International Journal of Research in Engineering and Science (IJRES)
ISSN (Online): 2320-9364, ISSN (Print): 2320-9356
www.ijres.org Volume 10 Issue 4 ǁ 2022 ǁ PP. 93-98

Face Recognition System

Batul Khambata
Department of Information Technology
Vidyalankar Institute of Technology (Wadala) Mumbai, India

Vaibhav Rajani
Department of Information Technology
Vidyalankar Institute of Technology (Wadala) Mumbai, India

Harsh Sangani
Department of Information Technology
Vidyalankar Institute of Technology (Wadala) Mumbai, India

Ameya Manjrekar
Department of Information Technology
Vidyalankar Institute of Technology (Wadala) Mumbai, India

Prof. Akshay Loke
Department of Information Technology
Vidyalankar Institute of Technology (Wadala) Mumbai, India

Abstract – *Ever since the addition of a Face Detection feature in smartphones and laptops, it has become one of the most popular tools to identify a person. There have been numerous algorithms developed to pin-point all facial features in order to recognize someone. Histogram of Oriented Gradients (HOG) is one of the most popular descriptors for facial recognition. The objective is to come up with a Face Recognition System that can be used on all types of computers – both low and high end and to integrate it in a Web Application built using Python and Flask.*

Keywords – *Histogram of Oriented Gradients, Face Detection, Python, Flask, OpenCV, descriptor.*

Date of Submission: 02-04-2022 Date of acceptance: 16-04-2022

I. INTRODUCTION

Research says that in our entire lifespan, we can recognize a number of faces even years after separation from them. This skill, if transferred somehow to a machine, will be extremely effective in terms of the number of faces and accuracy of identifying them. After much trial and error, there now are various computational models of face recognition. They are a revelation because they have a practical application in real life scenarios. These machines can be used for tasks like CCTV identification, criminal identification, identity verification, etc. [10]

Face Recognition has been around since a long time but the number of applications of this technique has increased drastically this decade. Popular social media applications like Snapchat and Instagram use Face Recognition in various filters in order to enhance user experience and keep the user engaged. [1]

In this project, we have visited various Face Recognition algorithms and chosen one of the most effective and easy to access algorithms out there. The main aim is to make the system accessible to everyone regardless of the computing power of their machines. In order to make it easily accessible, we have also integrated it into a web application.

II. PROPOSED SYSTEM

After an exhaustive debate, the method we decided to go for is called Histogram of Oriented Gradients (HOG). It is generally used in pattern recognition and image processing. The main advantage of this method is the splendid accuracy that it provides considering the amount of computing power it uses. While other methods like Convoluted Neural Networks may provide a better accuracy, they need a considerable higher processing power, that is not available with everyone. Since that defeated one of the main objectives of our projects, we decided to proceed with the Histogram of Oriented Gradients method. [2]

Another advantage of the HOG method is its ability to identify faces from a wide range of poses and different illumination backgrounds. In order to extract these HOG descriptors, first, the occurrences of edge orientations in an image are counted. Histogram channels can spread over 0 to 180 degrees or 0 to 360 degrees. [9]

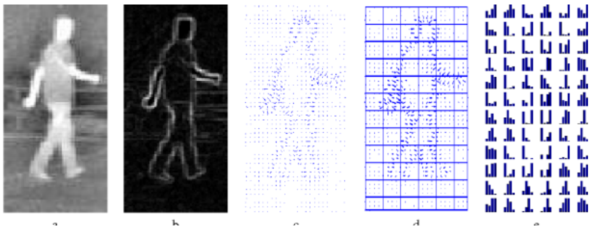


Fig. 1: Characterization of an image using HOG: Original image (a) (scaled down), gradient norm (b), gradient orientation (c), cell splitting (d) and histogram computation (e). [3]

III. ARCHITECTURE/FRAMEWORK

The proposed automated system is based on face recognition. Using the camera, we will capture all frames with subject(s) in it and compare it with our database. If a match is found, his/her name will be displayed. An efficient face detection algorithm enhances the performance of face recognition systems. Some of the algorithms proposed for face detection are Face geometry-based methods and Machine learning based methods. Face region is extracted and pre-processed for further processing. This pre-processing step involves with histogram equalization of the extracted face image and is resized. Histogram Equalization is the most common Histogram Normalization technique.[11][12] This improves the contrast of the image by making it clearer as it stretches the range of the intensities in an image. As we chose face recognition-based system enrolment of every individual is required i.e., we have to take the images of individuals and add it to the dataset. [8]

