

Video based Face Recognition and Tagging using Deep Learning

Aswathy V

M. tech student

Department of Computer Science And Engineering,LBS
Institute of Technology For Women
Thiruvananthapuram,India

Lijin Das S

Assistant Professor

Department of computer science and engineering,LBS
Institute of technology For Women
Thiruvananthapuram,India

Abstract— Over the years ,several studies has done in the area of face recognition for automatic face detection and tagging from video. The development of automatic face detection system is a challenging task. It can be used in many applications like security, gaming, law and enforcement etc.A face detection method is a technology having the ability to recognize and verify a person from an image or a video.Various methods are available for face recognition but commonly they function by comparing the facial features of detected image with faces which is already stored in database. It is an Biometric application based on Artificial Intelligence which can uniquely pick out a person by examine patterns based on the facial features of a human.Recently, it is shown an increase in studies of face recognition system which is able to detect faces from video automatically in real time.The proposed system is to detect and tag faces from a video using a large collection of stored face images.This paper presents an approach of real time attendance system which utilizes a Convolutional Neural Network (CNN) to detect and tag those detected faces.The accuracy obtained by using CNN is 96%, which is more efficient when compared to accuracy obtained by the traditional neural network systems.

Keywords— Face identification, Deep Learning, Convolution Neural Networks.Face Recognition, Face Tagging.

I. INTRODUCTION

Face detection is an useful technology used in several applications that identifies faces of person as digital images. In recent years, face recognition is very important due to increase in security features. It is used to identify a person from video sequence automatically using a large collection of stored images. It identifies image by getting features like size, shape of eyes, nose, jaws, cheek of detected person.Face recognition can be used in various applications like security applications,games,military,bank etc.A collection of images with pose variations are trained and stored in the database. When an image is detected, compare it to the trained model. If the face is detected then an system is generated to authenticate the person automatically in real time. Finally, in experiments only image of person from trained set is presented.Over the years ,several studies has done in the area of face recognition for automatic face detection from video. The development of automatic face detection system is a challenging task. It can be used in many applications like security, gaming, law and enforcement etc. Recently, it is

shown an increase in studies of face recognition system which is able to detect faces from video automatically in real time.This paper presents a Convolutional Neural Network (CNN) based technique for detecting and tagging faces.

II. PROPOSED METHODOLOGY

This paper presents video based face recognition and tagging system using CNN.The main goal of the proposed system is to take attendance automatically using camera and also detect and tag the recognised faces which can be used for identifying persons(criminals,famous personalities) from public places.Firstly,train the faces in database.

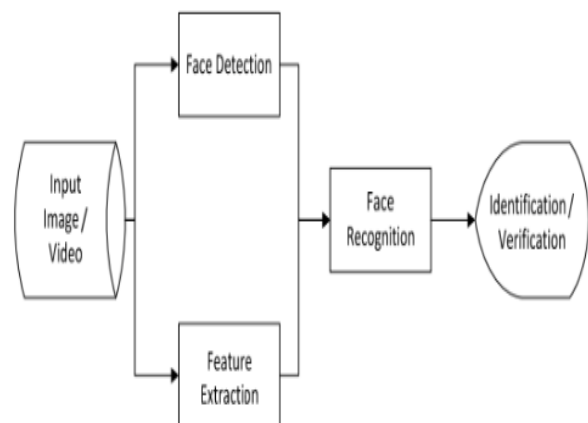


Fig 1. Block Diagram of the Proposed System

For that 200 images for each face with different poses are used.then,input a video to CNN architecture for recognising faces.Third step is to tag those recognised faces. Figure1 shows the block diagram of the proposed system. As shown in the figure, the trained system will able to detect ,recognise and tag the faces from video.

III. FACE IDENTIFICATION AND TAGGING

This module gives the outline of face identification and tagging in real time.For that an attendance system is generated.firstly register the details of students and capture their images and stored in database.Next procedure is to mark attendance by capturing image of a particular person

infront of camera and compare it with images those stored in database. This system consist of two stages.

- i) Detecting and recognising faces using face detector
- ii) Tag those detected faces

A) Face Identification

We start the process by training images from video, i.e., converting the video to frames. Capture upto 200 images per face at different poses is captured and stored on database. When we input an image, compare that image with already trained images in database. If any matches found, then tag those recognised faces. Otherwise update the database with new image.

B) Face Tagging.

Once the faces are identified, next step is to label those detected faces. For that we created a system to label the faces automatically once we have trained model of each face.

