Research on Face Detection Technology Based on MTCNN

Ning Zhang
School of Computer Science and Engineering
Xi'an Technological University
Xi'an, China
E-mail: 645708716@qq.com

Junmin Luo
School of Computer Science and Engineering
Xi'an Technological University
Xi'an, China
E-mail:robertjm@126.com

Wuqi Gao
School of Computer Science and Engineering
Xi'an Technological University
Xi'an, China
E-mail:gaowuqi@126.com

***Abstract*—Face detection is an important research direction in the field of target detection. For the input image, the position of the face is returned. In order to complete the task of face detection using deep learning, data input, feature extraction and face feature detection are three steps, among which feature extraction is the most important part. By studying the basic principles of current mainstream target detection algorithms, this paper compares the characteristics of Two-stage and One-stage detection models and their application in face detection tasks. At the same time, MTCNN(Multi-task convolution neural network) is deeply analyzed and its implementation principle is introduced in detail. The real effect of MTCNN in face detection task is verified by experiments. The results of the model are compared with those of yolov3 model in the wider face dataset.**

Keywords-Face Detection; Deep Learning; MTCNN

I. INTRODUCTION

Object detection is one of the basic tasks in the field of computer vision, which has been studied for nearly 20 years. From the perspective of application, target recognition includes general target detection and saliency target detection. Among them, general target detection includes face recognition, pedestrian detection, vehicle detection, object detection, etc.; salient object detection should imitate the effect of extracting important objects from images in human vision on the results of target detection.

The traditional target detection algorithm is based on moving window traversal to select the region, and then uses hog, sift and other features to extract the features in the sliding window. Finally, SVM, AdaBoost and other classifiers [1] are used to classify the extracted features. Manual feature construction is more complex, and the detection accuracy is limited. The algorithm based on sliding window [2] has high computational complexity and limited detection effect.

In recent years, with the rapid development of deep learning technology [3], the target detection algorithm has changed from the traditional algorithm based on manual features to the detection technology based on deep neural network. At present, target detection methods based on deep learning mainly include Two-stage detection model and One-stage detection model. The Two-stage detection model is mainly based on the regional convolution neural network series. At present, the widely used detection algorithms mainly include fast RCNN target detection algorithm, fast RCNN target detection algorithm [4], feature pyramid network target detection algorithm and mask- RCNN target detection algorithm. These algorithms are generally composed of Two-stages. First, the target boundary box is generated on the image, and then the corresponding feature images of different boundary boxes are obtained from the clipped local images through CNN network [5]. Finally, the final boundary box of the target is obtained by further refining the feature map. One stage target detection algorithm mainly includes Yolo target detection algorithm series and SSD target detection algorithm [6], etc. different from Two-stage target detection method, One-stage target detection does not carry out regional recommendation, but uniformly carries out dense sampling in different positions of the image. In sampling, different scales and aspect ratios [7] are used to directly obtain the target from the feature map extracted from CNN by using regression algorithm Boundary box and classification probability. Compared with the Two-stage model, One-stage does not recommend regions, so it greatly reduces the amount of calculation, which is of great significance for some tasks with real-time requirements.

Face detection can be regarded as a special case of target detection [8]. It is a classic problem deeply studied in target detection. It has important application value in security monitoring, person certificate comparison, human-computer interaction [9], social interaction and other fields. For face detection task, Two-stage model has higher accuracy, while

978-1-7281-7083-1/20/$31.00 ©2020 IEEE
DOI 10.1109/ICCNEA50255.2020.00040

One-stage model has higher speed [10] because it does not recommend regions, but its accuracy is lower than that of Two-stage model.

At present, the main factors affecting the effect of face detection are as follows:

- Face pose: the position of the camera will affect the facial pose of the face image, which may lead to the eyes not in a horizontal line [11].
- Cover: for a single face image, it may be covered by a cover; for multiple face images, the face images may block each other or be blocked by other backgrounds.
- Illumination intensity: different illumination intensity will make the gray distribution of face image uneven, resulting in large local contrast, which will affect the effect of face detection.
- Facial expression: human facial expression will directly affect the effect of face detection.

II. Research on MTCNN

MTCNN is a multitask neural network model for face detection. In order to take into account the performance and accuracy, and avoid the huge performance consumption caused by traditional ideas such as sliding window and classifier, it first uses small model to generate target region candidate box with certain possibility [12], and then uses more complex model for fine classification and higher precision region box regression, and makes this step recursive to form a three-layer network, namely p-net , R-Net, o-net, to achieve fast and efficient face detection. In the input layer, image pyramid is used to transform the scale of the initial image, and p-net is used to generate a large number of candidate target area frames. After that, R-Net is used for the first selection and border regression of these target area frames, and most of the negative examples are excluded. Then, the more complex and higher precision network o-net is used to discriminate and regress the remaining target area frames.

