ATLANTIS PRESS

Advances in Intelligent Systems Research (AISR), volume 155

2018 International Conference on Advanced Control, Automation and Artificial Intelligence (ACAAI 2018)

An Efficient Method of Face and Keypoint Detection Based on Shared Network

Xiaogang Tian, Xiaoye Fan, Feixue Tang* and Xixin Cao

School of software & Microelectronics, Peking University, China

*Corresponding author

***Abstract*—We propose a method to perform face detection and facial keypoint detection in parallel. On the basis of the traditional object detection framework Faster R-CNN for face detection, we have designed a facial keypoint detection network with the same base network. And the two networks share the weights of the base network by the method of alternate training. Depends on the feature map in Faster R-CNN, after detecting the face box, we can find the feature patch corresponding to the face box directly. In this way, we make the two models can be fused together. Compared with the traditional serial process, our model run faster more than 20%, with the same accuracy.**

Keywords-face detection; facial keypoint detection; fused network;

I. INTRODUCTION

Face detection and recognition is a hot spot currently in the field of computer vision. The development of face detection has gone through a variety of traditional algorithms such as methods based on HOG [1], DPM [2, 3] or cascade AdaBoost classifiers [4, 5]. In recent years, Deep Learning shows effective and high correctness on image processing and became state of the art of object detection. This development provides researchers many new ideas for face detection. Most of the current face recognition systems are based on deep learning object detection framework (e.g. Fast-RCNN [6], Faster-RCNN [7], SSD [8]). The normal pipeline is to detect the faces in the scene images to obtain the face boxes with face detection model. Then estimating facial keypoint of each box by keypoint detection model. This pipeline obviously is serial, which need to extract image features repeatedly in different models. However, since the primary responsibility of both networks above is to extract facial features. The network weights should be similar for the same base network. So the series of network structure will cause a large extent to repeat the calculation and reduce the efficient.

To avoid this problem, we propose a network architecture by integrate two steps in the pipeline. The main method is to make feature extraction networks of two models be shared. Only one feature extraction is performed at the same time during face and keypoint detection process. In order to the two models to share the base network better, the face detection model and facial detection model has the same base network structure, which we show in *Base Network Sharing*.

II. RELATED WORK

A. Face Detection

In the early stage of face detection, as well as the general pattern recognition problem, face detection is based on Geometric feature. This kind of methods compose feature vector of face detection by the distance between the important facial features and the relative position as well as recognize face by matching the feature vectors. Since 1990s, we have been able to detect face on the ideal image acquisition conditions, user cooperation, and small and medium-sized face database. Turk and Pentland proposed based on Eigenfaces method [9], Belhumeur et al [10] proposed Finsherface method, Moghaddam [11] proposed a method of Bayesian probability estimation based on the Gemini space, Malsburg et al. [12] proposed the elastic graph matching technique based on Gabor transform. The methods achieved good results in the standard dataset.

In recent years, with the development of object detection framework based on deep learning, face detection has also been greatly improved. As a specific object detection task, due to its simplicity of only one kind of object, the detection effect in these framework becomes particularly prominent. Our approach is based on the state of the art framework Faster-RCNN.

B. Keypoint Detection

Facial keypoint detection is one of the most important technology in face detection field, which has been widely studied in recent years. Generally, the approaches were divided into two categories: classifying search windows and directly predicting keypoint positions [13]. The classifier in the first method is trained separately for each kind of feature point, which called component detector, and then make the decision by logistic regression. Due to the partial occlusion and blur of the face image, the multiple candidate regions will be detected falsely. Adding the shape constraints to optimize the algorithm in this case. The direct prediction method is based on the regression prediction of the part or the whole face image and the feature points are obtained under the condition of geometric constraints

Convolutional networks have been successfully used in facial keypoint detection. Because CNN have deep convolution network structure, and can extract high-level feature

information about a face, making the keypoint prediction more accurate. Tang et al. designed a three-layer cascade CNN to extract the feature points of the face, and achieved good results. But these methods are based on face images with partial backgrounds. In our task, we only focus on face patch detection. The detection task is simpler, so our keypoint extraction method uses only a very deep network called VGG16 for feature extraction before keypoint regression and perform well. And it is also in favor of fusion with the face detection network because of same base network.

