International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 10 Issue 04, April-2021

Object Detection based Attendance System

Vedang Koli
Information Technology
Vidyavardhini's College of
Engineering and Technology
Vasai, India

Tejas Vedak
Information Technology
Vidyavardhini's College of
Engineering and Technology
Vasai, India

Devanshu Sharma
Information Technology
Vidyavardhini's College of
Engineering & Technology
Vasai, India

***Abstract*—The advancement of computation power and leading development in machine learning will enhance and help in optimizing the efficiency of object detection and tracking. Computer vision is in trending development phase and can totally change the scenario of visual artificial advancement. We will always be looking for ways that can get be brought together to create better models for this training spectrum. There are various attendance system based on QR scheme, registration technique, etc involving heavy intervention from humans. Attendance system are needed in every field of work and removing the intervention from human end can reduce lot of time and efforts. Here we are trying to automate the attendance system using facial recognition technique.**

Keywords—Face detection, CNN, AWS Rekognition, S3, Open CV, libraries, image, neural networks.

I. INTRODUCTION

Deep learning methods have become more common in recent years in the field of research into face detetction. This pattern is also greater in other areas of machine learning, such as computer vision and the processing of natural languages, those with wider populations and more competition. Ultimately, Methods of deep learning would likely play an important function in face detection. In the sense of deep learning, we'll have some basics of face detection. [1] With recent developments in machine learning and OpenCV, Face Detection tasks such as the identification of several faces can now be solved. AWS have several libraries for storing and detection of face images. [2] Extracting correct facial characteristics is typically the first step towards tackling the problem. In order to distinguish between the face images, we are keeping faceID as unique attribute to identify into database. [4] In this paper, we have proposed object detection based attendance system.

The main motivation behind this project was slow and manual procedure to mark attendance. It also have several drawbacks like fake or proxy signature on paper and and in order to prevent it, we will be implementing several technologies. The advancement of computation power and leading development in machine learning will enhance and help in optimizing the efficiency of object detection and tracking. Computer vision is in trending development phase and can totally change the scenario of visual artificial advancement. We will always be looking for ways that can get be brought together to create better models for this training spectrum.

II. LITERATURE REVIEW

Face recognition is one of the few biometric methods that possess the merits of both high accuracy and low intrusiveness. It has the accuracy of a physiological approach without being intrusive. Over past 30 years, many researchers have proposed different face recognition techniques, motivated by the increased number of real world applications requiring the recognition of human faces. There are several problems that make automatic face recognition a very difficult task. However, the face image of a person inputs to the database that is usually acquired under different conditions. The important of automatic face recognition is much be cope with numerous variations of images of the same face due to changes in the following parameters such as pose, illumination, expression, motion, facial hair, glasses, background of the image.

Face recognition technology is well advance that can applied for many commercial applications such as personal identification, security system, image- film processing, psychology, computer interaction, entertainment system, smart card, law enforcement, surveillance and so on. Face recognition can be done in both a still image and video sequence which has its origin in still-image face recognition. Different approaches of face recognition for still images can be categorized into three main groups such as holistic, feature based or hybrid approach.

Holistic approach:- In holistic approach or global feature, the whole face region is taken into account as input data into face detection system. Examples of holistic methods are eigen faces (most widely used method for face recognition), probabilistic eigen faces, fisher faces, support vector machines, nearest feature lines (NFL) and independent-component analysis approaches. They are all based on principal component-analysis (PCA) techniques that can be used to simplify a dataset into lower dimension while retaining the characteristics of dataset.

Feature-based approach:- In feature-based approaches or local feature that is the features on face such as nose, and then eyes are segmented and then used as input data for structural classifier. Pure geometry, dynamic link architecture, and hidden Markov model methods belong to this category. One of the most successful of these systems is the Elastic Bunch Graph Matching (EBGM) system, which is based on DLA. Wavelets, especially Gabor wavelets, play a building block role for facial representation in these graph matching methods. A typical local feature representation consists of wavelet coefficients for different scales and rotations based on fixed wavelet bases. These locally estimated wavelet coefficients are robust to illumination change, translation, distortion, rotation, and scaling. The grid is appropriately positioned over the image and is stored with each grid point's locally determined jet, and serves to represent the pattern classes. Recognition of a new image takes place by transforming the image into the grid of jets, and matching all stored model graphs to the image. Conformation of the DLA is done by establishing and dynamically modifying links between vertices in the model domain.

