Video-based face recognition based on deep convolutional neural network

Yilong Zhai
Faculty of Information Technology
Beijing University of Technology
Beijing, China
+86 18501306729
iceman1294@hotmail.com

Dongzhi He
Faculty of Information Technology
Beijing University of Technology
Beijing, China
+86 18501293512
victor@bjut.edu.cn

ABSTRACT
With the rise of artificial intelligence in recent years, the field of object recognition is making rapid progress. Face recognition is a major subarea of object recognition which has already played a significant role in our life. However, despite the extensive study on the field of face recognition, video-based face recognition is still a tough area which needs further research. In this paper, we propose a model based on deep convolutional network for video-based face recognition. Our model split video images into two sets, a set of key frames and the other set is made up with non-keys, for different tasks to lower the computational complexity of the model. Besides, we introduce spatial pyramid pooling and center loss to our method for classification task. Our method presented in this paper reached an accuracy of 96.06% on YouTube Faces dataset. The results indicate our approach possesses high precision as well as a strong real-time performance.

CCS Concepts
• **Computing methodologies → Artificial intelligence → Computer vision → Computer vision problems → Object recognition.**

Keywords
Face Recognition; Deep Learning; Image Processing; Computer Vision; Biometric Identification

1. INTRODUCTION
Biometrics has already been widely used for personal identification. Compared with fingerprints or any other biological characteristic, the top advantage of face recognition is that it does not require the cooperation of the subject to acquire facial expressions. With the rapid development of data mining and the popularity of video surveillance, video-based face recognition has aroused widespread attention in the past few years. Compared with static image, video carries a larger amount of data and a more complicated scene.

IVSP 2019, February 25–28, 2019, Shanghai, China

ACM ISBN 978-1-4503-6175-0/19/02…$15.00
DOI: 10.1145/3317640.3317655

With the support of big data and the development of artificial intelligence in recent years, deep neural networks have been widely employed in the field of image recognition [1]. Deep neural network based algorithms are more robust than the traditional manual feature based methods due to their strong nonlinear fitting ability, which is possible for the networks to obtain a powerful feature expressions. It's Yann LeCun who proposed to use the convolutional neural network to classify handwritten font images for the very first time and achieved promising result. Deep learning for image recognition made a breakthrough in 2012 when Geoffrey Hinton and his team claimed first place in the ImageNet image classification contest using a deep architecture called AlexNet [2]. Deep learning has become one of the hottest research topics since then. With the successful application of deep learning in image detection and classification, deep neural networks have also been introduced into the field of face recognition and made breakthrough. In 2014, Facebook proposed a convolutional neural network based face recognition method named DeepFace [3]. The recognition accuracy of DeepFace on the LFW dataset is nearly human-level.

2. RELATED WORK
Most of the existing video-based face recognition methods generally treat each video as a collection of images and perform image set matching for video-based face recognition. These methods can be divided into two categories: flow-based methods and frame-based methods. For flow-based methods, each video is modelled as a manifold, and the similarity between each set of videos is obtained by calculating the distance between the manifolds [4]. In previous studies, many models were used for manifold modelling, such as probabilistic appearance manifolds [5] and Grassmann manifolds [6]. These methods, however, consider each frame of the video as equal importance, which means that the accuracy of flow-based method can be greatly reduced as the number of low quality frames increases. On the other hands, frame-based methods usually employ a face detector in advance to extract the area of face which is applied for further recognition tasks [7-10].

Traditional face recognition methods rely on hand-made descriptors. These methods perform decently under restricted condition, but not good enough when it comes to uncontrolled condition (e.g. complex natural scene). With the success of deep learning in image classification, researchers began to apply deep neural networks to the field of face recognition. In 2014, Facebook AI Lab proposed a model named DeepFace for face recognition. DeepFace uses 3D models and convolutional layers for feature extraction, and perform classification with softmax function. The training dataset for DeepFace is combined with

approximately 4 million images of human faces belonging to more than 4,000 people. DeepFace achieved an accuracy of 97.35% on the LFW dataset.