A. MTCNN basic network structure

When MTCNN processes an image, it first performs the image resizing operation to scale the original image to different scales to generate image pyramid. Then the images of different scales are sent to the three sub networks for training in order to detect different sizes of human faces and realize multi-scale target detection.

P-net whole process proposal network, as shown in Figure 1 below, is the first subnet in MTCNN. The basic structure of the algorithm is a full connected convolution. The image pyramid constructed in the previous step of the algorithm is used to extract the preliminary features and calibrate the frame through a FCN, and roughly obtain the face candidate frame and frame regression vector. Then the candidate frames are regressed by the frame, and finally the candidate frames with high coincidence are merged by NMS algorithm. Most of the windows are filtered through the bounding box regression adjustment window and NMS. The main difference between FCN structure and general convolution network is that its convolution kernel size is 1*1, so that the network can accept images of any size as training set samples.

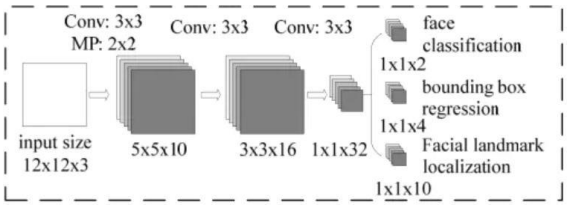

Figure 1. P-net structure

Full convolution network has two obvious characteristics. One is that it can accept images of any size, and the size of training samples and test samples can be inconsistent:

- When training the parameters of neural network model, because the training sample size of face detection algorithm in different scenes is not consistent, if the sample size is controlled manually, there will be some subjectivity.
- There may be different sizes and aspect ratios for the samples collected by different image acquisition devices. Changing the original size and aspect ratio by using the program will lose the image information.

The second feature is that the model training is more efficient, which avoids the repeated storage and convolution problems caused by the use of pixel blocks, especially the deep learning training network model needs a large number of training sets, which can reduce the training time and improve the efficiency.

The full name of R-Net is Refine-network, as shown in Figure 2. Its structure is more complex than the p-net network structure of the upper layer. The constraint conditions are mainly added, and the face prediction frame is screened again by the added constraints.

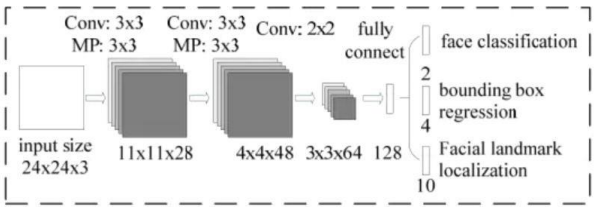

Figure 2. R-net structure

The R-Net network makes further judgment on the output window of the upper layer, and uses border regression and NMS algorithm to discard the face candidate frames with low score, so as to select several groups of locally optimal face candidate frames. It can be seen from Figure 2 that the R-Net network has one more full connection layer than the p-net network in the end. The function of the full connection layer is to output a 128 dimensions vector. Because of the full connection classification, R-Net will further filter the prediction box.

The full name of o-net is output network. The network structure is similar to R-Net, as shown in Figure 3. The function of o-ret network structure is to select the best candidate frame again, and output the five feature key points detected in the final. The network layer of o-ret is deeper than that of the previous layer, and the effect of o-ret on face detection is the best.

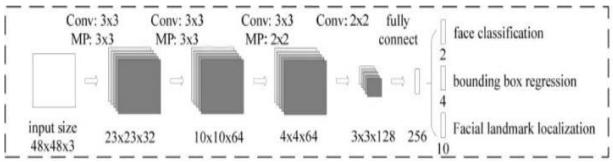

Figure 3. O-net structure

B. *Non maximum suppression*

The process of MTCNN algorithm is to generate multiple rectangular frames of different sizes in the face data through multi-scale transformation, carry out sliding detection on the face image, and extract and detect the features of the rectangular frames. Therefore, for the same potential face target, there may be many overlapping target candidate regions. In order to achieve a good face detection effect in face data, NMS algorithm is used in MTCNN algorithm.

NMS can be understood as local maximum search, as shown in Figure 4. Its purpose is to eliminate multiple highly coincident candidate frames of the same target to find the optimal target border. Firstly, the candidate frame with the highest confidence in the image is selected, and then the IOU values of other candidate frames and the maximum candidate frame are calculated. By setting a threshold, all candidate frames whose IOU value is greater than this threshold is deleted, avoiding the occurrence of repeated cross target candidate regions.