Hybrid approach:- The idea of this method comes from how human vision system perceives both holistic and local feature. The key factors that influence the performance of hybrid approach include how to determine which features should be combined and how to combine, so as to preserve their advantages and avert their disadvantages at the same time. These problems have close relationship with the multiple classifier system (MCS) and ensemble learning in the field of machine learning. Unfortunately, even in these fields, these problems remain unsolved. In spite of this, numerous efforts made in these fields indeed provide us some insights into solving these problems, and these lessons can be used as guidelines in designing a hybrid face recognition system. Hybrid approach that use both holistic and local information for recognition may be an effective way to reduce the complexity of classifiers and improve their generalization capability.

III. REPORT ON THE PRESENT INVESTIGATION

A. *Overview on Deep Learning Techniques*

There have been a number of significant early works related to the latest deep learning technologies on neural networks. Before we explore the aspects of deep learning, let us explain and describe some basic concepts of neural networks.

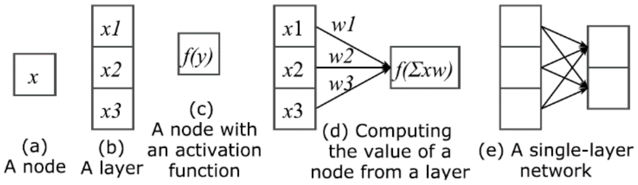


Fig. 1. Anatomy of single – layered neural networks

As in Figure 1(a), a node is analogous to a biological neuron and represents a scalar value. A layer consists of a number of nodes, as shown in Figure 1(b) and is a vector. Note the nodes are not interconnected within a network. As in figure 1(c), an activation function f() is normally added to each node. A node can be considered activated / deactivated by an activation function's output value. A node's value is typically determined as a weighted input sum followed by an activation function, i.e., f (w * x) where w is the weights and x is the inputs as shown in Figure 1 (d). Figure 1(e) shows a basic artificial neural network, which is an extension to Figure 1(d). There are intermediate layers in a multi-layered network as well as the input and output layers which are often called hidden layers. A network depth reflects the total number of layers in a network. Likewise a network's width(s) is the number of nodes in layer(s).[1] Deep neural networks (DNNs) model multilayered neural networks.

B. *Face Recognition Using Python*

- Recognize and manipulate faces from Python or from the command line with the world's simplest face recognition library.
- 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.
- Implementation:
  1) Load the images:
     a) Load the original image which needs to be registered for recognition, aka train image.
     b) Load the target image which needs to be recognized against the original image, aka test image.

  2) Face Location and Face Encodings:

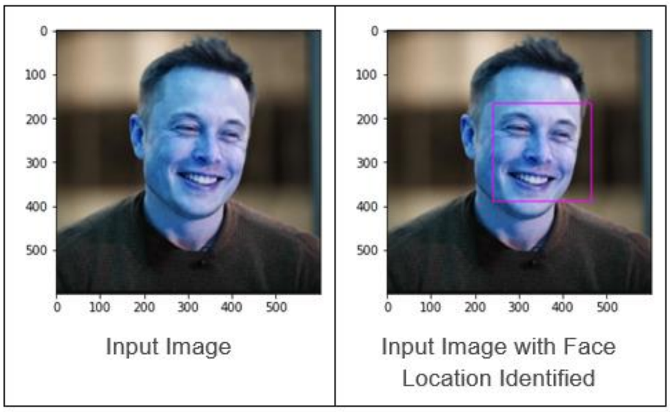


Fig. 2. Raw Image

a) The very first step is to identify the important features of the image, that is eyes, eyebrows, nose, lips, etc. This comes under face location.
b) Once the face location is identified, a rectangle / square box is created around it which will be used as a core component for recognition.
c) The algorithm creates encoding of the identified locations, which is done extracting important features only of the face.
d) This encoding is converted to vector representation which is then fed to the algorithm as input.
e) The Location and encoding is performed for both the images.

