Multidisciplinary Review Journal

Journal of Critical Reviews

ISSN- 2394-5125 Vol 7, Issue 6, 2020

Review Article

A MULTIPLE FACE RECOGNITION SYSTEM WITH DLIB'S RESNET NETWORK USING DEEP METRIC LEARNING

Dr Praveen Kumar S [1], Kesava Jayendra Varma V[2], Subramanya V[3], Venkata Sai Harish A [4]

[1] Dept. Of CSE, GIT, GITAM (Deemed to be University), spkmtech@gmail.com
[2,3,4] B. Tech students, Dept. Of CSE, GIT, GITAM (Deemed to be University), Visakhapatnam

Received: 24.02.2020 *Revised: 05.03.2020* *Accepted: 07.04.2020*

Abstract

Despite many approaches in facial recognition models, we often come across single facial recognition systems from an image.But the detection and recognition of a single face from an image is not very practical in an ever-changing world.For this, we need systems capable of detecting and recognizing multiple faces from a single image with which we can solve many real-world problems with a fewer number of images.So, we proposed an enhanced and efficient model, called HPMR (High Performance Multiple-face Recognition), to detect and recognize multiple faces from a single image.

In this paper, we used Dlib's ResNet network with 29 convolution layers to recognize faces. The network supports both Predictor 5 and Predictor 68 model to estimate facial landmarks. This model is trained on a data set of three million faces from well-known data sets like face scrub and VGG using deep metric learning. It uses HOG feature descriptor for multiple face detection. We used this model because it gives us better results compared to other existing models. Among many possible applications for multiple facial recognition, we have implemented an automated attendance system to demonstrate our approach.

Keywords: multiple face recognition, deep metric learning, histogram of gradients, ResNet network, convolution layers, HPMR

DOI: http://dx.doi.org/10.31838/jcr.07.06.147

INTRODUCTION

Face recognition is the process of identifying or verifying a person from any image, which is obtained from either image capturing device or from an individual frame in a video. A computer is not capable of such high-level perception by itself. For detecting and recognize faces we have to teach it using advanced concepts like deep metric learning and deep residual learning [2].

The first step in this process is finding faces in an image. To detect a face in an image it should be first converted to greyscale since we won't need colour. Now we study every pixel in the image and the pixels surrounding it. Our objective is to work out how dark the present pixel is compared to the pixels immediately surrounding it. Then we wish to draw an arrow showing the direction in which the image is getting darker. By doing this the result is that we convert the image into a simple representation of arrows called gradients. We then generate a general pattern by collecting gradients from lots of faces. Then compare it to the image we want to detect. This method of detection is called histogram of gradients or HOG [3]. Now the detection works properly when the face is straight forward. But if the face turns sideways then the computer sees them as different faces. To correct this, we use shape predictor 5 and 68 [4]. What it does is warp the picture to make the face straight forward. To do this it uses face landmark estimation. Landmarks are points on the face that can be used to uniquely identify a face like eyes, nose, mouth, etc. Predictor 5 uses 5 points whereas predictor 68 uses 68 points. Now the last step is encoding faces [1]. The encodings are 128 measurements of face arranged in a 2D array. The need for generating measurements instead of comparing images directly is to reduce the time to recognize significantly. These measurements are generated by training a deep convolution network [2]. This lets the computer figure out how to take measurements reliably for any face once trained. And the last step is to use a classifier like SVM classifier to compare the faces to and classify them based on a predetermined threshold.

Now, in the system we proposed for detecting multiple faces, we first detect all faces using hog in an image. Then we use any image handling tools to crop out individual faces selected using a rectangular box and generate encodes for every face separately. Face Recognition has a multitude of application in many fields like Law enforcement, forensic investigation, and attendance systems, validate identities at ATM, tracking missing people, etc.