In the same year, a research team from the Chinese University of Hong Kong which leads by Professor Tang Xiaoou proposed a network called Deep ID [11]. Deep ID model firstly divides the image into 60 patches, then extracts the face features via multiple different deep convolutional networks, and finally uses Joint Bayesian for face recognition. The team continued to release improved series of Deep ID such as Deep ID2 and Deep ID2+. The Deep ID2 exceeds human-level accuracy on the LFW dataset with an accuracy of 99.15%.

Google put forward an end-to-end model for face recognition named FaceNet in 2015 [12]. FaceNet uses a triplet loss function for classification instead of regular softmax loss to reduce intra-class variation, while each face is expressed by a feature vector. The goal of triplet loss is to ensure that feature vector of the current facial image are closer to the feature vectors of other images of the same person than to the feature vectors of images of other people. With this manner, the Euclidean distance of two features could be used to calculate how similar those two faces are. FaceNet achieved an accuracy of 99.63% on the LFW dataset and 95.12% on the YouTube Faces dataset.

3. METHOD

We purpose a model using deep neural network for real-time face recognition. Figure 1 shows how a regular frame-based method works. First a face detector is used to extract facial regions from each frame of image. Then the model calculates the similarity of extracted faces and the labelled faces from the database, and chooses the most similar person as the result of classification. $p_i, i \in \{1,2,3,4\}$ represents the possibility of the extracted faces belongs to the i-th class.

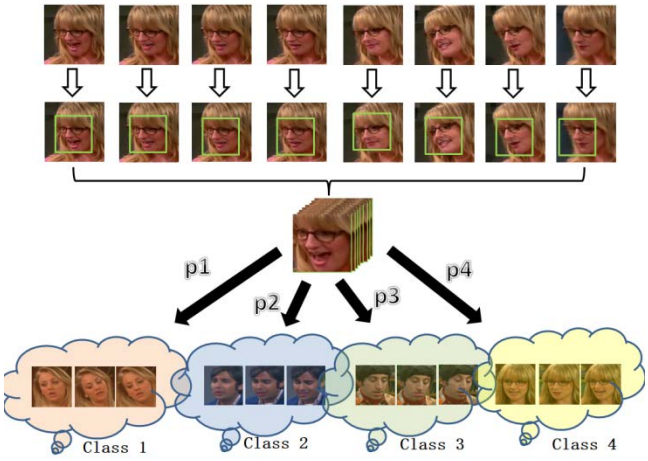

Figure 1. Frame-based method for face recognition.

However, face recognition can be a very complicated process. In order to meet the requirement of real-time performance, we do not apply face recognition for every frame of image. The overall framework of our model is shown in Figure 2. Our model divides frames of the video into two sets, a set of key frames and a set of non-keys. For the former, we apply the entire face detection, feature extraction and classification process. While for non-key frames, only face detection is performed, while the location of current face area is compared to its location of previous frame to judge if the two faces are same.

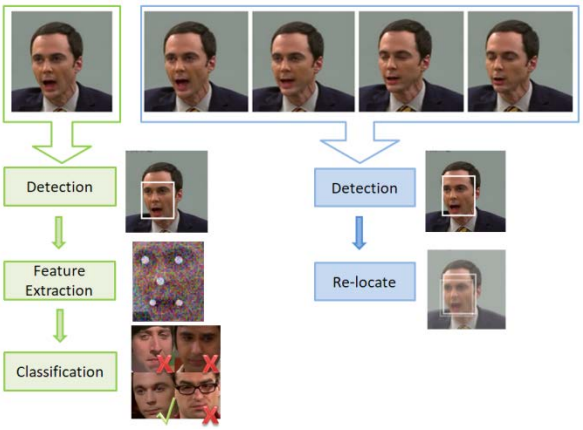

Figure 2. Overall structure of the model.