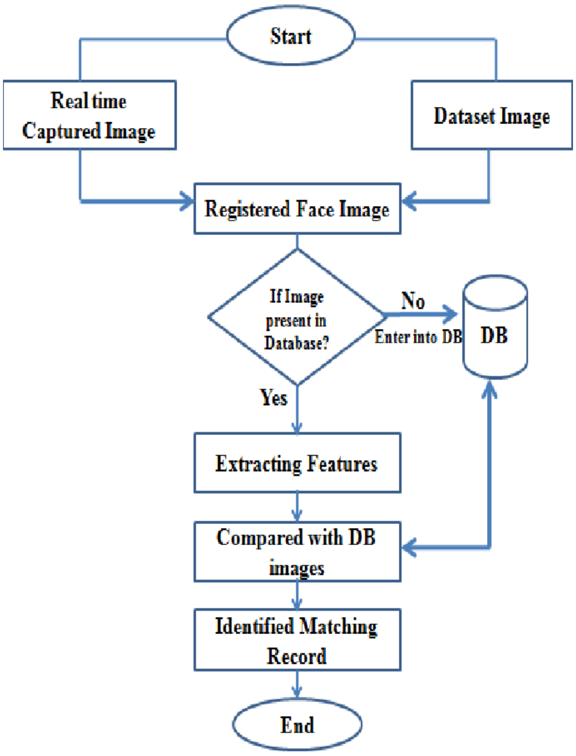


Fig. 2: Flowchart

Flowchart Description:

- All fames are captured by a webcam in real time.
- The faces are extracted from each frame by the algorithm.
- Features from these faces are extracted and are compared with the features of the faces that are present in the database.
- If a match is found, the corresponding name is displayed on the screen.
- If a match is not found, it simply displays “Unknown Image”.

IV. METHODOLOGY

This section focuses on the main logic and methodology behind the project.

A. Recognition and Feature Extraction: A face detection algorithm applies to identify the human faces in that image. Once the image is spotted by the algorithm, it extracts the features of the image. As mentioned earlier, we have used the HOG method to detect the faces in an image.

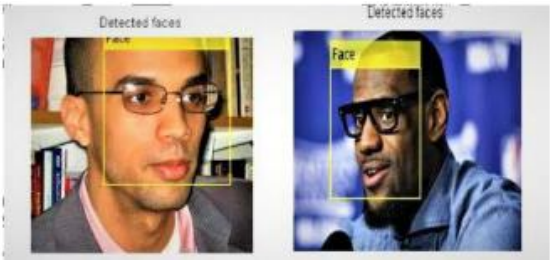


Fig. 3: Face Detection

B. Face Positioning: Out of the 68 specific points on a human face, the image is positioned and a python script is used to automatically detect these 68 points whilst keeping the image intact. [4]

C. Face Encoding: Once the face is detected and positioned, all unique features are extracted. The points detected in step 2 are then used for encoding the image.

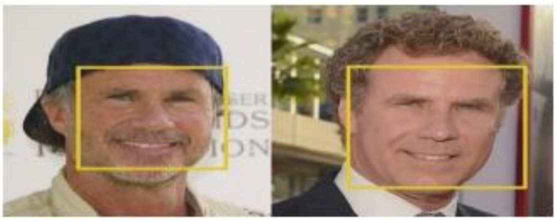


Fig. 4: Detecting Face Landmarks

D. Face Cross-Checking: After extracting the features and encoding, these features are then compared to the existing database using a python script. The face_recognition library is responsible for this comparison, and if the images match, it will return the name of the person.

