Journal of Engineering Sciences
Vol 11, Issue 6,June/2020
ISSN NO:0377-9254

COMPARISON OF TWO DIFFERENT DEEP LEARNING MODELS FOR PERSON IDENTIFICATION FROM FACE IMAGES

Varnima E K[1], C Ramachandran[2]

[1]*Mtech Student, College of Engineering Thalassery, KTU, India*

[2]*Associate Professor, College of Engineering Thalassery,, KTU, India*

***Abstract*— Face recognition is an area where researches are still going on to make it more effective. YOLO v3 (You Only Look Once) and a modern face recognition with deep learning introduced by Adam Geitgey is compared here. This paper focus on the accuracy, number of training images, ability to add new class etc. Custom dataset is taken here. Number of classes for classification is twenty.**

***Keywords*— Face recognition, You Only Look Once (YOLO), Deep Neural Network**

1. INTRODUCTION

Nowadays biometrics is the most preferred criteria for user identification. Some of the biometric recognition system include face recognition, iris recognition, ECG recognition, DNA matching, hand geometry recognition, finger geometry recognition etc. User identification is done here from face images. The main specialty of human is recognizing a person even after a long gap. Here the machine is used to do the same. The main reasons why face recognition still used include simplicity, faster processing, seamless integration, non-intrusive property etc. There are many techniques that can be used for face recognition. The various techniques include Principle Component Analysis (PCA), Convolution Neural Networks (CNN), Artificial Neural Network (ANN), Siamese Neural Network etc. Face recognition can be used in areas where security is the major concern. The other applications include health care, marketing, identity authentication, surveillance, fraud detection, unlocking without passwords etc. The system will be an aid in banks, airports, institutions, hospitals etc. Attendance marking is a main advantage of face recognition.

This paper compares the two deep learning techniques such as You Only Look Once (YOLO) v3 and modern face recognition with deep learning introduced by Adam Geitgey. The factors for comparing both of the above mentioned techniques include accuracy, number of training images, tilted face recognition and ability to add new class. Section 2 explains about the technique YOLO, section 3 explains about the modern deep learning face recognition, section 4 provides an experimental result and comparison. The section 5 gives a conclusion.

2. YOU ONLY LOOK ONCE (YOLO)

It is Convolutional Neural Network (CNN) architecture used for real time multiple object detection without loss of too much accuracy [6]. There are three versions of YOLO. YOLO v3 is the version used in this work.

2.1. ARCHITECTURE

Darknet – 53 is the YOLO architecture [1] used. It contains 53 convolution layers on which 53 more layers are stacked to make the total number of layer 106. Each convolution layers are followed by relu activation function and batch normalization without using maxpool layer throughout the network. Bounding box prediction is done by finding the four coordinates [2].

Table I. Darknet – 53

	Type	Filters	Size	Output
	Convolutional	32	3 × 3	256 × 256
	Convolutional	64	3 × 3 / 2	128 × 128
1×	Convolutional	32	1 × 1	
	Convolutional	64	3 × 3	
	Residual			128 × 128
	Convolutional	128	3 × 3 / 2	64 × 64
2×	Convolutional	64	1 × 1	
	Convolutional	128	3 × 3	
	Residual			64 × 64
	Convolutional	256	3 × 3 / 2	32 × 32
8×	Convolutional	128	1 × 1	
	Convolutional	256	3 × 3	
	Residual			32 × 32
	Convolutional	512	3 × 3 / 2	16 × 16
8×	Convolutional	256	1 × 1	
	Convolutional	512	3 × 3	
	Residual			16 × 16
	Convolutional	1024	3 × 3 / 2	8 × 8
4×	Convolutional	512	1 × 1	
	Convolutional	1024	3 × 3	
	Residual			8 × 8
	Avgpool		Global	
	Connected		1000	
	Softmax			

2.2. DATASET

The dataset used is custom dataset which are taken using mobile cameras or high definition cameras. Nearly 700 images of each class are considered for training. Found difficulty in finding varying images of single class. Some of the images had only slight variation. Twenty classes were considered and to avoid any dilemma, same numbers of images were included in each class. 90 degree to -90 degree tilted face images were taken for training [4]. Some of the images taken for training are shown in figure 2.1. The images in the dataset are resized to 300 x 300.

