COMP3007 COMPUTER VISION COURSEWORK REPORT

Wei Chuen Sea

20128825 HCYWS1

ABSTRACT

Face recognition is a rapidly growing field of research, it is also widely used especially in the security and biometric industry. The applications of facial recognition are very broad, and its growing popularity can be seen in everyday life. Due to the creation of large face datasets by large corporations such as Facebook and Google, along with the increasing processing powers of GPUs, much attention has been drawn to the research and development of face recognition techniques.

1. INTRODUCTION

This paper explores 2 face recognition techniques which aims to learn features from 100 training face images and correctly identify them among 1344 testing images. The methods are inspired from several notable and exciting works which will be elaborated in detail in this paper.

2. METHODS/METHODOLOGY

2.1. Method 1 (Transfer learning with VGG-Face)

2.1.1 Deep Neural Network

An Artificial Neural Network (ANN) is a network of connected nodes called artificial neurons. These artificial neurons receive an input signal and outputs a signal to other artificial neurons connected to it. The neurons are typically aggregated into input, hidden and output layers. Deep Neural Networks (DNN) are a type of ANN which has many layers in the hidden layer. The most common DNNs used for computer vision tasks are the Convolutional Neural Networks (CNNs)

A DNN was used as a face recognition technique because it has the ability to model complex relationships between the input and output which is suitable for face recognition. Furthermore, there is an abundance of pre-trained DNN models which has been trained on large face datasets. This is especially useful because the training dataset provided is very scarce compared to the testing dataset.

2.1.2 VGG face

The VGG-Face [1] pre-trained model was used as a transfer learning method for the face recognition task. VGG-Face is a very popular face recognition model which was developed by the Visual Geometry Group from Oxford University. VGG-Face has been trained on a huge face dataset which contains 2.6M images over 2,622 identities.

The training process as described by [1] involves using a softmax activation layer to classify the faces of the 2,622 identities involved. Subsequently, the softmax layer is removed, which changes the output of the network to be a face embedding. Then the model is further trained using a triplet loss function.

VGG-Face is composed of convolutional layers with ReLu activations along with some max pooling layers, and fully connected layers in the classifier layer.

One of the reasons that VGG-Face was used is because the layers used in the model was compatible with Matlab, this circumvents the need to implement an interface to call python scripts and makes the Matlab script created to be more user friendly and robust for markers.

The main advantage of using a pre-trained model is that it solves the problem of having too little training data.

As VGG-Face was already trained on a large face dataset, the weights and biases in the network has already “learned” facial features well. However, this model is used to classify the test data into 2,622 classes whereas the testing dataset provided has only 100 classes.

2.1.3 Transfer Learning

Transfer learning has been used in method 1 as a strategy to tackle the lack of training data provided. Transfer learning refers to a situation where what has been learned in one setting is exploited to improve generalization in another setting [2]. Transfer learning is normally used in deep learning, whereby a neural network is trained on a similar problem that is being solved. After training, one or more layers of the trained neural network is reused to solve the main problem. By using transfer learning, the features of faces can be “learned” despite having a small amount of training data.

As mentioned before, VGG-Face has 2622 possible outputs. Therefore, the output layers must be replaced with new layers which have suitable outputs to be able to correctly classify the images on the test dataset. The VGG-Face network has 41 layers in total, the layers that need to be replaced are the fully connected layer as well as the classification layer at layer 39 and 41. The fully connected layer replacement was configured to have a higher weight and bias learn rate, this was done to enable the classification layer to “learn” the correct outputs faster with a lower number of training epochs. As for the classification layer replacement, only the output labels were configured to match the test labels.

The weights of the front portion of the network in the newly modified network were also frozen during training. By freezing the weights, there will be lesser backpropagation required. This done to have a minor speed boost when training the model.

To summarize the steps taken for this method:

- Load in the VGG-Face model into Matlab
- Replace classification and fully connected layer to fit output size needed
- Freeze initial layers to speed up training
- Train network to train classifier

2.2. Method 2 (Viola-Jones Face Detection pre-processing and SURF feature extraction)

2.2.1 Viola-Jones Face Detection

Face detection is considered to be a type of object detection. The task of face detection is essentially to extract the location as well as sizes of a face contained in the image. Face detection and cropping is normally done before feature extraction [4] for face recognition tasks to decrease the computation needed.