III. OVERALL FRAMEWORK

We main focus on the fusion of shared base network. The whole structure of the model is shown in the FIGURE I. The whole network can be divided into three parts: face feature extraction, face detection and facial keypoint detection. The first part is shared by the second part and the third part. With the first part, the second part can form a face detection network. Similarly, the third part can form a facial keypoint detection network with the first part.

In the first part, we choose VGG16 as the feature extraction network. Due to its deep network structure and smaller convolution kernel size, it can extract high-level features of the image well. Our face detection network and facial keypoint detection network is both based on this network.

The second part is the face detection part. This part is based on Faster R-CNN. The goal is to get all face boxes of the image by putting the feature map get by the first part to roipooling layer and several full connection layers [7].

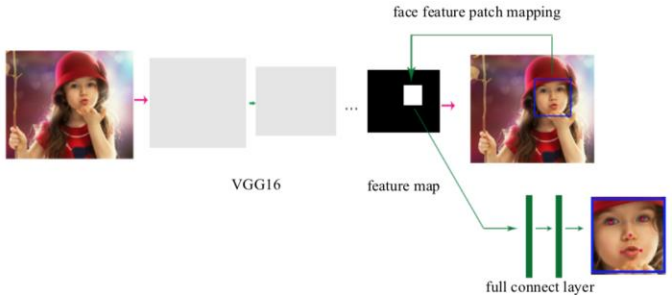

FIGURE I. THE STRUCTURE OF OUR MODEL.

In FIGURE I, we designed a keypoint detection network with VGG16. The base network of two detection network is the same and shared.

The third part is for facial keypoint detection. It consists of two fully connected layers and Euclidean loss layer. For each face box we get in second part, instead of inputting it into an independent facial keypoint detection network, our model maps the box back to the corresponding feature patches in the feature map get in first part. This feature patch is the face feature extracted by base feature-extract network (here is VGG16). And then predict facial keypoint by inputting the feature patch to several full connection layer. This make face detection network and facial keypoint detection network can share the base network and only one feature extraction is performed in the whole process.

A. *Base Network Sharing*

In the face detection network, the input image does not only contain the face, but also contains the background information, while the input image is usually filled with face in the keypoint detection network. In this way, the base networks of two models cannot be simply shared, since their weights has a certain difference. Nonetheless, it is possible to let them be shared, based on the following facts. During the face detection period, the network has been the focus of faces area, and works for the extraction of face features. Most background features is ignored by the base network. In the keypoint detection network, the base network is only concerned about the feature extraction of the face. Because of their similar functions, we can make them be shared with a certain method shown in *Network Training*.

B. *Face Feature Patch*

In the face box detection period, the face box we get is a four-dimensional vector (x, y, w, h), (x, y) is the upper left coordinates of the face box, w and h is the width and length of the face box detected. To get face feature patch, the four coordinates of the face box should be mapped to the corresponding location in the feature map taken by the base network.

VGG16 is mainly through the convolution layer network and pooling layer to extract image features, which all pooling layer size is 2×2. In the process of extracting the feature map, each time after convolution, the output layer size is only reduced by 1 unit pixels. While each after a pooling layer, the output layer height and width are reduced 1/2 of the input. It can be seen that the reduction in the size of the image is mainly through pool layers. the scale of map between box coordinate and face patch coordinate is decided by the pooling kernel size and the number of pooling layers.