3) Face Comparison

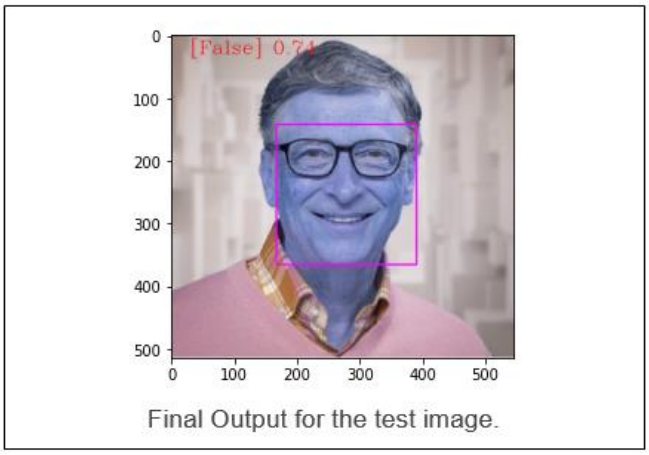


Fig. 3. Processed Image

C. Attendance Logic

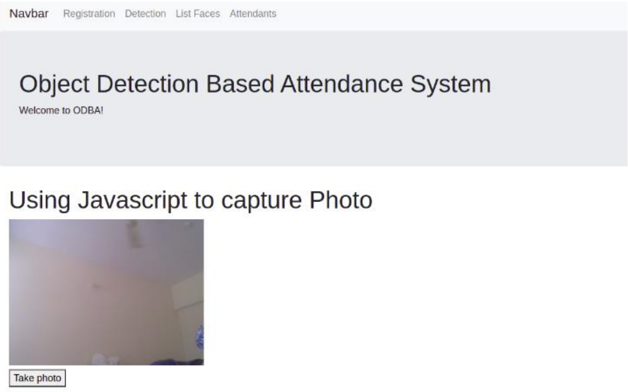
- A folder of training images is input to the face encoding algorithm which evaluates the encoded vectors of the training images.
- A video stream feed is taken as the input from a webcam where it detects the face and draws a rectangle around the face.
- The rectangle part is encoded and is sent for the comparison against all the train images.
- If similarity comparison surpasses a certain threshold then attendance is marked and the person's name is reflected.

D. Face API Using Node JS

- The answer to the first problem is face detection. Simply put, we will first locate all the faces in the input image. Face-api.js implements multiple face detectors for different use cases.
- The most accurate face detector is a SSD (Single Shot Multibox Detector), which is basically a CNN based on MobileNet V1 , with some additional box prediction layers stacked on top of the network.
- Furthermore, face-api.js implements an optimized Tiny Face Detector , basically an even tinier version of Tiny Yolo v2 utilizing depth wise seperable convolutions instead of regular convolutions, which is a much faster, but slightly less accurate face detector compared to SSD MobileNet V1.
- Lastly, there is also a MTCNN (Multi-task Cascaded Convolutional Neural Network) implementation, which is mostly around nowadays for experimental purposes however.
- The networks return the bounding boxes of each face, with their scores. E.g. the probability of each bounding box showing a face. The scores are used to filter the bounding boxes, as it might be that an image does not contain any face at all.
- Note, that face detection should also be performed even if there is only one person in order to retrieve the bounding box.
- Now we can feed the extracted and aligned face images into the face recognition network, which is based on a ResNet-34 like architecture and basically corresponds to the architecture implemented in dlib
- The network has been trained to learn to map the characteristics of a human face to a face descriptor (a feature vector with 128 values), which is also oftentimes referred to as face embeddings.
- Now to come back to our original problem of comparing two faces: We will use the face descriptor of each extracted face image and compare them with the face descriptors of the reference data.
- More precisely, we can compute the euclidean distance between two face descriptors and judge whether two faces are similar based on a threshold value (for 150 x 150 sized face images 0.6 is a good threshold value). Using euclidean distance works surprisingly well, but of course you can use any kind of classifier of your choice.