The Viola-Jones face detection algorithm has been used to crop the faces detected. Viola-Jones makes use of 3 major components to be able to detect faces quickly and effectively.

Viola-Jones [6] introduces a new method of representing images, called “Integral images”. Integral image is a way to easily sum the values in a rectangle subset of a pixel grid. This image representation saves a lot of time when calculating the summation of all the pixels.

In a 24x24 detector window, there are almost 160,000 features which are present but only a small subset of those are important for face detection. To solve this problem, AdaBoost is used. AdaBoost is a classification scheme that works by combining weak learners or Haar-like features into more accurate ensemble classifier. AdaBoost essentially evaluates every weak learner and assigns higher weight to strong classifiers. The result would be a boosted classifier.

After AdaBoost, cascading system classifiers are used. A 24x24 window is used to slide over the input image and find any region which contains a face. The main purpose of a cascading classifier is to get rid of non-faces quickly, further increasing face detection.

The main consideration for using Viola-Jones are speed and improving feature extraction. Below are the results of the Viola-Jones algorithm before and after cropping.

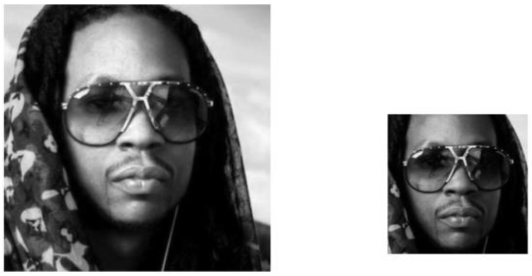


Figure 1— Test image 1 before(left) and after(right) Viola-Jones face detection and cropping

2.2.2 Feature Extraction using Speeded-Up Robust Features (SURF)

Image features describe the characteristics of an image, these characteristics are normally represented by vectors. There are several types of features which can be extracted, such as colour, texture and shape.

To accomplish the feature extraction task, SURF was used. SURF is an improved version of Scale-Invariant Feature Transform (SIFT) which was relatively slow. SURF approximates Laplacian of Gaussian (LoG) with Box Filter for finding the scale-space. The LoG approximation takes advantage of the Integral Images mentioned previously for faster calculation. SURF also uses the determinant of Hessian matrix for scale and location. SURF is ultimately used to detect interest points as well as the feature descriptors around the interest points [5].

To summarize the steps taken for this method:

- Crop faces with Viola-Jones face detection
- Use SURF feature extraction to obtain feature descriptors of images
- Find and store matches between all training and testing set
- Calculate maximum number of matches and assign the matched image as the output label

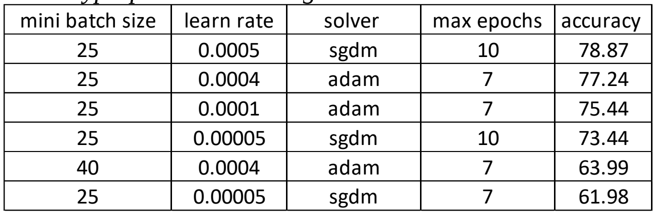
3. RESULTS/EVALUATION

3.1. Baseline Method

The baseline method managed to achieve an accuracy of 25%.

3.1. Method 1

3.1.1 Hyperparameter tuning

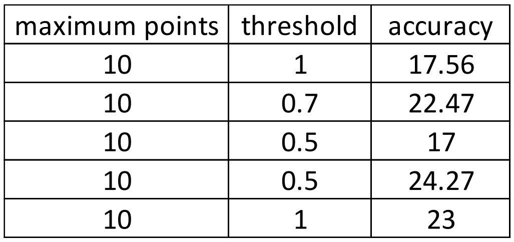


mini batch size	learn rate	solver	max epochs	accuracy
25	0.0005	sgdm	10	78.87
25	0.0004	adam	7	77.24
25	0.0001	adam	7	75.44
25	0.00005	sgdm	10	73.44
40	0.0004	adam	7	63.99
25	0.00005	sgdm	7	61.98

Table 1: Accuracy (%) after hyperparameter tuning for method 1 (run time has been excluded due to varying run time throughout the day)

An experiment has been carried out to determine the optimal mini batch size, learning rate, solver and the maximum number of epochs hyperparameters. The hyperparameters were selected based on the highest accuracy produced. The highest accuracy achieved with method 1 was 78.87%.