Assuming that the face box (x, y, w, h) correspond to the coordinates on the feature map (xs, ys, ws, hs), (x, y, w, h) and (xs, ys, ws, hs) satisfy:

$$xs = x \times scale, ys = y \times scale \quad (1)$$

$$ws = w \times scale, hs = h \times scale$$

In VGG16, there are four pooling layers, which kernel size are all 2×2, so the scale is 1/16.

Compared with the serial mode, our model lifts the time associated with the image being processed. More specifically, it is proportional to the number of faces in an image. Assume that each pair of images contains n faces and the base network cost time t to extract the face feature. Our model will probably save nt time throughout the process. When n is large, it will greatly improve the detection efficiency.

C. *Network Training*

We train the face detection network and keypoint network by alternate training method. First, based on VGG16 on imagenet, we train Faster R-CNN face detection network. Then fine-tuning our keypoint detection network on base network of Faster R-CNN. This completes an alternate training. Next, based on the keypoint network in the above step, fine-tuning

Faster R-CNN. Finally, fine-tune the keypoint detection network again using the same method to end our training.

In the first alternating training, all the layers except base network layers has not yet trained, so the learning rate and the number of iteration are all big enough. Subsequent alternation training is only fine-tuning on the basis of the last turn, the learning rate and the number of iterations are relatively small and gradually reduced. Each of the alternate training will balance the weight of the base network for two features extraction and correct the parameters of the subsequent full connection layers to achieve smaller loss and better result.

IV. EXPERIMENT

We trained two models with different datasets independently. For face detection network, we use dataset WiderFace for training, which contains more than 30,000 images and more than 390,000 faces. The face keypoint detection section is trained using the processed CelebA dataset. We cropped the face patch of images, and make the corresponding transformation to the coordinates of the keypoint. The purpose of this is to make the training data more consistent with our model, and make the function of the model more specific. Simultaneously, it can upgrade the effect of the mode.

We use mAP to evaluate the accuracy of face detection. And facial keypoint detection perform is measured by positive rate and average error of each facial point. The average detection error is measured as

$$\mathrm{Err} = \frac{1}{5l}\sqrt{(x\text{-}x')^2 + (y\text{-}y')} \quad (2)$$

Where (x, y) and (x', y') are the ground truth and the detected position, and l is the width of the bounding box returned by our face detector. If an average error is smaller than 0.1, it counted as positive.

A. *Experiment Setup*

We used the method of *Network Training* to train. In the alternate training, when we carry out the third alternate training, we found that the loss is higher than the second training, which shown in FIGURE II. So we only conducted two alternate training. Throughout the training process, we compared the effects to the results of different iterations in TABLE III, which training has reached a certain convergence time. The result show that the number of iterations has little effect on the final result. We selected a relatively good result corresponding to the number of iterations.

TABLE I. THE LEARNING RATES OF TWO TRAININGS

	Face detection	Keypoint detection
first_alt_train	0.01	0.001
second_alt_train	0.001	0.001

In our experiments, the learning rate and the number of iterations of each alternate training are shown in TABLE I and TABLE II. In order to show the superiority of our model, we trained the Faster R-CNN face detection model and keypoint detection model separately with the parameters of the first alternate training in TABLE I and TABLE II. And tested in the same dataset using the serial method to compare the results with our model.

TABLE II. THE ITERATION NUMBER OF TWO TRAININGS

	Face detection	Keypoint detection
first_alt_train	70000	100000
second_alt_train	70000	60000

TABLE III. THE BOX MAP AND KEYPOINT POSITIVE RATE OF DIFFERENT ITERATION NUMBER

iter_number	50000	60000	70000	80000
face box mAP	95.01%%	95.01%%	95.03%	95.01%
keypoint pos_rate	91.77%	91.81%	91.77%	91.77%

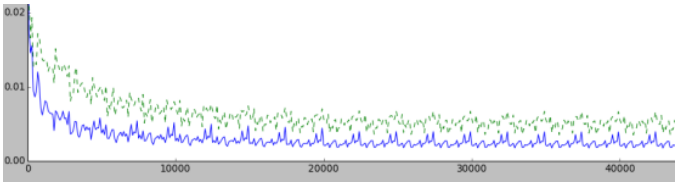

FIGURE II. THE KEYPOINT TRAINING PROCESS.