IV. ATTENDENCE SYSTEM

Daily attendance marking in classroom is very difficult. To improve the time consumption, an automated attendance system is generated. We have different steps to do.

1. Register:- Here the details of student to be entered such as name, mail, mobile number, gender. After that web camera will start capturing images upto 200 images per face and stored in database.

2. Training dataset:- In this stage, the images stored in the database are trained.

3. Attendance:- In this stage, again web camera will be on and starts to predict the face by comparing the images stored in the database. If matching found, tag that face and mark attendance of the corresponding person.

V. DATASETS

Dataset for training is obtained from face Image Database. In case of real time video, upto 200 images per face is captured live. Images are saved in png format.

VI. DEEP LEARNING

Deep learning is a Machine learning method based on artificial neural networks. Learning can be supervised, semi supervised or unsupervised. Supervised learning is a machine learning task that map input to output based on input-output pair. Semi supervised learning comes between unsupervised learning and supervised learning. Deep Neural Network (DNN) is an artificial neural network (ANN) with several layers between input and output layer. Deep learning is an feature of artificial intelligence (AI) that is anxious with reproducing the learning method that human beings use to attain undoubted types of awareness. At its simplest, deep learning can be thought of as a way to automate predictive analytics. While traditional machine learning algorithms are linear, deep learning algorithms are stacked in a hierarchy of increasing complexity and abstraction. To understand deep learning, imagine a toddler whose first word is *dog*. The toddler As the toddler continues to point to objects, he

becomes more aware of the features that all dogs possess. What the toddler does, without knowing it, is clarify a complex abstraction (the concept of dog) by building a hierarchy in which each level of abstraction is created with knowledge that was gained from the preceding layer of the hierarchy.

VII. CONVOLUTIONAL NEURAL NETWORKS (CNN)

CNNs are regularized versions of multilayer perceptrons. Multilayer perceptrons usually refer to fully connected networks, that is, each neuron in one layer is connected to all neurons in the next layer. The "fully-connectedness" of these networks make them prone to overfitting data. Typical ways of regularization includes adding some form of magnitude measurement of weights to the loss function. However, CNNs take a different approach towards regularization: they take advantage of the hierarchical pattern in data and assemble more complex patterns using smaller and simpler patterns. Therefore, on the scale of connectedness and complexity, CNNs are on the lower extreme. Convolutional networks were inspired by biological processes in that the connectivity pattern between neurons resembles the organization of the animal visual cortex. Individual cortical neurons respond to stimuli only in a restricted region of the visual field known as the receptive field. The receptive fields of different neurons partially overlap such that they cover the entire visual field. A convolutional neural network consists of an input and an output layer, as well as multiple hidden layers. The hidden layers of a CNN typically consist of convolutional layers, RELU layer i.e. activation function, pooling layers, fully connected layers and normalization layers. In convolutional layer, Each convolutional neuron processes data only for its receptive field. Although fully connected feedforward neural networks can be used to learn features as well as classify data. Convolutional networks may include local or global pooling layers. Pooling layers reduce the dimensions of the data by combining the outputs of neuron clusters at one layer into a single neuron in the next layer. Local pooling combines small clusters, typically 2 x 2. Global pooling acts on all the neurons of the convolutional layer. In addition, pooling may compute a max or an average. Max pooling uses the maximum value from each of a cluster of neurons at the prior layer. Average pooling uses the average value from each of a cluster of neurons at the prior layer.

Fully connected layers connect every neuron in one layer to every neuron in another layer. It is in principle the same as the traditional multi-layer perceptron neural network (MLP). The flattened matrix goes through a fully connected layer to classify the images.

VIII. RESULTS

The proposed system first detect the face from given input image. Once face is detected, then extract the features of face and compare it with trained images in the dataset. If any matching found, recognise that face and tag it using corresponding trained name. Fig 2(a) shows an original image and fig 2(b) shows tagged image.



Fig 2(a). Original image before tagging

```
aswathy@aswathy-HP-Pavilion-15-Notebook-PC: ~/Desktop/face_tagging
File Edit View Search Terminal Help
aswathy@aswathy-HP-Pavilion-15-Notebook-PC:~/Desktop/face_tagging$ source env/bin/activate
(env) aswathy@aswathy-HP-Pavilion-15-Notebook-PC:~/Desktop/face_tagging$ python main.py

==== Real Time Face Tagging Using Deep Learning ====

1. Register
2. Attendance
3. Display Users
4. Display Attendance
5. Train System
6. Exit

What would you like to do ?1
Enter User name: ASWATHY
Enter User Email address: aswathy@gmail.com
Enter User Mobile Number: 7034458808
Enter User Gender: female
```

