Joint Face detection and Facial Expression Recognition with MTCNN

Jia Xiang
College of Computer Science and Engineering
HNUST
Xiangtan, China
iaiaxx@outlook.com

Gengming Zhu
College of Computer Science and Engineering
HNUST
Xiangtan, China
zhugm@hnust.edu.cn

***Abstract*—The Multi-task Cascaded Convolutional Networks (MTCNN) has recently demonstrated impressive results on jointly face detection and alignment. By using the hard sample ming and training a model on FER2013 datasets, we exploit the inherent correlation between face detection and facial expression recognition, and report the results of facial expression recognition based on MTCNN.**

Keywords-convolutional network; face detection; facial expression recognition; MTCNN

I. INTRODUCTION

Humans interact with each other mainly through speech, but also through body gestures, to emphasize certain parts of their speech and to display emotions. One of the important ways humans display emotions is through facial expressions which is key to nonverbal communication between humans. Computers and other electronic devices in our daily lives will become more user-friendly if they can adequately interpret a person's facial expressions, thereby improving human-machine interfaces. Facial expression recognition can be implemented in all computer interfaces.

Consequently, facial expression recognition (FER) has been widely studied and significant progress has been made in this field. In a 1971 paper titled "Constants Across Cultures in the Face and Emotion", Ekman et al. identified six facial expressions that are universal across all cultures: anger, disgust, fear, happiness, sadness, and surprise[1]. As seen in the previously described papers, most approaches developed to solve FER use customized features for short sequences of facial expressions, and there have been several advances in the past few years in terms of face detection, feature extraction mechanisms and the techniques used for FER.

Convolutional Neural Networks (CNNs) have gained much popularity in recent years for vision-related applications and have the potential to achieve some of the higher accuracies in FER[2]. However, most of the available face detection and facial expression recognition methods ignore the inherent correlation between these two tasks., but there are several existed works attempt to jointly solve face detection and face alignment[3, 4]. The famous method, represented by the multi-task cascaded convolution networks (MTCNN) of Kiapeng Zhang, et, al. It demonstrates impressive result on face detection and alignment (JDA). It have proposed a cascaded CNNs to integrate these two tasks by multi-task learning.

In this paper, we exploit the inherent correlation between face detection and facial expression recognition based MTCNN and implement the two tasks. The input into our system is an image; then, we use this framework to predict the face location and facial expression label which should be one these labels: anger, happiness, fear, sadness, disgust and neutral, and compare to a variety of other recent high-performing detectors.

II. RELATED WORK

Face detection is one of the mostly studied problems in vision. Viola and Jones[5] proposed a cascaded face detector, which is the first time to apply Haar-Like features in AdaBoost for training cascaded classifiers. It has a practical performance on real-time with higher accuracy than before, but is not able to effectively handle non-frontal faces and faces in the wild. To overcome this problem, more robust features methods have been proposed, such as HOG, Gabor, SIFT and SURF. Moreover, a simple strategy is to combine multiple features to enhance the robustness of detection. Zhu et al. [6] proposed a multiple deformable part models to detect faces with different views and expressions. However, they cost much time to train and are limited on the variety of scene. Recently, CNN often achieves better performance than traditional handcraft features methods in computer vision tasks. [7] used cascaded CNNs for face detection, is a boosted cascade structure. [8] applied the Faster R-CNN[9], one of state of the art generic object detection, has realized end-to-end optimization, and achieved promising results. [10] constructed a model to perform face detection in parallel with face alignment, and achieved high performance in terms of both accuracy and speed.

The problem of classifying emotions from facial expressions in images is widely studied. Similar to some of the methods of face detections, FER also can be implemented by the handcraft features methods. One of the first paper to apply neural nets to this end is the EMPATH paper from 2002[11], it proceeds by performing Gabor filtering on the raw images followed by various transformations and PCA before applying a 3 layer neural net. Several recent works on FER successfully utilize CNNs for feature extraction and inference. Yu and Zhang[12]

978-1-5386-3013-6/17 $31.00 © 2017 IEEE
DOI 10.1109/ICISCE.2017.95

achieved state-of-the-art results in EmotiW in 2015 using CNNs to perform FER. They used an ensemble of CNNs with five convolutional layers. [13] have also achieved state of the art results in FER. Their network consisted of two convolutional layers, max-pooling, and 4 Inception layers as introduced by GoogLeNet.