IV. RESULTS AND IMPLEMENTATION

A. Registration

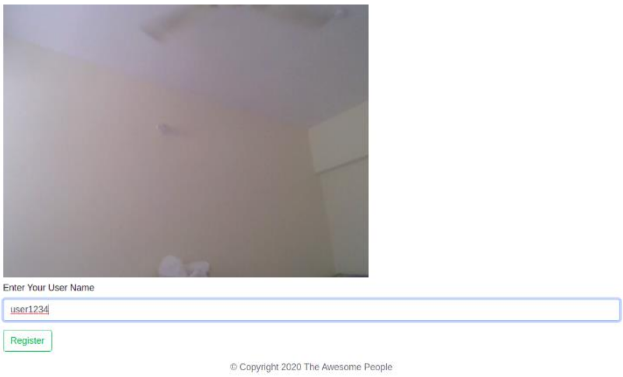


Object Detection Based Attendance System

Welcome to ODBA!

Using Javascript to capture Photo

Fig. 4. Home Page


Enter Your User Name

user1234

© Copyright 2020 The Awesome People

Fig. 5. Registration Page

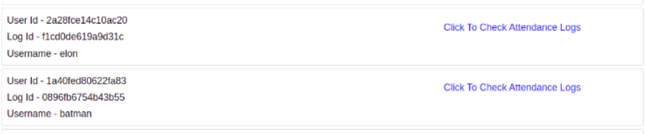
- A live feed is enabled from a webcam where a user can click his/her photo which is then converted to ".png" format automatically.
- The user then writes his/her name and submit it to the API.

- Once submitted the registration api endpoint uploads the image to S3 Bucket and sends the S3 uploaded response to Rekognition API.
- The Rekognition API with the help of S3 Bucket name and key it adds the face to the collection and now this will be known as registered face.
- The face id and image id returned from Rekognition is then added in the registration database (collection) and a log is created under the attendance Log database (collection).

B. Detection

Fig. 6. Detection Page

- Similar to the registration process the webcam is enabled and the user can click his/her still image and can be submitted to the API.
- Once submitted the detection API will upload the image to S3 Bucket and after the image details are sent to the AWS Rekognition Service.
- The searchFacesByImage function of the Rekognition compares with all the indexed faces (registered faces) and returns an array of matching images with the highest threshold.
- If an empty array is returned that means no indexed face is matching and probably the user is not registered.



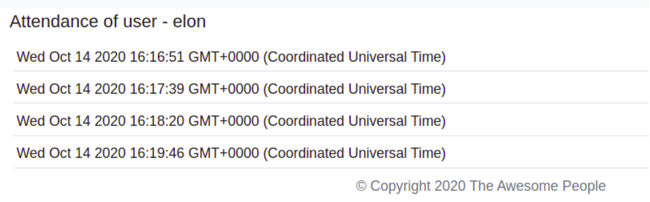
User Id - 2a28fce14c10ac20
Log Id - f1cd0de619a9d31c
Username - elon

Click To Check Attendance Logs

User Id - 1a40fed80622fa83
Log Id - 0896fb6754b43b55
Username - batman

Click To Check Attendance Logs

Fig. 7. Attendance log of users



Attendance of user - elon

Wed Oct 14 2020 16:16:51 GMT+0000 (Coordinated Universal Time)

Wed Oct 14 2020 16:17:39 GMT+0000 (Coordinated Universal Time)

Wed Oct 14 2020 16:18:20 GMT+0000 (Coordinated Universal Time)

Wed Oct 14 2020 16:19:46 GMT+0000 (Coordinated Universal Time)

© Copyright 2020 The Awesome People

Fig. 8. Attendance record of users

V. FUTURE WORK

We have done the literature survey, now we got a sense of the direction we needed to pick with regards to developing the project. We also began our development work and completed some of the phases of our development sequence. We completed the phases of taking input as image and details from user for registration. We also completed the detection of already registered user for marking attendance.