3.1 Face detection

Our face detector is implemented based on Multi-task Cascaded Convolutional Neural Networks (MTCNN) [13]. This can be summarized as following steps:

First, generate a number of potential regions (bounding boxes) of human face using convolutional neural network. Second, eliminate low rated bounding boxes and redundant bounding boxes, and refine the remaining candidates. Finally, return the final bounding box location of human face, as well as the coordinates of facial landmark.

Three cascaded convolutional neural networks are used to accomplish the face detection method. There are three tasks in MTCNN according to their paper: Face & non-face classification, facial bounding box regression, and facial landmark localization.

3.1.1 Face & non-face classification

This can be treated as a binary classification which can be resolved by cross entropy loss:

$$L_i^{det} = -\{y_i^{det} \log(p_i) + (1 - y_i^{det})[1 - \log(p_i)]\} \quad (1)$$

Where p_i represents the probability of the area of bounding box to be a face region, and y_i^{det} stands for the ground truth.

3.1.2 Facial bounding box regression

This can be treated as a regression task which could be resolved by L2-norm loss function:

$$L_i^{box} = ||\widehat{y_i^{box}} - y_i^{box}||_2^2 \quad (2)$$

Where $\widehat{y_i^{box}}$ represents the location of the predicted bounding box, y_i^{box} represents the ground truth of the location of the face region. There are four coordinates in total, representing the four vertex of face area respectively.

3.1.3 Facial landmark localization

It's a similar task to facial bounding box refining, use L2-norm loss function once again to locate the facial landmark:

$$L_i^{mark} = ||\widehat{y_i^{mark}} - y_i^{mark}||_2^2 \quad (3)$$

Where $\widehat{y_i^{mark}}$ represents the location of the predicted facial landmark, y_i^{mark} represents the ground truth of the location of the

face region There are five coordinates in total, representing the five facial landmark (center of the left eye, center of the right eye, nose tip, corner of the left mouth and corner of the right mouth) respectively.

3.1.4 Overall model

The entire training target is set as below:

$$\min \sum_{i=1}^{n} \sum_{j\in\{det,box,mark\}} \alpha_j \beta_i^j L_i^j \quad (4)$$

Where α_j stands for the weight of the task j, $\beta_i^j \in \{0,1\}$ represents the type of the sample i. We revise the model based on the original MTCNN to meet our following requirements:

In order to get a more precise face area for further recognition, the weight of facial bounding box refining task is increased. We set $\alpha_{box} = 1$ in our model.

Since facial features are extracted in later process, we remodel the network to eliminate facial landmark detection task in P-Net and R-Net. Meanwhile, reduce its weight in O-Net. We set $\alpha_{mark} = 0.5$ in our model.

The final output of MTCNN only keeps a single bounding box. However, we want our model to be capable of multiple face recognition. Therefore, we choose to keep all high rated bounding boxes instead of outputting the best rated one.

In short, we aim to make our face detector works more like a bounding box generator for multiple faces

3.2 Face Recognition

As stated previously, we use deep convolutional network for face recognition. The deep model is based on Inception-ResNet-V1. A refined model of FaceNet is presented in this paper.

FaceNet is one of the state of the art algorithms for face recognition today. Its main contribution is the employment of triplet loss for classification instead of the traditional softmax. The structure of FaceNet is presented in Figure 3. We apply FaceNet to our video-based face recognition with several modifications to make it more adaptable to video-based face recognition task.

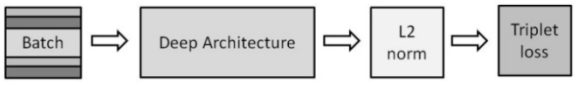

Figure 3. Structure of FaceNet.

Figure 4. Fixing the size of input image.