CNN have achieved some of the highest accuracies in both face detection and facial expression recognition. In this work, mainly Reference [4], we inherent face detection and facial expressions recognition based on MTCNN.

III. Overview of the MTCNN

A. Overview

The MTCNN, which has inherited correlation between the face detection and alignment to boost up their performance. It essentially consists of three parts: (1) a Proposal Network (P-Net) for generating a list of the candidate windows. Then, we use it for classifying face and non-face and estimate bounding box regression vectors to face location, and non-maximum suppression (NMS) candidate merge. (2) a Refine Network (R-Net), unlike P-Net, it doesn't obtain the region proposals but rejects a lot of false candidates. (3) It is similar to R-Net, called O-Net. In this network, it will output five facial landmarks' positions.

B. Training

As the same as the MTCNN, we leverage three tasks: face classification, bounding box regression, and facial emotions classification. The cost functions of face classification and bounding box regression are the same with MTCNN. Using the cross-entropy loss to solve the face classification, and the Euclidean loss for each sample.

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

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

Equation (1) is cost function of face classification, where p_i is the probability produced by the network that indicates a sample being a face. The notation $y_i^{det} \in \{0, 1\}$ denotes the ground-truth label. Equation (2) is formulated as a regression problem, where $\hat{y}_i^{box}$ regression target obtained from the network and $y_i^{box} \in \mathbb{R}^4$ is the ground-truth coordinate. There are four coordinates, including left top, height and width. Similar to the face detection task, facial emotions classification is formulated as a seven-classification problem, we also use the cross-entropy loss:

$$L_i^{\text{emotion}} = -(y_i^{emotion}\log(p_i) + (1-y_i^{emotion})(1-\log(p_i))) \quad (3)$$

where $y_i^{\text{emotion}} \in \{0, 1, 2, 3, 4, 5, 6\}$ denotes the ground-truth label. p_i is the probability produced by the network that indicates a sample being one emotion of the facial expressions. label. p_i is the probability produced by the network that indicates a sample being one emotion of the facial expressions. Equation (3) is the Multi-source formulate.

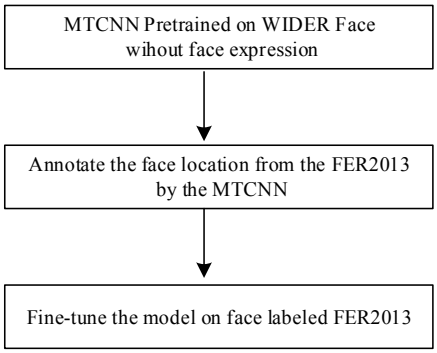

Figure 1. Flowchart of the training pocedure

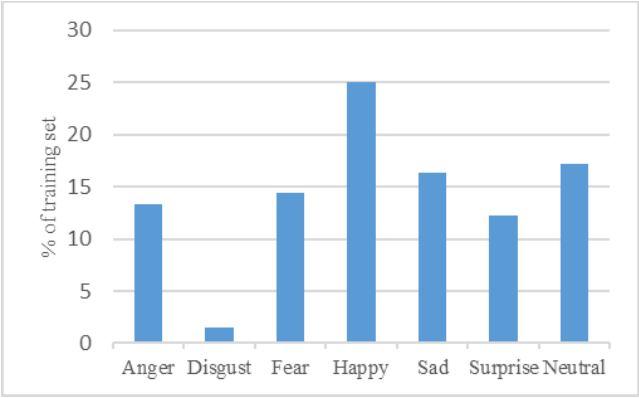

Figure 2. Distribution of the emotions

$$\min \sum_{i=1}^{N} \sum_{j \in \{det, box, emotion\}} \alpha_i \beta_j L_i^j \quad (4)$$

where N is the number of traing smaples. α_i denotes on the task importance.