LITERATURE REVIEW

In "Real-Time Smart Attendance System using Face Recognition Techniques", S. Sawhney used CCTV surveillance footage for user's data and the CNN model for image scaling and identifying. The problem with this approach is there is a lot of unwanted data [8] from the video. Processing a video stream requires large scale servers with more GPU power and the CNN model [13] will require more processing power eventually increasing the investment and maintenance costs. Using pictures instead of the video stream will reduce these costs. CCTV cameras cannot give high-resolution pictures sufficient for processing and also increases investment and maintenance costs. The method proposed in this paper works on any devices which can capture images and connect to the internet and reduces the investment cost and can work on small scale servers reducing maintenance costs.

In"Automatic attendance management system using face detection", Varadharajan says implementing a biometrics-based attendance management system instead of the traditional manual methods. The method proposed in his paper is a similar approach considering the implementation, practicality and feasibility. It doesn't require installation of any hardware[12]. Just a cloud computer and any existing cameras are the only requirements. The methods and algorithm used gives 90% accuracy in the detection [9] and identification of faces.

In "Attendance Management Using Facial Recognition", Rajath S Bharadwaj used a computer [10] where every student needs to go and take a snap of their own for attendance. Moreover, usage of excel sheets over a DBMS is not feasible due to data redundancy [11] issues and also retrieval is the difficulty. The model proposed in this paper addresses this problem. Moreover, asking students to go to a spot for attendance is not possible and the model cannot be used for implementing. The model proposed in this paper uses the faculty's android device for taking pictures and session data. Rest is handled by the server to post the attendance.

PROPOSED SYSTEM

A. Architecture

The system is designed using REST protocol. When the image capturing device captures an image, a HTTP post request is made to the API.Then the API will take the new images and storage manager stores with unique filenames. When process request is received, the face identification process is triggered. It first acquires all the required files from storage manager and face detection process is started. Then faces are cropped and recognition process starts. Then a list of recognised and unrecognised people is stored in the database. The database consists of three collections, one stores information about existing profiles, one stores if that person presents at that time and other stores unrecognised people data. The dashboard on the other hand enables the end-user to create profiles for new users, check if a person is present for a certain date, check unrecognised people and update or delete their temporary profiles. This is in turn handled by the API for CRUD (Create Read Update Delete) operations.

B.System Design

The block diagram in figure 1 describes the proposed design for multiple face recognition system. The system requires a device capable of capturing and transmitting images over HTTP. The working is explained in brief below:

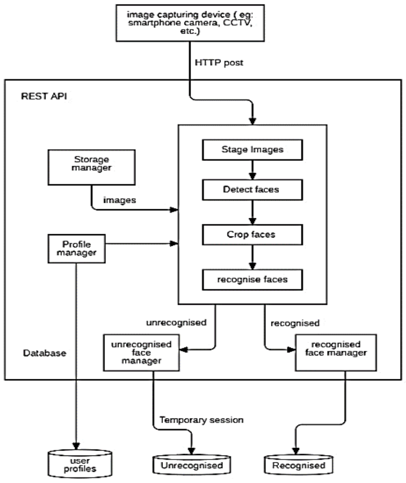

Figure 1 System Designof HPMR

- Image capturing device: The main role of this is to take pictures and send to the webserver for identifying.
- Web Server (REST API): This module works to manage storage, process images and CRUD operations in the database. It accomplishes the main driver sequence of the entire system integrating each component to complete the system.
- Dashboard: This part is used to create a database for the faces by uploading their images for facial recognition. It is also used to view who is there for a given time and date.
- Database: This is the database which stores entire data. It consists of three collections which stores profiles, unrecognized and recognised data.

SYSTEM IMPLEMENTATION

So far, we have discussed multiple face recognition. Here we will see an example of how it can be used practically. We have used our approach to implement an automatic attendance system to test our approach.