We also completed the integration of AWS rekognition library for identification with AWS S3 for saving FaceID. Also we are saving ImageID in MongoDB in a systematic manner. After completing some important phases, we will try to upgrade the flow of the implementation where the user's face can be detected automatically and the user should not need to click a photo manually. And last but not least, we would also be starting work with respect to developing a UI/UX for this project.

A. Proposed Technologies

We have worked on several technologies before approaching the problem and tried to find out easy and feasible solution for it. We are using AWS rekognition library with integration with AWS S3 which makes it easy to fetch data and functions efficiently. Also we have kept our frontend as simple as possible in order to make it user friendly.

We are also keeping track of timestamp of marking attendance for particular person to cross check or to maintain records. Apart from this, for our UI/UX, we intend to use the latest web development technologies like HTML, CSS, JS. We have considered the EJS library for creating a Single Page Application.

VI. CONCLUSIONS

This will be the final chapter of the report summarizing and concluding our overall progress with respect to the project. So, in this semester, basically we picked up from where we left off in the previous semester. In the previous semester, we chose this topic for our BE project because we wanted to solve real world problem and implement it. This project is both challenging and also proving to be a catalyst of knowledge for us as we have been coming across many new concepts, technologies, techniques and findings right from day one.

This project has also helped us learn and adapt to ways of working in a team, both onsite and remotely and using the various collaborative tools for the same. So last semester, after the topic was approved by the panel, we started doing our research and going through the scholarly papers. We did a decent literature review beginning from Deep Learning for Face Detection all the way up to domain specific papers for Face Detection and use of CNN. These papers solved a very big problem of ours and that is it gave us the right direction to follow to achieve what we intend to.

So keeping on the same, we were able to get a clear picture of the things we needed to do and in what sequence. Once, we had a clear understanding of all the things we needed to do, we started executing. Since none of the team members had an experience of working on a Deep Learning based project before, it was a first for all of us. That is also the same reason why development is an extra bit of challenge and is taking us more time to get things done with trials and errors. We knew all the phases that are required and their sequence of those phases that we would need to follow to achieve our intended project. We are on the right track to implement it on smaller scale and on getting desired results, we can use in our college campus as well.

REFERENCES

[1] Object Detection and Tracking using Tensorflow by R. Sujeetha, Vaibhav Mishra

[2] Object Detection with Deep Learning; A Review by Zhong – Qiu Zhao, Member, IEEE, Peng Zheng, Shou – tao Xu, and Xindong Wu, Fellow, IEEE

[3] Deep Network Pruning For Object Detection by Sanjukta Ghosh, Shashi K K Srinivasa, Peter Amon, Andreas Hutter, Andre Kaup

[4] Joseph Redmon, AliFarhadi, YOLO9000:Better, Stronger, Faster, 2017 IEEE, pp. 6517-6526 Chengtao Cai, Boyu Wang, Xin Liang, A New Family Monitoring Alarm System Based on Improved YOLO Network, pp.4269-4274 Alexaender M., Micheal M. and Ron Kimel, 3-DFace Recognition, unpublished, First version: May 18, 2004; Second version: December 10,2004.

[5] R. Bruneelli and T. Poggio, "Face Recognition: Features versus Templates", IEEE, 1993,(15)10:1042-1052[6] Albiol, A Oliver, J., Messi, J.M. bronsky using depth cameras. Computer Vision, IET. Vol 6(5),378-387.

[6] ICCV 2009 - International Conference on Computer Vision, Sep 2009, Kyoto, Japan. IEEE, pp.498-505, 2009, 〈 10.1109/ICCV.2009.5459197

[7] H. Rowley, S. Baluja, and T. Kanade. Neural network-based face detection. In IEEE PAMI, volume 20,1998.

[8] Better face Recongnition software computers at recognizing faces in recent tests. By Mark Willaims Pontin May 30,2007MIT

[9] Joseph Redmon, Santosh Divvala, Ross Girshick, Ali Farhadi, You Only Look Once: 2016 IEEE Conference, pp.780-788