ISSN: 0975-833X

Available online at http://www.journalcra.com

International Journal of Current Research
Vol. 11, Issue, 07, pp. 5190-5192, July, 2019

DOI: https://doi.org/10.24941/ijcr.36008.07.2019

INTERNATIONAL JOURNAL OF CURRENT RESEARCH

RESEARCH ARTICLE

SKETCH BASED FACE RECOGNITION

***Anush Mandya Nagesh, Prajwala Naidu Babu, Sagar T., Sanjana Vidhuran and Dr. Nalini, N.**

Computer Science and Engineering, Nitte Meenakshi Institutre of Technology, Bangalore, India

ARTICLE INFO

Article History:
Received 17th April, 2019
Received in revised form
03rd May, 2019
Accepted 17th June, 2019
Published online 25th July, 2019

Key Words:

Face recognition, Histogram of Oriented Gradients, Convolutional neural networks.

**Corresponding author:*
Anush Mandya Nagesh

ABSTRACT

Face recognition is a task of identifying an individual in an image using a set of images stored in a database. The applications of this activity are vast, few of them include automatic attendance marking system, security issues concerning smartphones and other smart devices, law enforcement and so on. Most of the existing systems focus on image to image face recognition, whereas our system can perform sketch to image recognition. The system can either implement Convolutional Neural Network (CNN) or Histogram of Oriented Gradients (HOG). Apart from recognizing a face in a sketch, this paper will also concentrate on testing the speed of execution of both algorithms. A graph will be determined to show the comparison between the two algorithms.

***Citation: Anush Mandya Nagesh, Prajwala Naidu Babu, Sagar T., Sanjana Vidhuran and Dr. Nalini, N.,* 2019.** "Sketch Based Face Recognition", *International Journal of Current Research*, 11, (07), 5190-5192.

INTRODUCTION

Face recognition plays a prominent role in surveillance and security. The technology makes recognising and convicting criminals far easier than the manual method. The police department uses photos, videos and sketches to identify the offender. This process is usually carried out manually and involves a lot of labour and time if the database is immense. Using a software that performs this task saves resources (Priya Guptaa *et al.,* 2018). There are many ways the system of face recognition can work, but the base idea is to identify a person by comparing few selected features in the input with the database. Basically, the process of face recognition is carried out in two steps. The first step involves, extracting features of an individual from the input image. The second step involves, matching the recognizable extracted input features to the features of the images in the database and then finally present the output (Adrian Rhesa Septian Siswanto *et al.,* 2014). The system by default uses CNN algorithm to perform facial recognition. On explicitly mentioning 'hog' in the input parameters, the system will perform the activity using HOG algorithm. The choice of algorithm is completely dependent on the 'accuracy' to 'speed of execution' ratio. CNN is a deep learning algorithm which produces higher accuracy than HOG, but when speed becomes a parameter it is wise to choose HOG.

Building the dataset

The most important part of a face recognition system is its dataset. More number of images in the dataset implies higher accuracy (Liu Xia *et al.,* 2008). Creation of a dataset can be done in many ways; 1) Face enrolment via OpenCV and webcam, 2) Downloading face images programmatically and 3) Manual collection of face images. Collecting images via webcam and OpenCV is done when an 'on-site' face recognition system is being constructed. This method requires the physical presence of the individual. Creating a dataset by downloading images programmatically is a method that is used when the person is not physically present or a public figure but has a very popular social media life. By leveraging the social media APIs one can obtain the images. The last method is the most undesirable method. It involves manually finding the images online and saving it to your disk. In our project we will be using the first method (face enrolment via webcam and OpenCV).

A. Collection of face images via webcam and OpenCV: The first step is to load OpenCV's Haar Cascade for face detection. Then, start the video stream and allow the camera to warm up. Next, iterate over the frames in the video stream and take a frame that is needed and make a copy of it, resize it and store it in the disk. This will make the face detection process faster. Now, detect the face in the stored frame and draw a rectangle around it. By pressing the key 'k' the original frame is saved which can later be used for face recognition. Upon pressing button 'q' the loop breaks. Finally, the terminal displays the number of images captured and starts the cleaning up process. Fig 1. depicts a portion of the database we have created. The important question at this point is, what is the minimum number of images for a successful face recognition?

Fig 1. Dataset created using OpenCV and webcam

B. Encoding the face: Encoding the face means extracting and quantifying the facial features and storing it in a PICKLE file. Fig 2. shows the process of quantifying faces that are stored in the dataset. Encoding a face is important, without this step there wouldn't be any parameter to compare the similarity in images. The first step to quantify a face is to detect them. So, we specify the path to the input images in our dataset. Next, iterate over the image paths and obtain the person's name from the image path. Then, convert the image from BGR to RGB (format accepted by Dlib). Then, facial embeddings for the face is computed. For each facial encoding the corresponding person's name is attached. Finally, the data collected after computation is dumped into a pickle file.