A. System Prerequisites

The first step in implementing the system is to create a database of enrolled students. In actual implementation, this step must be a part of the admission process where we collect the necessary information of the students. This is accomplished using dashboard as mentioned in the proposed system. The set of images which is known image data set is stored in a folder named known. The facial recognition algorithmthen uses the image data set to create a dictionary of face encoding containing keys that are the unique id of the image stored and value that is 2D array containing 128 measurements of face.This is stored physically in JSON format for faster access to image data and to recognize it.In our implementation of the proposed model, we have created a script known_face_encodings for this purpose. This function works as follows:

1. Iterate through every image inknown folder
2. Generate face encoding for each image
3. Store in the temporary dictionary variable
4. Dump to JSON file using pickle library

After this, the system is ready to accept images to recognize and post attendance if the image is present in the database. This is accomplished using an android app which does the following:

1. Select date and session
2. Capture image of a group of students
3. Compress the image within the app before the transfer
4. Send the image to the server
5. Confirm (which starts the actual image processing and recognition consequently after which attendance will be posted)

B. Face Detection and Cropping

The captured image from an android app is sent as a list to theserver.The image is nothing but a matrix of numbers which correspond to the pixel values. The software doesn't know where in this collection of numbers the faces are present, which are the input for our algorithm. Thus, face detection performs this task.We use the method 'face_locations(image)' from face_recognitionlibrarywhere the image is the captured image. This function detects the face based on theHistogram of Oriented Gradients or justHOG[3] for short. The sequence of steps in this algorithm is as follows.

1. Read the image captured in the previous step
2. The faces are detected from the above image as explained
3. Crop the area of the image where the faces are marked and save into a folder named unknown as individual image files in JPEG format.

The algorithm detects all the faces clearly visible in the captured image of the classroom. Each student should be in an upright position and facing the camera to avoid exclusion of their presence by the system.

C. Face Recognition

The following happens after faces are cropped and stored.

1. Each image in unknown folder is taken and its face encodings are generated using face encodings (unknown_image) method from face recognition library where unknown_image is an image in unknown folder.
2. The encoding is directly compared against all the encodings of known images stored in JSON file
3. When an image is recognized then the names are appended to the identified image list.

4. When an image is not recognized then the names are appended to the unidentified image list.
5. Both identified and unidentified list are returned as a dictionary to the main server-side python script which is built using flask library.

D. Store Recognized Entries

Whenever the algorithm finds a match, the name of the images identified are first stored in a dictionary with two keys where one represents identified and other represents unidentified. Based on the list we then update the corresponding field of the person in the firebase database with a '1' on that particular date using Pyrebase library. Else by default, it is marked as '0' which means that the person is absent. Firebaseprovides a very efficient way of storing the data.

E. View the Attendance in dashboard

The attendance is posted for a given date and session in Firebase real-time database. Unidentified people images and date and session details are stored by generating temporary ids for each. The angular application fetches these details and displays it in a clean UI. See the visual output of angular web application in the result section.

RESULTS AND DISCUSSION

We have tested this model and observed its behaviour in different sample size. Table 1, shows the results of the test cases.

Table 1Sample results

No. of faces in image	No. of faces detected	No. of faces recognized
10	10	10
20	20	20
30	30	30
40	40	40
50	50	50
60	59	57
70	66	62
80	75	69
90	81	73
100	85	75

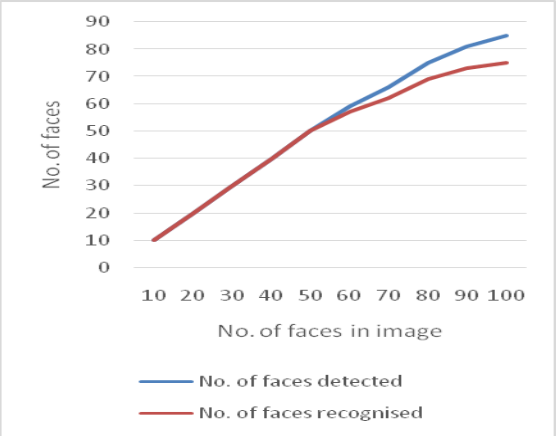

Figure 2 Graph representation

From the graph we can observe that detection of faces in an image decreases with increase in number of faces in same image. This is an expected output as the faces become more distorted. From the graph we can calculate the accuracy which is 93.81% in detecting and 94.18% in face recognition and with total accuracy as 88.36%