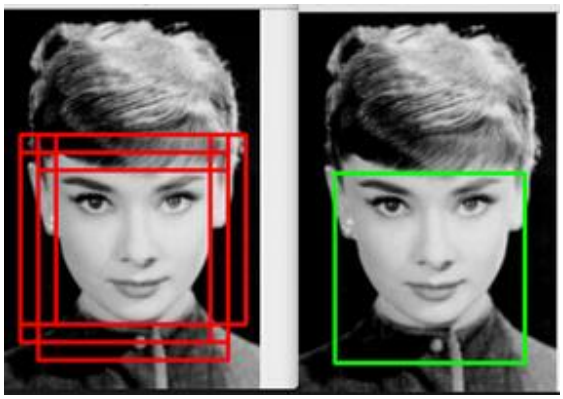

Figure 4. NMS example

The output of MTCNN can be decomposed into three parts: face classification, face candidate frame regression and landmark location, among which face classification and face candidate frame regression are the main contents of algorithm prediction.

For face classification, MTCNN sets the learning objective as a binary classification problem, and uses cross entropy loss function for each sample x_i:

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

p_i is the sample x_i predicted by neural network Probability of belonging to face the label of {1,0}.0 is the real image indicates that there is no face in the image, and 1 indicates that there is a face in the image.

For the candidate frame regression task, the bounding box regression algorithm is used to make the target frame predicted by the network model close to or coincide with the real target frame. MTCNN algorithm uses a 4dimensions vector (x, y, w, h) to output the predicted window, which represents the upper left coordinate of the prediction window and the width and height of the window respectively. For each human face candidate frame, the difference square loss function is used to predict the deviation between the candidate frame of network output and the nearest ground truth:

$$L_i^{box} = \left\| y_i - y_i^{box} \right\|^2 \quad (2)$$

III. MTCNN IMPLEMENTATION

A. *Experimental environment*

This experiment usesPython as the programming language, pycharm as the experimental debugging platform in Windows 10 operating system, uses tensorflow to train network model parameters, and uses some functions of OpenCV image library to assist in drawing and displaying face detection window.

Equipment	Model
CPU	Intel(R) Xeon(R) W-2123
memory	32G
operating system	Windows10
graphics card	NVIDIA-GeForce-RTX 2080 Ti
Deep learning framework	Tensorflow

B. *Data set*

In deep learning, training convolutional neural model usually needs a large number of labeled data as training set, according to different task objectives and application scenarios, including image, video, text, voice and other types. The data set used in this experiment is the wide face data set.

The wider face data set is the basic data set for face detection, which is mainly derived from the public data set wide, as shown in Figure 5.

Figure 5. wider face example

This data set was provided by the Chinese University of Hong Kong. They selected 61 event categories of wider and randomly selected 40%, 10% and 50% for each category as training, verification and test sets. It contains 32203 images and 393703 faces, which vary greatly in scale, pose and occlusion. The common feature is that the image resolution is high, the width of all images is zoomed to 1024, and the minimum labeled face is 7x7, all of them are color images; moreover, the face data of each image is too much, and each image contains about 12 faces on average, and there are many dense faces.

The evaluation method of MTCNN detection algorithm on test set is the same as Pascal VOC database. In order to meet the experimental requirements, need to code the wider face data into the standard format of VOC, and then input it into the network.

C. *Training process*

In the training stage, the image is randomly cropped before input into the network, and horizontal flipping is used as data enhancement strategy. The model was trained and iterated on the wider face dataset with 100 epochs. The data input size of p-net is 12 * 12 * 3, that of R-Net is 24 * 24 * 3, and that of o-net is 48 * 48 * 3. Data set annotation is divided into four categories, IOU < 0.3 is negative sample, IOU > 0.65 is positive sample, IOU is 0.4 ~ 0.65 is part of the face, IOU is 0.3 ~ 0.4 is an unclear area.

Two different learning rates, 0.001 and 0.0001, were used to train the iterative model. The learning rate is 0.005, and the learning rate is reduced once every 100 backward propagation. The experimental results show that when the larger learning rate of 0.001 is used, the model iteration is faster and the training time is shorter, but the final detection effect of the model is not ideal; when using 0.0001 as the basic learning rate, the iteration time of the model is delayed, but the final effect of the model is ideal.

Large batch can let the model see more samples in the same iteration, which can make learning more stable and achieve better results. But at the same time, with the increase of batches, the mean value of the overall sample noise remains unchanged, but the variance decreases, and the sample noise helps the optimizer to avoid the local optimum and improve the overall generalization ability; in this experiment, the batch size is set to 32, that is, 32 samples are propagated backward each time.

D. *Result analysis*

After the training model is verified on the wider face verification set, the map obtained is 85.7%, which is a good result. For the image input into MTCNN, the boundary box and five feature points of the face can be returned, as shown in Figure 6.

Figure 6. experimental result

At the same time, it is trained on the same data set as yolov3 algorithm, and the results are shown in Figure 7.