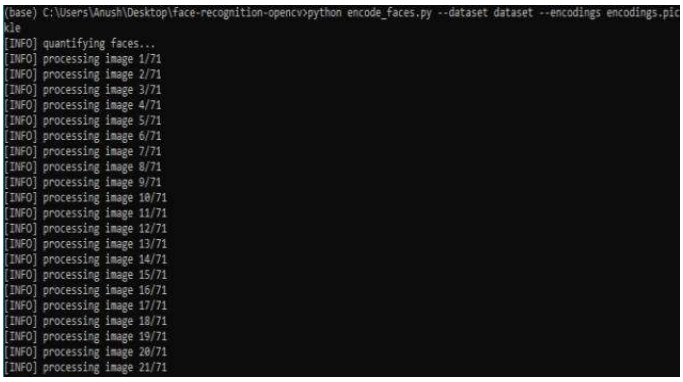

Fig 2. Facial encoding process

Face Recognition

Facial recognition is the foundation of many technological advancements and will continue to be the base for many more future technologies. Face recognition can be achieved via many methods and techniques such as Principal Component Analysis (PCA), Linear Discriminant Analysis (LDA), Convolutional neural network (CNN), Histogram of Oriented Gradients and so on (Sourabh Hanamsheth and Milind Rane, 2018). In our system we will be implementing CNN and HOG algorithms to recognise an individual's face or a group of faces.

MATERIALS AND METHODS

Now that we have extracted and quantified the facial features, we can start the process of face recognition. The algorithm that the user desires to use can be specified initially in the face recognition code. Python provides a feature in which the command line can be executed at run time. This enables us to easily add more arguments without any hassle. Next, known faces and embeddings are loaded. The input image is converted from BGR to RGB so that it is compatible with Dlib library. The (x, y) co-ordinates of faces in each input images are detected and facial embeddings are computed. Then, an attempt is made to match each face in the input image with our known encodings. The input image is looped to find a match with the images in the dataset. Count is maintained each time a face matches with an image in the dataset. Based on the number of votes, the face is recognized. Finally, the name obtained is printed above the rectangular box in the output. If no matches are found, the value 'Unknown' is printed over the face during the output. Fig 3. shows two images that are converted to sketches and are then given as inputs separately to the program, the result of this event was successful face recognition each time, even though one of the pictures of an individual is a side profile. Similarly, Fig 4. is an image of a person that is converted to a sketch. Due to bad lighting and picture quality the conversion was not very clear and accurate. This same image was passed as input to the code, a successful face recognition was achieved regardless of the sketch quality.

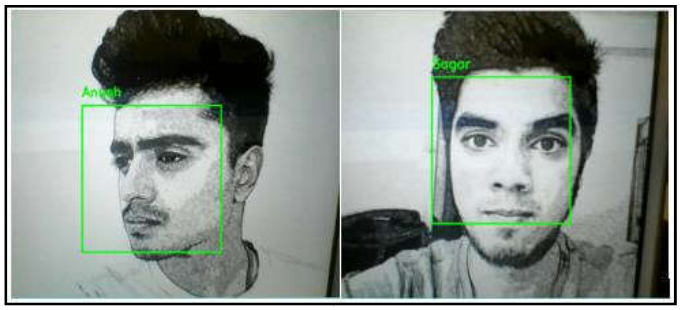

Fig 3. Successful face recognition

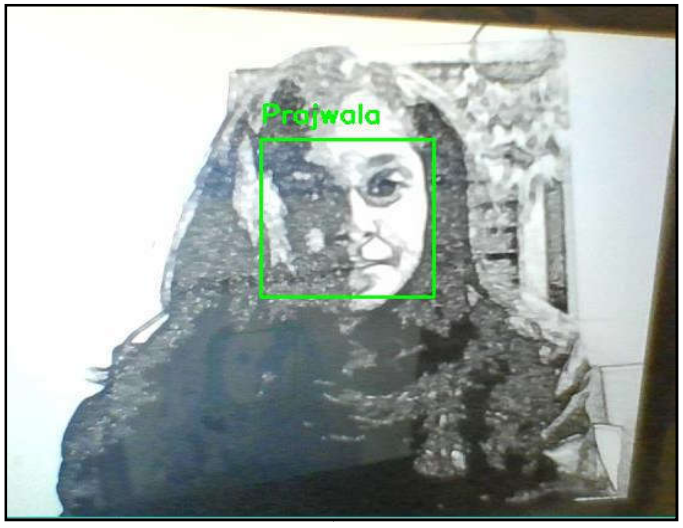

Fig. 4. Face recognized even though the image is unclear

The input image which is fed into the program needs to undergo certain changes to meet our project requirements. The input image captured using a webcam needs to be converted into sketch. This conversion can be obtained by uploading the captured image on a website, click on the link to visit the page. The generated sketch is then downloaded. This newly generated sketch needs to pass through the 'build dataset' code so that the sketch becomes a compatible input for the 'face recognition' program.

