folder

**TRAINER PROCESS**

* + - Second process in face recognition is TRAINING THE MACHINE.
		- For this we use both HAAR cascade and training file. Local Binary Pattern (LBP) may be a simple yet very efficient texture operator which labels the pixels of a picture by thresholding the neighborhood of every pixel and considers the result as a binary number. Suppose we've got a facial image in grayscale.
		- Consider a little a part of this image as a window of 3x3 pixels containing the intensity of every pixel (0~255). Then, we'd like to require the central value of the matrix to be used because the threshold. This value is accustomed define the new values from the 8 neighbours. For each neighbour of the central value (threshold), we set a brand new binary value.
		- We set 1 for values equal or more than the brink and 0 for values less than the brink. Now, the matrix will contain only binary values (ignoring the central value.) we'd like to concatenate each binary value from each position from the matrix line by line into a replacement binary value (e.g., 10001101).
		- Note: Some authors use other approaches to concatenate the binary values (e.g., clockwise direction), but the ultimate result are the identical.
		- Then, we convert this binary value to a decimal value and set it to the central value of the matrix, which is truly a pixel from the initial image. At the top of this procedure, we've got a brand new image which represents better the characteristics of the initial image.



Fig 6: Local Binary Pattern Process

**TRAINER ALGORITHM**

* + - * **Step 1**: Initially we import the files which are required for training the machine i.e. CV, OS, NumPy and from Python Imaging Library (PIL) we want to import images.
			* **Step 2**: For training the machine to recognizer the face we create a face recognizer using face. LBPH Face Recognizer.
			* **Step 3**: Creating the trail for dataset.
			* **Step 4**: Create a detector by using Haar cascade.
			* **Step 5**: We use function to urge the pictures and label
			* **Step 6**: When the condition is completed the face sample and ids return.
			* **Step 7**: Now the recognized faces and ids are sent to trainer for training.
			* **Step 8**: Finally, the trained faces and ids are saved in trainer. Yml which is used for detecting the face.

**DETECTOR PROCESS**

* + - Detector code algorithms target the detection of frontal human faces.
		- It's analogous to image detection within which the image of an individual is matched bit by bit.
		- Image matches with the image stores in database.
		- Any facial feature changes within the database will invalidate the matching process.
		- Each possible face candidate is normalized to cut back both the lighting effect, which is caused by uneven illumination; and also the shirring effect, which is thanks to head movement.



Fig 7: Face Detection Features

1. **RESULTS**



Fig 8: Output image

The above Fig8 represents the final output of the face recognition project. In this figure it represents the person who is already present in the data set and when the person is in-front of the camera, it recognizes and displays the name of the person on the display screen and responds with a voice message by using pyaudio.

1. **CONCLUSION**

In current work we created the framework to assess the confront discovery and acknowledgment strategies which are considered to be a seat stamp. A few strategies performed reliably over distinctive datasets though other strategies carry on exceptionally haphazardly be that as it may based on normal test comes about execution is assessed, twenty-one data samples been utilized for this particular reason. Confront location and acknowledgment method’s result summery is shown in the fig3. In current framework Haar-like [13] highlights detailed generally well but it has much wrong discovery than LBP [14] which might be consider being a future work in reconnaissance to diminish untrue location in Haar-like [15] highlights and for the acknowledgment portion gabor

[16] is detailed well as it’s qualities overcomes the complexities arised in datasets.

1. **REFERENCES**
2. R. Jafri, H. R. Arabnia, “A Survey of Face Recognition Techniques”, Journal of Information Processing Systems, Vol.5, No.2, June 2009.
3. C. A. Hansen, “Face Recognition”, Institute for Computer Science University of Tromso, Norway.
4. M. D. Kelly. Visual identification of people by computer. PhD thesis, Stanford University, Stanford, CA, USA, 1971.
5. T. Kanade. Computer Recognition of Human Faces, 47, 1977.
6. W. Zhao, R. Chellappa, P. J. Phillips & A. Rosenfeld, “Face recognitions literature survey”, ACM Computing Surveys, Vol. 35, No. 4, December 2003, pp. 399–458.
7. C. Gonzalez, R. E. Woods, S. liddins, "Digital Image processing Using MATLAB".
8. S. Suhas, A. Kurhe, Dr.P. Khanale, “Face Recognition Using Principal Component Analysis and Linear Discriminant Analysis on Holistic Approach in Facial Images Database”, IOSR Journal of Engineering e-ISSN: 2250-3021, p-ISSN: 2278-8719, Vol. 2, Issue 12 (Dec. 2012), ||V4|| PP 15-23
9. M. A. Turk and A. P. Pentland, "Face Recognition Using Eigenfaces", 1991.
10. S. Asadi, Dr. D. V. Subba R. V. Saikrishna, "A Comparative study of Face Recognition with PCA and Cross-Correlation Technique", IJCA(0975-8887), Volume 10- No.8, November 2010.
11. E. A. Abusham, A. T. B. Jin, W. E. Kiong, "Face Recognition Based on Nonlinear Feature Approach", American Journal of Applied Sciences, 2008.
12. A. Nigam, P. Gupta, "A New Distance Measure for Face Recognition System", 2009 Fifth International Conference on Image and Graphics
13. A. Wager, Frances Wickham Lee and John P. Glaser.Managing health care information systems: A practicalapproach for health care executives (I), Jossey-Bass: A Wiley imprint San Francisco, 2005.
14. Rajeev Agarwal - Voice Browsing the Web for Information access: <http://www.w3.org/Voice/1998/Workshop/RajeevAgarwal.html>
15. T. Mita, T. Kaneko, O. Hori, Joint Haar-like Features forFace Detection, “Proceeding of the International Conference on Computer Vision”, 1550- Tenth IEEE 5499/05 ©2005 IEEE.
16. T. Ahonen, A. Hadid, M. Peitikainen, Face recognition withlocal binary patterns. “In Proc.of European Conference ofComputer Vision”, 2004.
17. L. Wiskott, M. Fellous, N. Krger, and C. Malsburg, Facerecognition by elastic bunch graph matching, “IEEE Trans”,on PAMI, 19:775–779, 1997