International Journal of Research in Engineering, Science and Management
Volume-3, Issue-4, April-2020
www.ijresm.com | ISSN (Online): 2581-5792

Criminal Identification System Using Face Detection and Recognition

Abhinav Kumar[1], Aakash Jain[2], Aayush Chaudhary[3], Akshay Malik[4], Anuradha Taluja[5]
[1,2,3,4]*Student, Department of Computer Science Engineering, Meerut Institute of Engineering and Technology, Faridabad, India*
[5]*Professor, Department of Computer Science Engineering, Meerut Institute of Engineering and Technology, Faridabad, India*

***Abstract*: Face Recognition is a currently developing technology with multiple real-life applications. The developed system uses Convolutional Neural Networks in order to extract relevant facial features. These features allow to compare faces between them in an efficient way. The system can be trained to recognize a set of people, and to learn in an on-line way, by integrating the new people it processes and improving its predictions on the ones it already has. The accuracy in a set of 100 people has surpassed the 98%, and it has proven to robustly scale along with the number of people in the system. Face recognition has become a popular area of research in machine learning used typically in security systems but in multimedia information processing it is also very useful. Criminal face identification is one of its application.**

Crime rate is increasing at an exponential rate which leads to increase in number of criminals, which leads to great fear. Crime prevention by effectively identification of criminal is the main issue before the police and on the other hand the availability of police officers is inadequate. In this paper, facial recognition system for criminal recognition is anticipated using Inception model of convolutional neural network. This system will be able to detect face and recognize faces of criminals automatically in real time. This system would also just require single image of the criminal to recognise him, also known as one shot learning.

***Keywords*: Criminal Identification, Security, Convolution Neural Network, Inception model, One-shot learning.**

1. Introduction

A lot of security methods have been developed that help in possession of private data safe and restricting the chances of a security opening. Face recognition which is one of the uncommon biometric methods that have the merits of both high precision and low indiscreetness is a computer program that uses a criminal's face to repeatedly identify and confirm the criminal from a digital photo or a video frame from a real time video captured through webcam. Selected facial features are compared from the image. This is a widely used in biometric systems for authentication, authorization, verification and identification. A lot of company is using face recognition in their security cameras, access controls and many more. Facebook is using face recognition in their website for the persistence of creating a digital profile for the user accessing their website. In developed countries, the law enforcement creates face database to be used with their face recognition system to relate any suspect with the database. In many countries, most cases are investigated by using thumbprint identification to identify any suspect for the case. However, because of unlimited knowledge through internet usage, most criminals are aware of thumbprint identification. This paper to proposes a criminal identification system where the identification of the criminal is done by face through face recognition rather than the use of thumb print.

2. Literature review

The best way to execute our proposal is by using CNNs (Convolutional Neural Network), which is a category of Neural Networks. CNN are a type of feed-forward neural networks made up of many layers. CNNs consist of layer or kernels or filters or neurons having learnable parameters or weights and biases. Each layer or kernel takes some inputs and performs convolution. The structure of CNN contains pooling, ReLU (Rectified Linear Unit), Convolutional and Fully Connected layers.

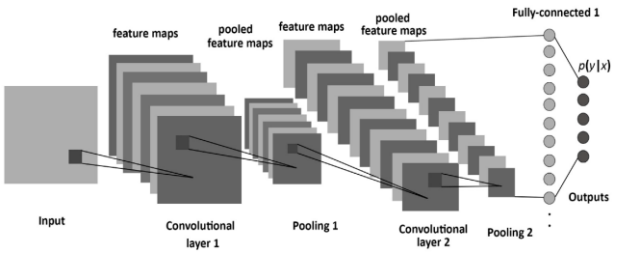

Fig. 1. Example of Convolutional Network

A. Pooling

The purpose of the pool layers is to achieve local intervention by minimizing the correction of feature maps. Each feature pool map is the same as the previous feature map. Their units consist of input from a small number of n × n units. This pooling window can be the size of an argument, and the windows can pass.

B. Convolutional Neural Network

The main purpose of the Convolution layer is to extract

features from the input / image data. This conversion protects the spatial coordinates between pixels by reading image features using small image frames. The input image is convoluted by using a set of learnable neurons. This produces a feature map or activation map in the output image and after that the feature maps are fed as input data to the next convolutional layer.