Fig 3(a) Registering student details

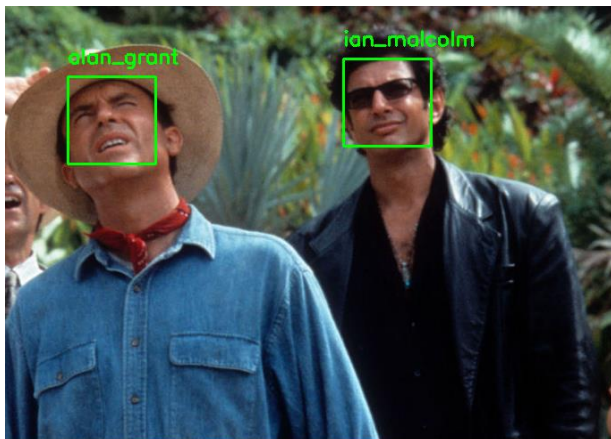


Fig 2(b) Output image after tagging

For automated attendance system, first we want to register the details of student shows in fig 3(a). then web camera starts capturing images of particular person (upto 200 image per face) and stored in database shown in 3(b). Next step is to train the datasets which is shown in fig 3(c). To mark the attendance, again the camera starts capturing image and trying to predict the face by comparing it with trained images which is already stored in the database. If matches found, tag that face and mark attendance. Face tagging is shown in fig 3(d) and attendance is shown in fig 3(e).

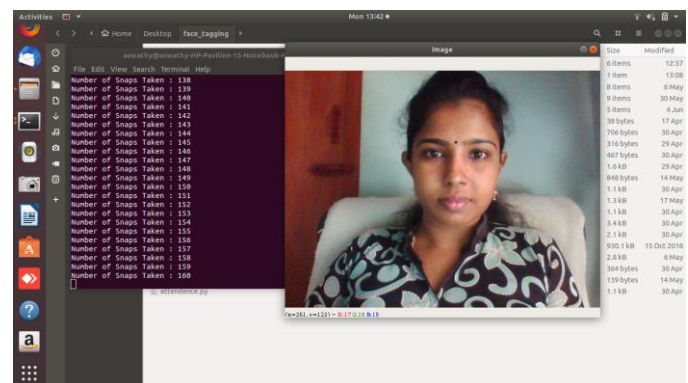


fig 3(b) Capturing images during registration

```
aswathy@aswathy-HP-Pavilion-15-Notebook-PC: ~/Desktop/face_tagging
File Edit View Search Terminal Help
HY_91.jpg_inception_v3.txt
INFO:tensorflow:Creating bottleneck at tf_files/bottlenecks/ASWATHY_female/ASWATHY_91.jpg_inception_v3.txt
HY_57.jpg_inception_v3.txt
INFO:tensorflow:Creating bottleneck at tf_files/bottlenecks/ASWATHY_female/ASWATHY_57.jpg_inception_v3.txt
HY_106.jpg_inception_v3.txt
INFO:tensorflow:Creating bottleneck at tf_files/bottlenecks/ASWATHY_female/ASWATHY_106.jpg_inception_v3.txt
HY_125.jpg_inception_v3.txt
INFO:tensorflow:Creating bottleneck at tf_files/bottlenecks/ASWATHY_female/ASWATHY_125.jpg_inception_v3.txt
HY_183.jpg_inception_v3.txt
INFO:tensorflow:Creating bottleneck at tf_files/bottlenecks/ASWATHY_female/ASWATHY_183.jpg_inception_v3.txt
HY_104.jpg_inception_v3.txt
INFO:tensorflow:Creating bottleneck at tf_files/bottlenecks/ASWATHY_female/ASWATHY_104.jpg_inception_v3.txt
HY_175.jpg_inception_v3.txt
INFO:tensorflow:Creating bottleneck at tf_files/bottlenecks/ASWATHY_female/ASWATHY_175.jpg_inception_v3.txt
HY_55.jpg_inception_v3.txt
INFO:tensorflow:Creating bottleneck at tf_files/bottlenecks/ASWATHY_female/ASWATHY_55.jpg_inception_v3.txt
HY_20.jpg_inception_v3.txt
INFO:tensorflow:Creating bottleneck at tf_files/bottlenecks/ASWATHY_female/ASWATHY_20.jpg_inception_v3.txt
HY_114.jpg_inception_v3.txt
INFO:tensorflow:Creating bottleneck at tf_files/bottlenecks/ASWATHY_female/ASWATHY_114.jpg_inception_v3.txt
HY_156.jpg_inception_v3.txt
```