Fig. 1 Dataset

2.3. LABELLING

Labelimg is the software used for labeling. The .txt file is made corresponding to each image. The first integer in the .txt file indicates the class id. The other value gives the coordinates for bounding box.

2.4. TRAINING AND TESTING

Training is done for extracting the features. The main features that differentiate a person's face from another are eyes, eyebrows, lips, nose etc. Feature extraction is done using darknet-53 architecture. OpenCV is used for testing. Training is done up to 30 epochs. 80% of the total image is trained and 20 % is used for testing. GPU-Nvidia GeForce GTX 1050 CPU-intel i5 is the system used for training.

3. MODERN FACE RECOGNITION WITH DEEP LEARNING

This is a pretrained network introduced by Adam Geitgey. The steps done in that model is just explained here to understand about the model. Built using dlib's state-of-art face recognition built with deep learning [3].

3.1. FACE DETECTION

To detect a face the first thing is to locate it. The locating of the image is done using Histogram of Oriented Gradients (HOG) [5]. For doing this first the image converted into black and white because the color image cannot give any important information for face recognition. Each pixel in the image is taken one at a time. The surrounding pixels of the corresponding pixels are also noted and an arrow is drawn in the direction the image is getting darker. This replaces whole image with many arrows and these arrows are called gradients. This is achieved by divided an image into square of 16 x 16. In each square the numbers of gradients pointing in major directions are counted and the square is replaced with arrow directions that were strongest. It gives a very simple representation. The HOG imaged formed after these steps is compared with the existing HOG patterns that was extracted from a bunch of training faces. This will correctly locate the face in the image. Multiples faces are also detected.

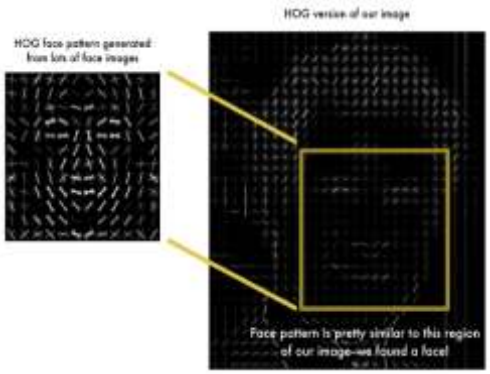

Fig. 2 Face detection

3.2. POSING AND PROJECTING

Now the faces are isolated from the image. The main problem is the tilted images looks different to the computer. Face landmark estimation [7] is used to warp each picture so that the eyes and lips are always in the same place. It will be easier to compare the faces. The basic idea in this technique is identifying the 68 specific points (landmarks) that exist on each face. Top of chin, outside edge of each eye, inner edge of eyebrows, nose-bridge, nose tip, upper lip, lower lip etc are some of the landmarks. Here a machine learning algorithm was used to find these 68 specific points on the face. Now the location of eyes and mouths are known. The image is slightly rotated, scaled, sheared so

that the eyes and mouth are centered as possible. Only affine transformations are done.

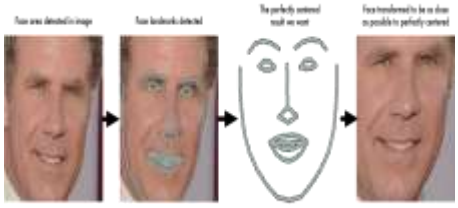

Fig. 3 Posing and projecting

3.3. ENCODING FACES

A deep convolution neural network is trained to generate 128 measurements for each face. The training process is done by loading a face image of a known person, another picture of same person and picture of a different person. The similar images will have same 128 measurements and different images will have dissimilar 128 measurements. This step is repeated many times on many images. Hence neural networks learns reliably generate 128 measurements of each person [8]. 128 measurements of each face are called as embeddings. This process of training a CNN to output face encodings requires lot of data and computer power. But once if the network is trained this can be used for finding the embedding of any unknown faces. The fine folks at OpenFace already did this and can directly use the published trained networks. So to get 128 measurements just run face image through that pretrained network.