CONCLUSION

The model proposed in this paper is very effective as it gives quick and accurate results. Since our model stores only the encodings of the images rather than the images themselves, thereby reducing the space and time needed to retrieve images from the database and process them each time an image needs to be recognized. The system is also capable of batch processing multiple images at a time. Thus, the aim of the paper to demonstrate multiple face detection and recognition is successfully achieved by using Dlib's ResNet Network.

FUTURE SCOPE

We implemented a semi-automated attendance system using the multiple face recognition model proposed in this paper. This model can be further enhanced by making it fully automatic with the available digital identification and tracking each and every movement of people and thus improving the safety and security.

REFERENCES

1. F. Schroff, D. Kalenichenko and J. Philbin, "FaceNet: A unified embedding for face recognition and clustering," 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Boston, MA, 2015, pp. 815-823.
2. K. He, X. Zhang, S. Ren and J. Sun, "Deep Residual Learning for Image Recognition," 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, 2016, pp. 770-778.
3. N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05), San Diego, CA, USA, 2005, pp. 886-893 vol. 1.
4. V. Kazemi and J. Sullivan, "One millisecond face alignment with an ensemble of regression trees," 2014 IEEE Conference on Computer Vision and Pattern Recognition, Columbus, OH, 2014, pp. 1867-1874.
5. S. Sawhney, K. Kacker, S. Jain, S. N. Singh and R. Garg, "Real-Time Smart Attendance System using Face Recognition Techniques," 2019 9th International Conference on Cloud Computing, Data Science & Engineering (Confluence), Noida, India, 2019, pp. 522-525.
6. Varadharajan, R. Dharani, S. Jeevitha, B. Kavinmathi and S. Hemalatha, "Automatic attendance management system using face detection," 2016 Online International Conference on Green Engineering and Technologies (IC-GET), Coimbatore, 2016, pp. 1-3.
7. Rajath S Bharadwaj, Tejus S Rao, Vinay T R, "Attendance Management Using Facial Recognition", International Journal of Innovative Technology and Exploring Engineering (IJITEE), Volume-8 Issue-6, April 2019
8. Praveen Kumar S, Srinivas Y, Bhargav K, " An n-gram analysis approach for sharing authentication of data using model based techniques", Test Engineering and Management, 2020.
9. Praveen Kumar S, Sahithi Choudary A, " An Innovative ModelBased Approach for CreditCard Fraud Detection Using Predictive Analysis and Logical Regression." , International Journal of Innovative Technology and Exploring Engineering (IJITEE), Scopus, 2019, 8 , 1683-1688
10. 10.Praveen Kumar S, Srinivas Y, Vamsi Krishna M, "A Data Leakage Identification System Based on Truncated Skew Symmetric Gaussian Mixture Model.", International Journal of Recent Technology and Engineering (IJRTE), Scopus, 2018, 7, 111-113

11. Praveen Kumar S, Srinivas Y, Vamsi Krishna M, "Latent Semantic Indexing based Approach for Identification of Frequent Itemset.",Jour of Adv Research in Dynamical & Control Systems, Scopus, 2018, Vol. 10, 686-690.
12. Praveen Kumar S, Srinivas Y, Vamsi Krishna M, "A mechanism for identifying the guilt agent in a network using vector quantization and skew Gaussian distribution."International Journal of Engineering & Technology, Scopus, 2018, 7, 149-151
13. 13.Praveen Kumar S, Srinivas Y, Ranjansenapat M, Kumar A, "An Enhanced Model for Preventing Guilt Agents and Providing Data Security in Distributed Environment."IEEE Conference International conference on Signal Processing, Communication, Power and Embedded System (SCOPES), others, 2016, 1, 337-339