Fig 3(c) Training the images

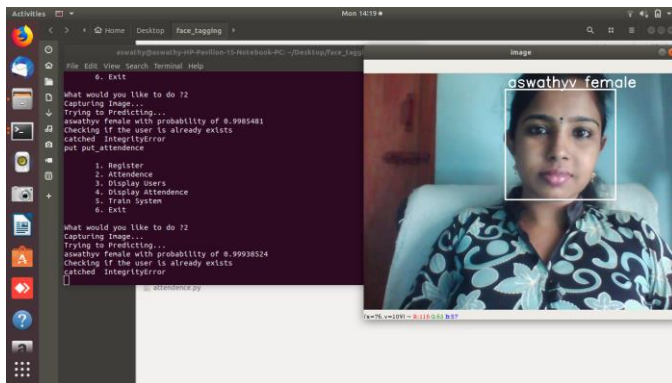


Fig 3(d) Tag the detected face

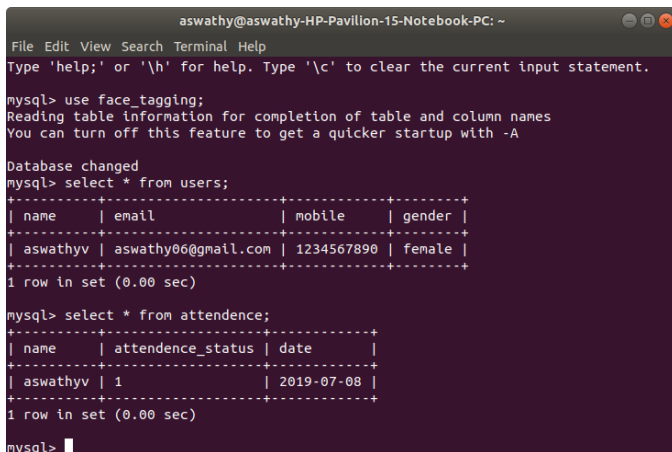


Fig 3 (e) Attendance marked automatically after tagging

IX. CONCLUSION

A convolutional neural network-based system was implemented to detect and recognise faces from video and tag it automatically. This can be useful in case of finding criminals from video surveillance cameras. The proposed system provides 95% accuracy for detecting and tagging faces from video. CNN based method is better as compared to other traditional existing methods.

REFERENCES

- [1] S. Ren, K. He, R. Girshick, and J. Sun, "Faster r-cnn: Towards real-time object detection with region proposal networks," in Advances in neural information processing systems, 2015, pp. 91–99.
- [2] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in Proceedings of the IEEE conference on computer vision and pattern recognition, 2016, pp. 770–778.
- [3] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going deeper with convolutions," in Proceedings of the IEEE conference on computer vision and pattern recognition, 2015, pp. 1–9.
- [4] S. S. Farfate, M. J. Saberian, and L. Li, "Multi-view face detection using deep convolutional neural networks," CoRR, vol. abs/1502.02766, 2015. [Online]. Available: <http://arxiv.org/abs/1502.02766>
- [5] K. Dang and S. Sharma, "Review and comparison of face detection algorithms," in Cloud Computing, Data Science & Engineering- Confluence, 2017 7th International Conference on. IEEE, 2017, pp. 629–633.
- [6] H. Li, Z. Lin, X. Shen, J. Brandt, and G. Hua, "A convolutional neural network cascade for face detection," in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2015, pp. 5325–5334.
- [7] L. Chi, H. Zhang, and M. Chen, "End-to-end face detection and recognition," arXiv preprint arXiv:1703.10818, 2017.
- [8] X. Fontaine, R. Achanta, and S. Susstrunk, "Face recognition in realworld images," in 2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), March 2017, pp. 1482–1486.
- [9] S. Agrawal and P. Khatri, "Facial expression detection techniques: Based on viola and jones algorithm and principal component analysis," in 2015 Fifth International Conference on Advanced Computing Communication Technologies, Feb 2015, pp. 108–112.
- [10] E. B. Putranto, P. A. Situmorang, and A. S. Girsang, "Face recognition using eigenface with naive bayes," in 2016 11th International Conference on Knowledge, Information and Creativity Support Systems (KICSS), Nov 2016, pp. 1–4.
- [11] N. Stekas and D. v. d. Heuvel, "Face recognition using local binary patterns histograms (lbph) on an fpga-based system on chip (soc)," in 2016 IEEE International Parallel and Distributed Processing Symposium Workshops (IPDPSW), May 2016, pp. 300–304.