3.2.1 Spatial Pyramid Pooling

The classification layer of neural network normally requires a fixed-size input. This is generally solved by fixing the size of the input image of the entire network, as shown in Figure 4. However, this approach may cause the distortion of the input images by scaling or stretching, which could possibly do harm to the performance of the recognition.

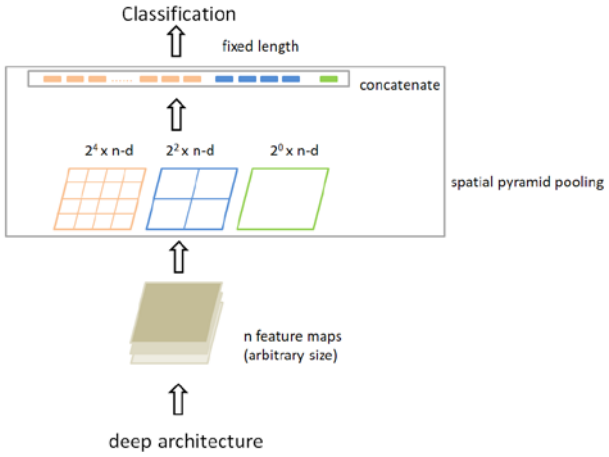

Figure 5. Spatial pyramid pooling.

We adopt the spatial pyramid pooling [14] to obtain fixed-length feature vectors as output of the network. Therefore, the model is able to accept image of arbitrary size as input.

Figure 5 shows how the spatial pyramid pooling layer works. Feature maps are splitted by filters of different sizes after they are extracted by the deep convolution network. Then we perform max pooling for features of each filter. Outputs are stitched into a fixed length vector as an input for classification. With this manner, it not only enables the neural network to accept an image of any size as an input, but also make the aspect ratio of the input image arbitrary.

3.2.2 Center loss

Deep neural networks generally use softmax for classification tasks, which loss function can be formulated as:

$$L_S = -\log \frac{e^{W_{y_i} x_i + b_{y_i}}}{\sum_{j=1}^{n} e^{W_j x_i + b_j}} \quad (5)$$

Where x_i represents the feature of the y_ith class, W represents the weight of the fully connected layer, while b represents the bias.

Technically, features should possess maximal inter-class variance and minimal intra-class variance for better performance in classification. Due to this reason, FaceNet uses Triplet loss instead of softmax loss. A triplet is made up with an image (anchor), an image of a face from the same person (positive), and an image of a face from a different person (negative), while distance between the anchor and the positive should be less than those between anchor and negative. This can be represented as:

$$||x^a - x^p||_2^2 + \alpha < ||x^a - x^n||_2^2 \quad (6)$$

Where x^a is the feature from anchor, x^p is the feature from positive and x^n is the feature from negative. α stands for the gap between positive and negative. A loss value is generated when x^a, x^p and x^n do not meet the condition. The total triplet loss can be formulated as:

$$\sum_{i=1}^{n} L(||x^a - x^p||_2^2 + \alpha - ||x^a - x^n||_2^2) \quad (7)$$

where

$$L(x) = \begin{cases} x, & if\ x > 0 \\ 0, & otherwise. \end{cases} \quad (8)$$

However, there are plenty combinations of x^a, x^p and x^n which can meet the condition, thus no loss value can be generated, which

leads to a slow convergence of the training process. In order to solve this, we use center loss instead to minimize the variance of intra-class of the extracted features. The formula of center loss can be expressed as:

$$L_C = \frac{1}{2}\sum_{i=1}^{n} ||x_i - c_{y_i}||_2^2 \quad (9)$$

Where c_{y_i} represents the center of the y_ith class. We use mini-batch to reduce the amount of calculation, and the calculation of the center is based on a part of data during each iteration. In this paper, a joint loss of center loss and softmax loss is employed for the neural network. The formula is displayed below. We set λ = 0.005 for our model:

$$L_{total} = L_s + \lambda L_C \quad (10)$$

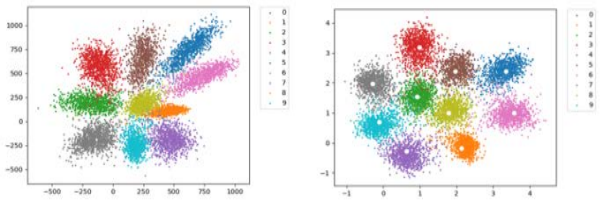

Figure 6. Features distribution using softmax loss (left), and center loss combined with softmax loss(right) on the MNIST dataset.

3.3 Face re-location

To reduce the complexity of calculation and improve the real-time performance, we begin to perform face recognition for the detected face since its appearance, and end the recognition process if there are N frames in a row share the same result of classification. These series of frames is defined as key frames, which are possibly followed by several non-key frames. In non-key frames, we only perform face re-location after detection.

Figure 7. Facial bounding boxes of two adjacent frames.

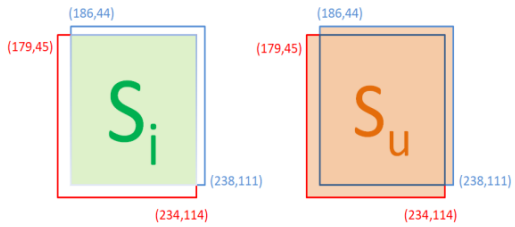

Figure 8. Intersection over Union. IOU = S_i /S_u.

In non-key frames, the bounding box of the face is generated and compared with the bounding box in previous frame, as shown in Figure 7. We calculate the ratio of the intersection area to the combined area of the two bounding boxes (IOU, Intersection over Union, shown in Figure 8). A number α is set as a boundary. When IOU > α, face is considered to be the same as the person in the previous frame. Otherwise, the current frame is treated as a new key-frame and repeats the process of recognition. We set N = 3 and α = 0.8. We also set the maximum count of consecutive non-key frames does not exceed 20, so we are able to reduce the amount of calculation while ensuring the accuracy of recognition.

4. RESULTS

Since a large amount of video frames is needed for training, we select YouTube Faces as our dataset. The YouTube Faces dataset captures a total of 3,425 videos featuring 1,595 different people, with a total frame count of more than 600,000. In our approach, the first 80% consecutive video frames of each video are used for training while the rest are used for verification. The results are presented in Table 1, which shows the recognition rate can be slightly improved compared to FaceNet.

Table 1. Comparing to FaceNet on YouTube Faces

Model	Accuracy(%)
FaceNet	95.12
Our method	96.06

When proceeding with videos with resolution of 1024x576, our model achieves an average processing rate of 25 fps by using the Nvidia GTX 970 graphics card. Typically, the frame rate of a film is 24 fps which means our results can marginally meet the standard of real-time performance. However, the processing rate could improve dramatically if a more powerful graphics card is used.

Figure 9. Face recognition for a frame contains multiple. faces.

5. CONCLUSION

In this paper, we propose a video-based face recognition model using deep convolutional neural network. Compared to FaceNet, we perform spatial pyramid pooling to deal with the feature vectors to make it possible for the model to accept images of any size as input. We also choose to combine center loss and softmax loss to train the neural network to minimize intra-class variance. Finally, our model divide the video into key frames and non-key frames while classification task is only carried out in key frames to reduce the amount of calculation and enhance the real time performance. Experiments indicate that our approach is well-suited for video-based face recognition.

6. ACKNOWLEDGMENTS

This research is supported by the Beijing Municipal Education Committee - the fund of the scientific and technological innovation platform. Fund number: PXM2015_014204_500211.

7. REFERENCES

[1] Sulis Setiowati, Zulfanahri, Eka Legya Franita, Igi Ardiyanto. A review of optimization method in face recognition: Comparison deep learning and non-deep learning methods.