Convolution neural network (CNN)

CNN is a class of deep neural networks, most commonly used to analyzing images. Convolutional networks were inspired by

biological processes, in that the connectivity pattern between neurons resembles the organization of the animal visual cortex. A convolutional neural network consists of an input and an output layer, as well as multiple hidden layers (Musab Coúkun *et al.,* 2017). Out of the many hidden layers, convolutional layer is the first one. Every layer receives an input. A layer that first receives the input transforms it using various kinds of filters and the output of the transformed input is sent to the next layer. Convolutional layers are used to detect patterns in an image. Filters help detect patterns. Patterns such as edge detector, geometric shapes and so on. Going deeper it can detect specific objects and going further down, it can detect birds, animals, dogs and so on.

Histogram of Oriented Gradients (HOG)

The histogram of oriented gradients (HOG) is a feature descriptor used in computer vision and image processing for the purpose of object detection. Gradient values are mapped to 0-255. Pixels with large negative change will be black and pixels with large positive change will be white. To obtain gradient magnitude and gradient angle we first need to calculate both horizontal and vertical gradients (Sourabh Hanamsheth and Milind Rane, 2018).

Gradient magnitude= $\sqrt{(fx^2 + fy^2)}$

Gradient angle = $Tan^{-1}\frac{fx}{fy}$

Comparison between CNN and HOG

Table 1. Values of CNN and HOG

X Data	CNN	HOG
1	4.2	1.3
2	4.3	0.98
3	4.69	1.14
4	4.8	1.25
5	4.5	1.01
6	4.46	0.82
7	4.36	1.1
8	4.5	0.9

Convolutional neural network and Histogram of oriented gradients are undoubtedly exceptional algorithms to recognise face but a comparative study is required to determine the best out of the two algorithms. CNN is known for its accuracy while HOG is known for its speed. We are going to test this by recording the time taken to execute and display the output of the same set of images by both the algorithms. For this experiment we are considering 9 images which will be subjected to both the algorithms. The time is noted down and a bar graph is plotted to see the difference in execution speeds. Table 1. contains the values of time taken to execute and recognise faces. A same image is run on both algorithms to get one set of values.

Fig 5. is a pictorial representation of the table in the form of a bar graph, this is done to get a clearer understanding of the difference between the two algorithms.

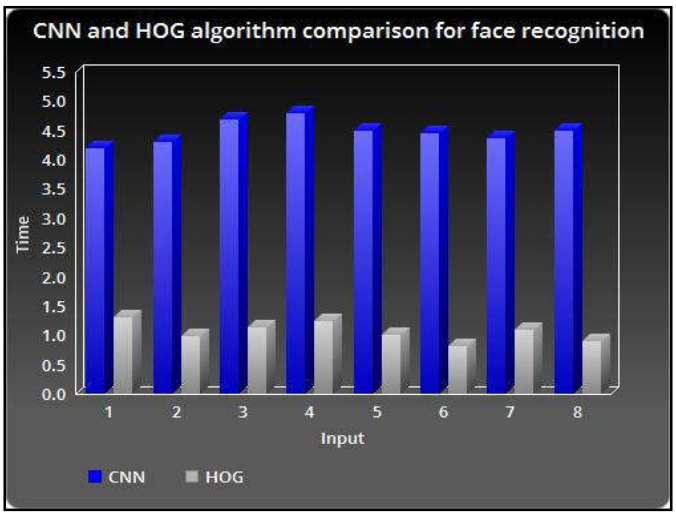

Fig. 5. Bar graph depicting the comparison of execution speed between CNN and HOG

Conclusion

This paper presents a system that can recognise face in a sketch. The paper also talks about the comparison between CNN and HOG algorithms. Upon comparison we can come to a conclusion that each of the algorithm have their own benefits. The graph obtained suggests that HOG is faster compared to CNN whereas, CNN is far more accurate.

REFERENCES

Adrian Rhesa Septian Siswanto, Anto Satriyo Nugroho, Maulahikmah Galinium, 2014. "Implementation of Face Recognition Algorithm for Biometrics Based Time Attendance System", *Information Technology*, Swiss German University EduTown BSD City, Tangerang 15339, Indonesia, 2014.

Liu Xia, Li Tingjun, Jiao Wei, Guo Qingchang, 2008. "The study of face recognition based on the large samples", Harbin Engineering University, Harbin University of Science and Technology Harbin,China, 2008.

Musab Coúkun, Ayúegül Uçar, Özal Yildirim, Yakup Demir, 2017. "Face Recognition Based on Convolutional Neural Network", Firat University Elazig, Turkey, 2017.

Priya Guptaa, Nidhi Saxena, Meetika Sharma, Jagriti Tripathia, 2018. "Deep Neural Network for Human Face Recognition", Maharaja Agrasen College, University of Delhi, Vasundhara Enclave, Delhi - 110096, India, I.J. Engineering and Manufacturing, 2018.

Sourabh Hanamsheth, Milind Rane, 2018."Face Recognition using Histogram of Oriented Gradients" *International Journal of Advance Research in Computer Science and Management Studies,* Volume 6, Issue 1, Dept. of Electronics Engineering Vishwakarma Institute of Technology Pune – India, January 2018.

****