In FIGURE II. The line above (dotted line) is the third alternate training loss, the line below (solid line) is the second alternate training loss.

B. *Comparison with serial method*

We tested our model on AFLW dataset, which contains totally 1.7w + images and 2w + faces and corresponding keypoint. For the detection time of a single image, it may be unsure due to the impact of different machine and software environment. In order to test the performance of our model accurately and more convincingly. We compare the average time of a batch of images detection, which can eliminate uncertainty interference.

In order to show the superiority of our model, we trained the Faster R-CNN face detection model and keypoint detection model separately with the parameters of the first alternate training in TABLE I and TABLE II. And tested in the same dataset using the serial method to compare the results with our model.

TABLE IV. THE PERFORM COMPARISON OF TWO METHODS IN DIFFERENT FACE NUMBER CONTAINED IN A IMAGE

Faces/img	1	2	3	4	5
serial	0.1818	0.1838	0.1909	0.1911	0.1917
fusion(our)	0.1398	0.1419	0.1451	0.1447	0.1433
proportion	23.10%	22.80%	24%	24.28%	25.25%

We also tested the process time with different batch size, the comparison results shown in the FIGURE III. To be more convincing, we also added up the average processing time of all the images. To be observation clearly, we counted the average time for different face number contained in an image, in TABLE IV.

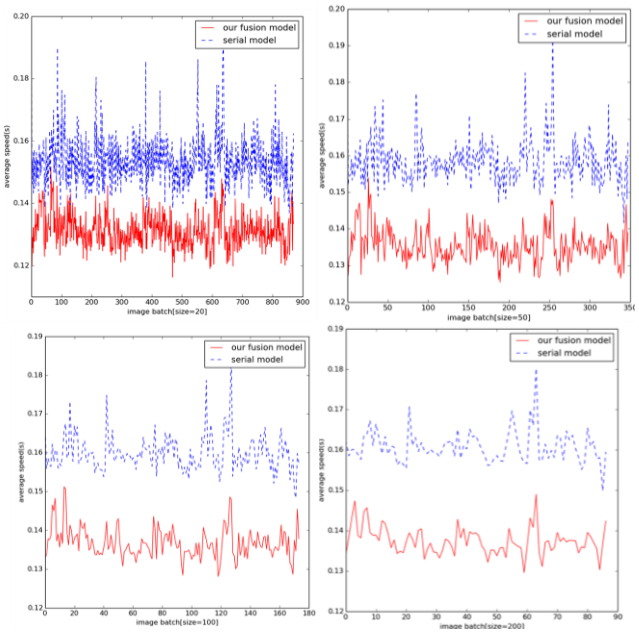

FIGURE III. THE AVERAGE DETECTION TIME FOR DIFFERENT BATCHSIZE IMAGES.THE SOLID LINE IS OUR MODEL AND THE DOTTED LINE IS SERIAL MODEL.

The run time of our fusion model has improved compared with serial, shown in FIGURE IV. When the number of face in an image is less, the speed didn't improve obviously. With face number contained in an image increasing. The speed increase of our model is more and more obvious. Because our model eliminates more facial feature extraction time.

We compared the result accuracy of two models. For face detection, we use mAP to measure. And for keypoint detection, we use the failure rate to measure. The comparison results are shown in the FIGURE V and FIGURE VI, respectively. It can be seen from the FIGURE V that the discriminate of keypoint detection result between the two models is not significant. And the effect of face detection is extremely closed to the current start-of-art detection frame (the Faster-Rcnn we trained in serial model can represent the state-of-art detection frame).