3.2. Method 2



maximum points	threshold	accuracy
10	1	17.56
10	0.7	22.47
10	0.5	17
10	0.5	24.27
10	1	23

Table 2: Accuracy (%) after hyperparameter tuning for method 2 (run time has been excluded due to varying run time throughout the day)

The optimal maximum points, threshold and accuracy hyperparameters were selected based on the highest accuracy produced. The highest accuracy achieved with method 2 was 24.27%.

4. DISCUSSION

Based on observations made when implementing both methods, the images which made it difficult for the methods to classify images correctly has some similar characteristics which can be summarized as follows:

- the images had some parts of their face covered by hair or face accessories (occlusion)
- the images were extremely blurred or “not human-like” where some of them were drawings
- the subject has aged significantly
- some images had drastic pose changes

Another problem in this dataset is that for each label, there is only training example. This makes training a network to be challenging due to the lack of training data.

However, despite the challenges faced in the dataset. It is clear that method 1 significantly outperforms method 2 in accuracy. This may be attributed by the classification power of a deep learning network trained on large training set and modern network architectures.

Whereas for method 2, it can only be inferred that matching the image and assigning the highest number of feature points matches as the label is not a good enough classifier for face recognition. The accuracy obtained does not even surpass the baseline method despite taking as much time to train as method 1. Future works for method 2 could be to conduct more pre-processing on the data.

4.1. Method 1 pro and cons

Deep neural networks have good classification power. Furthermore, the features of the input image are automatically “learned” without the need for feature extraction. Subsequently, it is quite flexible and is able to solve similar problems despite having different input data, especially when transfer learning is used.

However, the drawbacks of deep neural networks is that it requires a huge amount of data to train. Also, loading a pre-trained model takes up a significant amount of hard disk space. The biggest drawback is that deep neural networks are problem specific, a face recognition network cannot be used to for stock price prediction for example.

4.2. Method 2 pro and cons

Viola-Jones despite being almost 2 decades old still has a high face detection rate, its speed has enabled researches to run real time face detection algorithms. Whereas SURF, when used correctly, can be used to match points in 2 images with high accuracy. Both of the algorithms used in method 2 are well known for their speed and reliability.

However, Viola-Jones takes a long time to train because of the number of feature classifiers that is required. While SURF features learned are task-specific and not very flexible.

5. CONCLUSION

In conclusion, 2 face recognition methods have been explored in this paper. This paper discusses the algorithms and machine learning concepts used to accomplish the task, as well as the steps required to replicate the work done. There are many possible future developments especially in method 2, such as hyperparameter tuning and feature detection improvements which will definitely be explored soon.

References

[1] O. Parkhi, A. Vedaldi and A. Zisserman, “Deep Face Recognition,” *Proceedings of the British Machine Vision Conference (BMVC),*

pp. 41.1-41.12, 2015.

[2] I. Goodfellow, Y. Bengio and A. Courville, Deep Learning (Adaptive Computation and Machine Learning series), The MIT Press, 2016.

[3] S. Gupta, K. Thakur and M. Kumar, “2D-human face recognition using SIFT and SURF descriptors of face’s feature regions.,” *The Visual Computer,* vol. 37, p. 447–456, 2021.

[4] B. Anand and P. Shah, “Face Recognition using SURF Features and SVM Classifier,” *International Journal of Electronics Engineering Research,* vol. 8, no. 1, pp. 1-8, 2016.

[5] H. Bay, A. Ess, T. Tuytelaars and L. V. Gool, “Speeded-Up Robust Features (SURF),” *Computer Vision and Image Understanding,* vol. 110, no. 3, pp. 346-359, 2008.

[6] P. Viola and M. Jones, “Rapid Object Detection using a Boosted Cascade of Simple Features,” in *Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition. CVPR 2001,* 2001.