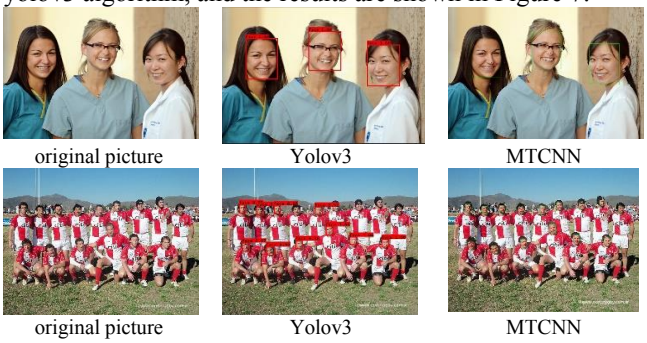

Figure 7. comparison of Yolov3 and MTCNN

In Figure 7, the image in the left most column is the original image, the image in the middle column is the image of yolov3 model target detection results, and the image in the rightmost column is the MTCNN model target detection result. It can be seen from Figure 7 that MTCNN and yolov3 algorithms can output the position of faces in the image accurately when the number of faces in the image is small. When there are many faces in a picture, yolov3 has a high rate of missing detection, and MTCNN can also get the position of face more accurately.

In extreme cases, as shown in Figure 8, yolov3 algorithm can hardly recognize the faces in the image when there are many faces with small targets in a picture. Although MTCNN has missed the detection, it can still recognize most of the faces.

Figure 8. Multi face comparison

MTCNN > Yolov3

IV. Conclusion

This paper mainly studies the application of MTCNN in face detection. Through targeted learning have a deep understanding of the hot field of computer vision, and mastered the basic principles of a variety of target detection algorithms. And through the implementation of the algorithm understand the core idea of face detection task and the problems to be solved in the future.

Compared with the traditional high-resolution face image, off face, low pixel and uneven illumination conditions bring some difficulties and challenges to face detection and facial key point location. It is easy to face missed detection, false detection and inaccurate key point positioning. Therefore, the research on how to improve the detection accuracy of off face, low resolution face image and the positioning accuracy of facial key points under unconstrained conditions can broaden the application scope of face detection in machine vision field. At the same time, the research on the detection speed of MTCNN can increase its availability in the tasks with high real-time requirements. In the follow-up work will do more in-depth research from these two directions.

References

[1] Li Peikang, Yuan Fangfang, A brief review of target detection methods [J]. Journal of science and technology, 2020 (18): 157.

[2] Li Furing. Multi feature fusion based on mtcnn for student fatigue detection [J]. Information technology, 2020, 44 (06): 108-113 + 120.

[3] Yang Shaopeng, Liu Hongzhe, Wang Xueqiao. Small size face detection based on feature image fusion [J]. Computer science, 2020, 47 (06): 126-132.

[4] Hu Fuyuan, Li linyan, Shang Xinru, Shen Junyu, Dai Yongliang. A survey of target detection algorithms based on convolutional neural networks [J]. Journal of Soochow University of science and Technology (NATURAL SCIENCE EDITION), 2020, 37 (02): 1-10 + 25.

[5] Cao Yan, LiHuan, Wabg Tianbao. A survey of target detection algorithms based on deep learning [J]. Computer and modernization, 2020 (05): 63-69.

[6] Jiang Jiwei, He Mingxiang, Sun Kai. Real time face detection method based on improved yolov3 [J]. Computer applications and software, 2020, 37 (05): 200-204.

[7] Li Xiangbing, Chen Lian. Natural scene face detection algorithm based on improved fast RCNN [J / OL]. Computer Engineering: 1-8 [2020-07-11] https://doi.org/10.19678/j.issn.1000-3428.00.56723.

[8] Wu Jiyu, Chen Shixin. An improved mtcnn face detection algorithm [J]. Software guide, 2019, 18 (12): 78-81.

[9] Shi Chao, Li Shengcang, Li li. Research on face target detection and tracking algorithm based on mtcnn and improved KCF [C]. Chinese society of command and control. Proceedings of the 7th China command and control conference in 2019. China Society of command and control: China Society of command and control, 2019:322-326.

[10] Lin Jingdong, Wu Xinyi, Cai Yi, Yin Hongpeng. A survey of convolutional neural network structure optimization [J]. Actaautomatica Sinica, 2020, 46 (01): 24-37.

[11] Yao Kun. Overview of face detection technology [J]. Electronic technology and software engineering, 2014 (13): 122.

[12] Dong Yachao, Bao Jun, Liu Hongzhe. Research progress of small target detection technology based on deep learning [C]. Network application branch of China Computer Users Association. Proceedings of the 23rd annual meeting of new network technology and application in 2019 of network application branch of China Computer Users Association. Network application branch of China Computer Users Association: Key Laboratory of information service engineering, Beijing United University, 2019: 172-176.