C. Activation Function

ReLU is a non-linear operation and includes units that use a rectifier. It works with a feature wise which means that each pixel is used and it also contains all the wrong values in the feature map by zero. To understand how ReLU works, we assume that there is a neuron input given as x and from where the multiplicity is defined as f (x) = max (0, x) in the neural networks.

D. Fully Connected Layer

The term Fully connected layer (FCL) means that all threads that belong to the previous layer are connected to all of the images in the following list. The output from the converter, pools, and ReLU components is of high quality image input. The purpose of using FCL is to apply these image classification features to various classes based on the training dataset.

E. One Shot Learning

Learning about one shot is a distinguishing function when a few, if a few examples, are used to distinguish many new examples in the future. These include activities seen in the field of facial recognition, where people should be properly distinguished by different lighting conditions, hair styles, facial features given one or a few template images. Today's face recognition programs approach the problem of single-shot learning with face-rich image recognition, called facial embedding, which can be easily calculated and compared verification and identification tasks. Historically, embedding has been studied for single-gun learning problems using the Siamese network. Training of Siamese networks on comparative loss functions resulted in better performance, later leading to triple loss operations used in the FaceNet program by Google which found temporal effects on benchmark recognition tasks.

In one shot study, one image per person is stored in a database, which is transmitted to the neural network to produce an embedding vector. This embedding vector is compared to the vector created for the criminal to be recognized. If there is a match between the two veins then the system accepts the criminal, otherwise that person is not in that database.

3. Proposed algorithm

The algorithm is mainly carried out in four steps as below:

Step 1: This CNN 1x1 filter out is used for dimension reduction. The idea at the back of 1x1 convolution is to maintain the input size (top and width) intact however cut back channels. Example: changing an 64x64x3 RGB image to 64x64x1 picture. Along with 1x1, different smaller but spatially spread-out filters are used like 3x3, 5x5 and 7x7. Since max-polling was a success to down sample the image, filters are implemented in parallel and ultimately all of the intermediate outputs are concatenated for the following stage. This makes the inception module wider in the middle but makes it easier to train the model as the depth of model is decreased.

Step 2: Considering the depth of the network it was bound to vanishing gradient trouble throughout back-propagation, hence auxiliary outputs had been tapped at middle layers and brought weighted average before including it to general loss, that is:

$$loss\_total = loss\_final + (1/3 * loss\_aux1) + (1/3 * loss\_aux2)$$

Step 3: If the face captured with the aid of a webcam has a similar 128-bit embedding vector as stored inside the database then it could recognize the criminal. All the pictures stored in the file gadget are converted to a dictionary with names as key and embedding vectors as value.

Step 4: To evaluate two pictures for similarity, we compute the distance among their embeddings. This may be finished by means of both calculating Euclidean(L2) distance or Cosine distance between the 128-dimensional vectors. If the gap or difference is much less than a threshold or hyperparameter, then both the faces are of same person otherwise not.

For any data science projects fetching the appropriate dataset is the crucial aspect for achieving better performances. In our experiment of facial recognition we have taken 10 individual classes of Labeled Faces (LFW) dataset. In ideal case each class can have thousands of images, but our system cannot handle those many computations. We also taken a student database of 20 classes from our college with each class having exactly 60 images. We have taken equal no of images because no of images in one class varies largely with the other class in LFW dataset. Now from Jupyter Notebook we have downloaded the Labeled Faces full dataset and stored in a directory. We have adopted the process of doing images deep-funneled, it is a process of aligning images this results in better representation of the dissimilarities between face images by reducing the variability in the background of images. The one of the most problems of recognizing a face in images is a intra class variability between images of the same individual can often be greater than inter class variability between images of different individuals. The proposed deep-funneled method has been expressed the improved accuracy on the task of facial identification compared to un-aligned face images.

4. Results and Discussions

A. Homepage

Homepage is the main page of Criminal Identification System application. It contains two buttons for: Adding New entry of a criminal and identify previously registered face.

Fig. 2. Homepage

B. Criminal Registration

Criminal Registration page will ask the user to select at least 5 images of the criminal that needs to be registered and also provides input form for providing various details of the criminal like his First name, Last name, Previous Criminal Record, Contact number etc. After selecting images and filling details, user will click register. The criminal will be successfully registered if any error doesn't occur**.**

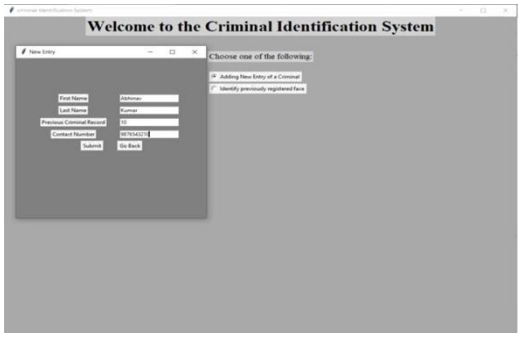

Fig. 3. Registration window

C. Detect Criminal Page

This page. allows the user to browse an image from the system and helps in detecting one or more criminals in it.