3.4. FINDING THE PERSON'S NAME FROM THE ENCODING

Find the person in the database having closest measurements as that of the test image. Use simple linear SVM (Support-Vector Machines). Train a classifier that takes in the measurements from new test image and tells which known person is the closest match. The result of the classifier is the name of the person.

3.5. DATASET

There are 20 classes taken. The single image, as reference, of each class is to be included in the dataset. The encodings of these images are to be compared with the test image given.

Fig. 4 Dataset

4. EXPERIMENAL RESULT AND COMPARISON

The first technique used was YOLO. Here the dataset used is custom dataset and the training was done on this dataset. -90 degree to 90 degree tilt face images are fed in the dataset so, the system was capable to identify the person from 90 degree rotated images. The accuracy was 70 % which is low compared to the next technique. This was because of the custom dataset training. Accuracy can be increased only if more varying images of the same person are included. Most of the images have no much variation. The main drawback of this technique is requirement of more images for a single class. If a new class is to be added then the new dataset along with the old dataset of previous classes should be again trained. This is much time consuming work. The training was done up to 30 epochs. The collection of more images for a single class is tedious job because it is not possible to get thousands of varying images of a single person. Multiple faces can be recognized from single image. Another problem faced while working with YOLO is the bounding box. It is found that at the time of prediction bounding box may be too large and was not correctly around the face. And also in extreme cases distant faces are not recognized.

Fig. 5 Problems faced in YOLO

Fig. 6 Correct output

Now when the second technique is considered it is a pretrained network. It gave an accuracy of 99.38% on the Labeled Faces in the Wild benchmark. The main advantage of the model is when adding a new class only single image of the person is to be added in the dataset. The network is so accurate to recognize other image of the same person from this single reference image. By using this technique the requirement of more number of images and problem of adding new class is solved. This network can recognize the person from small degree tilted images but can't recognize the person from 90 degree tilt. Multiple people recognition was possible. The image of a person without beard and mustaches was given as the reference image and when an input image of the same person with both mustache and beard given the model predicted correctly. Images with wearing accessories like spectacles gave correct output. Person whose reference image is not fed in the dataset is predicted as unknown.

Fig. 7 Reference image and correctly predicted output

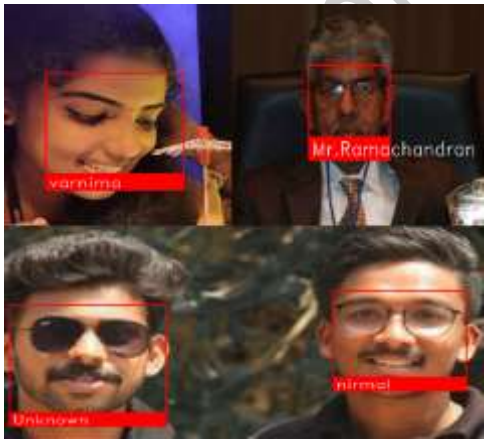

Fig. 8 Correct output

5. CONCLUSION

YOLO and a modern deep learning face recognition system proposed by Adam Geitgey are compared here. The steps used for recognizing a person using both the technique are explained in detail. It is found that the second method provide good accuracy compared to first and also the modern face recognition method can easily add a new class. The requirement of more dataset is in YOLO. 90 degree tilted images are not recognized by the modern face recognition method. The main problem for accuracy fall in YOLO is because of the custom dataset. Accuracy will be increased if more varying image of same class is trained. Face recognition for security purposes, unlocking, e-attendance systems etc.

REFERENCES

[1] J. Redmon, A. Farhadi, "YOLOv3: An Incremental Improvement" 2018 [online] Available:

[2] J. Redmon and A. Farhadi, "Yolo9000: better, faster, stronger," arXiv preprint, 2017.

[3] Adam Geitgey, "Machine Learning is Fun! Part 4: Modern Face Recognition with Deep Learning",medium,2016

CONFERENCE PROCEEDINGS

[4] D. Garg, P. Goel, S. Pandya, A. Ganatra and K. Kotecha, "A Deep Learning Approach for Face Detection using YOLO," 2018 IEEE Punecon, Pune, India, 2018, pp. 1-4.

[5] 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

[6] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in Proceedings of the IEEE conference on computer vision and pattern recognition, 2016, pp. 779– 788

[7] 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

[8] 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