9th International Conference on Information Technology and Electrical Engineering, ICITEE 2017. DOI=https://dx.doi.org/10.1109/ICITEED.2017.8250484

[2] Alex Krizhevsky, Ilya Sutskever, Geoffrey E. Hinton. ImageNet Classification with Deep Convolutional Neural Networks. *26th Annual Conference on Neural Information Processing Systems 2012*, NIPS 2012. DOI=https://dx.doi.org/10.1145/3065386

[3] Yaniv Taigman, Ming Yang, Marc'Aurelio Ranzato, Lior Wolf. DeepFace: Closing the gap to human-level performance in face verification. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, p 1701-1708, September 24, 2014. DOI= https://dx.doi.org/10.1109/CVPR.2014.220

[4] Wang, Ruiping, Shan, Shiguang, Chen, Xilin, Dai, Qionghai, Gao, Wen. Manifold-manifold distance and its application to face recognition with image sets. *IEEE Transactions on Image Processing*, v 21, n 10, p 4466-4479, 2012. DOI=https://dx.doi.org/10.1109/TIP.2012.2206039

[5] Lee, Kuang-Chih, Ho, Jeffrey, Yang\Ming-Hsuan, Kriegman David. Video-based face recognition using probabilistic appearance manifolds. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, v 1, p I/313-I/320, 2003.

[6] Lui, Yui Man, Beveridge, J. Ross. Grassmann registration manifolds for face recognition. *10th European Conference on Computer Vision, ECCV 2008*, October 12, 2008.

[7] Wu Wei, Liu Chuanchang, Su Zhiyuan. Novel real-time face recognition from video streams. *2017 International Conference on Computer Systems, Electronics and Control, ICCSEC 2017*. DOI= https://dx.doi.org/10.1109/ICCSEC.2017.8446960

[8] Parchami Mostafa, Bashbaghi Saman, Granger Eric. Video-based face recognition using ensemble of haar-like deep convolutional neural networks. Proceedings of the *International Joint Conference on Neural Networks*, v 2017-May, p 4625-4632, June 30, 2017. DOI=https://dx.doi.org/10.1109/IJCNN.2017.7966443

[9] Wen, Yandong, Zhang, Kaipeng, Li Zhifeng, Qiao Yu. A discriminative feature learning approach for deep face recognition. Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), v 9911 *LNCS*, p 499-515, 2016. DOI=https://dx.doi.org/10.1007/978-3-319-46478-7_31

[10] Rao, Yongming, Lu Jiwen, Zhou Jie. Attention-Aware Deep Reinforcement Learning for Video Face Recognition. Proceedings - *2017 IEEE International Conference on Computer Vision*, ICCV 2017. DOI= https://dx.doi.org/10.1109/ICCV.2017.424

[11] Sun, Yi, Wang, Xiaogang, Tang, Xiaoou. Deep learning face representation from predicting 10,000 classes. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, p 1891-1898, September 24, 2014. DOI=https://dx.doi.org/10.1109/CVPR.2014.244

[12] Florian Schroff, Dmitry Kalenichenko, James Philbin. FaceNet: A unified embedding for face recognition and clustering. *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2015*. DOI= https://dx.doi.org/10.1109/CVPR.2015.7298682

[13] Kaipeng Zhang, Zhanpeng Zhang, Zhifeng Li. Joint Face Detection and Alignment Using Multitask Cascaded Convolutional Networks. *IEEE Signal Processing Letters*, v 23, n 10, p 1499-1503, October 2016. DOI= https://dx.doi.org/10.1109/LSP.2016.2603342

[14] Kaiming He, Xiangyu Zhang, Shaoqing Ren, Jian Sun. Spatial Pyramid Pooling in Deep Convolutional Networks for Visual Recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, v 37, n 9, p 1904-1916, September 1, 2015. DOI=https://dx.doi.org/10.1007/978-3-319-10578-9_23