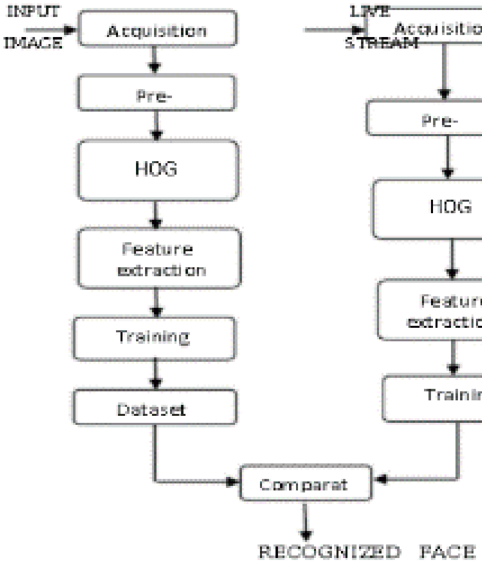


Fig. 5: Methodology

V. IMPLEMENTATION

The various tools used for the implementation of our project are:

- Front–End: For the front-end development, we have used python language and its popular libraries which are widely used all over the world. [5]
- OpenCV: Open-Source Computer Vision Library is a python library designed to solve computer vision problems.
- Dlib library: It is a general-purpose cross platform software library developed in C++ and it incorporates sufficient AI algorithms and tools for creating complex software programs to solve real world problems. [5]
- NumPy Module: It is a Python package that contains multi-dimensional array/cluster objects and the records for processing of arrays. Using NumPy, a developer can also perform logical and mathematical operations on arrays.
- Face_Recognition: This is another library in Python that is designed to recognize and manipulate faces from Python. It is built using dlib's state-of-the-art face recognition built with deep learning. The model has an accuracy of 99.38% on the Labeled Faces in the Wild benchmark. [6]
- Dataset: We have a locally stored image dataset with all our test cases. It is stored in the images folder.
- Web-Application: A Web Application has been created using Python and Flask in order to deploy the system.
- Flask: Flask is a micro web framework written in Python. It is classified as a microframework because it does not require particular tools or libraries. It has no database abstraction layer, form validation, or any other components where pre-existing third-party libraries provide common functions. [7]

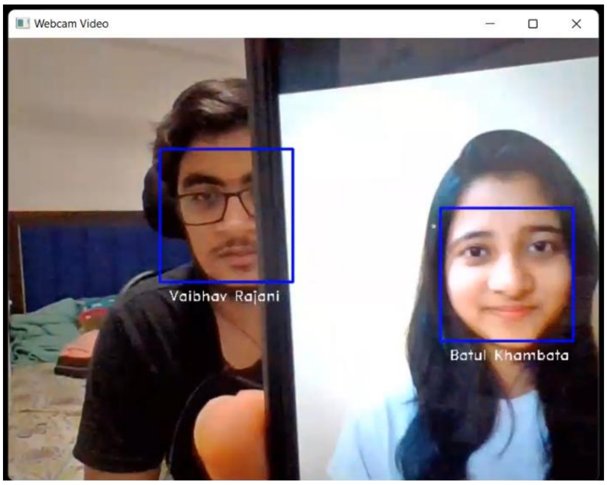
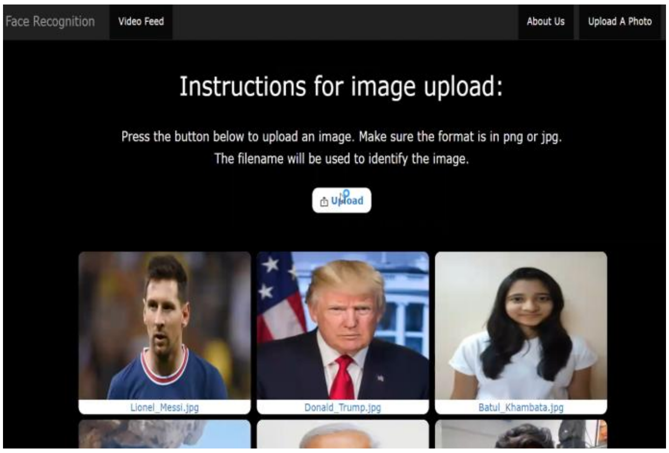


Fig. 6: Detection of a face



Instructions for image upload:

Press the button below to upload an image. Make sure the format is in png or jpg.
The filename will be used to identify the image.