We also compared the keypoint detection results with other state-of-art methods. Compared with other methods, our keypoint detection method has realtively high failure rate, since the base network need balance face detection accuracy and facial keypoint detection accuracy in our model.

V. CONCLUSION

In this paper, we propose a method to efficiently carry out face detection and facial keypoint detection. Using shared feature extraction network, we can achieve more than 20% faster than serial method by ensured accuracy. However, for the smaller face box and keypoint not fully showed in face (e.g. half face), our model still has many insufficient.

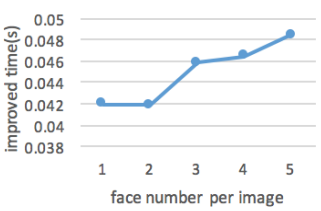

FIGURE IV. IMPROVED TIME WITH THE FACE NUMBER PER IMAGE.(LEFT)

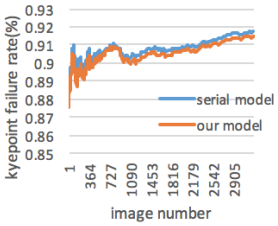

FIGURE V. THE COMPARISON OF KEYPOINT DETECTION FAILURE RATE.(RIGHT)

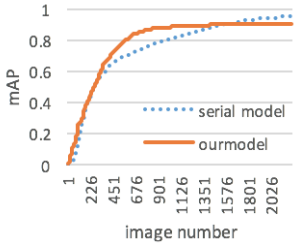

FIGURE VI. THE COMPARISON OF FACE DETECTION MAP.(LEFT)

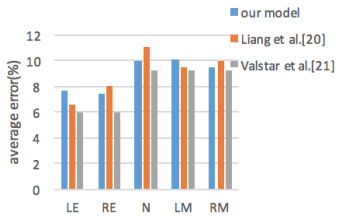

FIGURE VII. AVERAGE DETECTION ERRORS COMPARED WITH OTHER METHOD.(RIGHT)

REFERENCES

[1] N. Dalal and B. Triggs. Histograms of Oriented Gradients for Human Detection. in cvpr, 2005.

[2] X. Zhu and D. Ramanan. Face detection, pose estimation, and landmark localization in the wild. In CVPR, 2012.

[3] M. Mathias, R. Benenson, M. Pedersoli, and L. Van Gool. Face detection without bells and whistles. In ECCV. 2014.

[4] J. Li and Y. Zhang. Learning surf cascade for fast and accurate object detection. In CVPR, 2013.

[5] P. A. Viola and M. J. Jones. Rapid object detection using a boosted cascade of simple features. In CVPR, 2001.

[6] R. Girshick, "Fast R-CNN," in IEEE International Conference on Computer Vision. In ICCV, 2015.

[7] Ren, S., He, K., Girshick, R., Sun, J.: Faster R-CNN: Towards real-time object detection with region proposal networks. In NIPS, 2015.

[8] Wei Liu, D. Anguelov, D. Erhan3, et al. SSD: Single Shot MultiBox Detector. In ECCV, 2016.

[9] Wu B, Ai H, Huang C,et al. Fast rotation invariant multi-view face detection based on real adaboost. Sixth IEEE International Conference on. IEEE ,2004.

[10] Belhumeur P N, Hespanha J P, Kriegman D. Eigenfaces vs. fisherfaces: Recognition using class specific linear projection[J]. Pattern Analysis Machine Intelligence, IEEE Transactions on, 1997, 19(7): 711-720.

[11] Moghaddam B, Pentland A. Probabilistic visual learning for object detection[C]. Computer Vision, 1995. Proceedings. Fifth International Conference on. IEEE, 1995: 786-793.

[12] Jiang H, Wang J, Yuan Z, et al. Salient object detection: A discriminative regional feature integration approach. In CVPR, 2013.

[13] X. Cao, Y. Wei, F. Wen, and J. Sun. Face alignment by explicit shape regression. In Proc. CVPR, 2012.