User can also see the profile of the criminal by clicking on detected criminal names.

Fig. 4. Detection window

5. Conclusion

In this project, we are able to detect and recognize faces of the criminals in a real time video through webcam or through a picture. We have trained inception model of convolution neural network to recognize an image. The major advantage of this method is its high recognition accuracy.

6. Future scope

Criminal identity to use face recognition has a lot of potential in the future to detect criminals. In this paper we introduced a face recognition gadget using the Inception-V3 version of the TensorFlow platform to teach the CNN model. After doing all the testing the results obtained from our proposed gadget found the pictures of the human face well. Hopefully in the future we can improve this method and gain more accuracy.

References

[1] Li Deng and Dong Yu, Deep Learning: Methods and Applications, published as Foundations and Trends in Signal Processing, Volume 7, Issues 3-4, pp. 320-324.
[2] Chuanqi Tan et al, A Survey on Deep Transfer Learning, 2016.
[3] Francois Chollet Google, Xception: Deep Learning with Depthwise Separable Convolutions, 2017.
[4] Anuradha. S. G, Kavya.B, Akshatha.S, Kothapalli Jyothi, Gudipati Ashalatha, Automated Face Detection & Recognition for Detecting Impersonation of Candidate in Examination System, International Journal of Scientific & Engineering Research, Volume 7, Issue 3, March-2016.
[5] Arunkumar, Vasanth Kumar, Naveenly King, Aravindan, ATM Security Using Face Recognition, International Journal of Current Engineering and Scientific Research (IJCESR), vol. 5, no. 4, 2018.
[6] Liton Chandra Paul, Abdulla Al Sumam, Face Recognition Using Principal Component Analysis Method, International Journal of Advanced Research in Computer Engineering & Technology, Volume 1, Issue 9, November 2012.
[7] Patrik Kamencay, Miroslav Benco, Tomas Mizdos, Roman Radil, A New Method for Face Recognition Using Convolutional Neural Network, Digital Image Processing and Computer Graphics, vol. 15, 2017.
[8] Nalini Nagendran, Ashwini Kolhe, Security and Safety with Facial Recognition Feature for Next Generation Automobiles, International Journal of Recent Technology and Engineering, Volume 7 Issue 4, November 2018.
[9] Olga Russakovsky et. al, ImageNet Large Scale Visual Recognition Challenge, 2015.
[10] Karen Simonyan, Andrew Zisserman, Very Deep Convolutional Networks for Large-Scale Image Recognition, Published as a conference paper at ICLR, 2015.
[11] Christian Szegedy et al, Going deeper with convolutions, 2014.
[12] Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jonathon Shlens, et al. Rethinking the Inception Architecture for Computer Vision, 2015.
[13] Mart´ın Abadi et al. TensorFlow: Large-Scale Machine Learning on Heterogeneous Distributed Systems. 2016,
Available: www.tensorflow.org.
[14] Anaconda software Available: https://www.anaconda.com/download.
[15] Google Brain et al. TensorFlow: A System for Large-Scale Machine Learning. In proceedings of the twelfth USENIX Symposium on Operating Systems Design and Implementation (OSDI '16), 2016, pp. 265-267, 277.
[16] Indra den Bakker, Python Deep Learning Cookbook, published by packet publishing Ltd, pp. 265-278,365,422.
[17] Navin Kumar Manaswi, Deep Learning with Applications Using Python, Available: https://doi.org/10.1007/978-1-4842-3516-4, pp. 171-197.
[18] Josh Patterson and Adam Gibson, Deep Learning - A Practitioner's Approach, published by O'Relly media, Inc, pp. 1-141, 293-304.
[19] Ian Goodfellow, Yoshua Bengio and Aaron Courville, accompanied by the website www.deeplearningbook.org, pp. 96-365, 422, 438-473.
[20] https://medium.com/data-science-group-iitr/building-a-convolutional-neural-network-in-python-with-tensorflow-d251c3ca8117