Fig. 7: Image Upload Interface

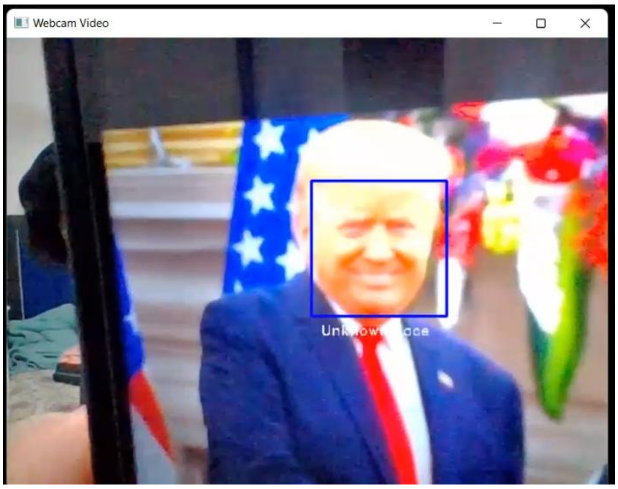


Fig. 8: “Unknown Face” (If the image is not present in database)

VI. SUGGESTED IMPROVEMENTS

There is always room for improvement in all projects. We have tried a minimalistic approach so that it can be accessed using low computational powers too. 100% accuracy of this setup is not guaranteed and there may be a few errors here and there sometimes.

It might also not work with twins or 2 people with extremely similar features. There is always a future scope to enhance these systems with a powerful GPU and hence a better and more accurate algorithm for face detection like CNN.

VII. CONCLUSION

The Face Recognition System is simple, accurate and works efficiently. This system can work automatically once all the faces that need to be recognized are fed to the system. With the help of HOG models, we have successfully achieved high-performance levels in the field of human face recognition, even with multiple faces and different light settings.

REFERENCES

[1]. C. Zhang and Z. Zhang. A survey of recent advances in face detection. Technical report, Tech. rep., Microsoft Research, 2010.

[2]. W. Schwartz, A. Kembhavi, D. Harwood, and L. Davis. Human Detection Using Partial Least Squares Analysis. 2009

[3]. Bertozzi, M., Broggi, A., Rose, M.S., Felisa, M., Rakotomamonjy, A., & Suard, F. (2007). A Pedestrian Detector Using Histograms of Oriented Gradients and a Support Vector Machine Classifier. *2007 IEEE Intelligent Transportation Systems Conference*, 143-148.

[4]. Qutbuddin, Sakina & Larik, Sehrish. (2021). Face Recognition for Automated Attendance using HOG & Machine Learning.

[5]. International Journal of Engineering Applied Sciences and Technology, 2021 Vol. 5, Issue 12, ISSN No. 2455-2143, Pages 233-241 Published Online April 2021 in IJEAST (http://www.ijeast.com)

[6]. Labeled Faces in the Wild benchmark: http://vis-www.cs.umass.edu/lfw/

[7]. Flask WiKi: https://en.wikipedia.org/wiki/Flask_(web_framework)

[8]. Rekha, N., and M. Z. Kurian, 2014. "Face detection in real time based on HOG." International Journal of Advanced Research in Computer Engineering & Technology (IJARCET) 3.4 (2014): 1345-1352.

[9]. Dhanush Gowda H.L, K Vishal, Keertiraj B. R, Neha Kumari Dubey and Pooja M. R., June-2020, “Face Recognition based Attendance System”, International Journal of Engineering Research & Technology (IJERT), ISSN: 2278-0181, Vol. 9 Issue 06.

[10]. Li, Xiang-Yu, and Zhen-Xian Lin, 2017. "Face recognition based on HOG and fast PCA algorithm." The Euro-China Conference on Intelligent Data Analysis and Applications. Springer, Cham.

[11]. Cahit Gurel and Abdulkadir Erden, June 2012. “Design of a Face Recognition System”, The 15th Conference on Machine Design International and Production June 19 – 22, 2012, Pamukkale, Denizli, Turkey.

[12]. Himanshu Tiwari, April 2018, “Live Attendance System via Face Recognition”, International Journalfor Research in Applied Science & Engineering Technology (IJRASET), ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 6.887 Volume 6 Issue IV.