Training of the MTCNN can be done is an manner using stochastic gradient descent (SGD) for both classification and regression branches. We train the P-Net with the collected datasets firstly, and train the R-Net with the predicted results of P-Net secondly. Lastly, for training O-Net, the datasets is collected from predicted results of both P-Net and R-Net.

C. Training

Our method follows the similar deep learning framework of MTCNN, It is similar to R-Net, called O-Net. In this network, we aim to classify the emotion of facial expressions. According we propose to modify the MTCNN architecture for facial expression recognition and train our facial expression recognition model by following the proposed procedure as shown in Fig. 1.

IV. Experiments

In this section, we train and test the networks was done using the Caffe open source framework for Deep Convolutional Neural Networks, and report experiments on comparisons of facial expression recognition and also on performance of top face detectors.

A. Training Dataset

The data we use is comprised of 48×48 pixel grayscale images of faces from the Kaggle competition Challenges in Representation Learning: Facial Expression Recognition Challenge[14]. It is a large, publicly available FER dataset consisting of 35,887 face crops that are almost centered and each face occupies about the same amount of space in each image. The dataset is split into training, validation, and test sets with 28,709, 3,589, and 3,589 images, labels of seven categories of facial expressions: Anger, Disgust, Fear, Happy, Sad, Surprise, Neutral. The human accuracy on this dataset is around 65.5%. The class distribution of the dataset can be found in Fig. 2.

To compensate for the relatively small size of the dataset, we made use of the popular data augmentation technique that consists in flipping images horizontally. As emotions should intuitively not change based on whether or not facial expressions are mirrored, it seemed a sensible choice. The training data for network is described as follows:

1) Pre-trained: According to *Equation* (4), we use ($\alpha_{det} = 1$, $\alpha_{box} = 0.5$, $\alpha_{emotion} = 0$) in MTCNN for training the only face detection task.

2) Annotation: In order to make the FER2013 datasets labeled with the face annotation, we use the pre-trained MTCNN to finish it.

3) Fine-tune: Similar to the Pre-trained stage, we use ($\alpha_{det} = 1$, $\alpha_{box} = 0.5$, $\alpha_{emotion} = 1$) to impelement the trainging of the facial expression recognition task.

B. Parameters

To choose the parameters (regularization strength, learning rate and decay), we randomly sampled in log space and retained those that yielded the best validation accuracy. The parameters we ended up using the following:

- Batch size: 100.
- Learning rate: 0.0001.
- Learning rate decay: 0.85.
- Momentum: 0.9.
- Number of epochs: 40.

C. Hard Negative Mining

Hard negative mining has been shown as an effective strategy for boosting the performance of deep learning, especially for object detection tasks including face detection. The idea behind this method is that, hard negatives are the regions where the network has failed to make correct prediction. Thus, the hard negatives are fed into the network again as a reinforcement for improving our trained model. The resulting training process will then be able to improve our model towards fewer false positives and better classification performance.

In our approach, hard negatives were harvested from the pre-trained model from the P-Net of our training process. We then consider a region as hard negative if its intersection over union (IOU) over the ground truth region was less than 0.5. During the hard negative training process, we explicitly add those hard negatives into the ROIs for fine-tuning the model, and balance the ratio of foreground and background to be about 1:3.

D. Results

We evaluated results against FER2013's validation and test sets[14]. The final validation accuracy we obtained is 60.7%. A example plot of our accuracy evolution can be found in Fig. 3. And the loss history is shown in Fig. 4. As can be seen, the training accuracy increases while the test accuracy remains almost constant after 15 epochs, This means we are slightly overfitting our data. The confusion matrix on the validation set is shown in Table. 1. Disgust is the class where our network fares the worst, and Happy where it is more successful.

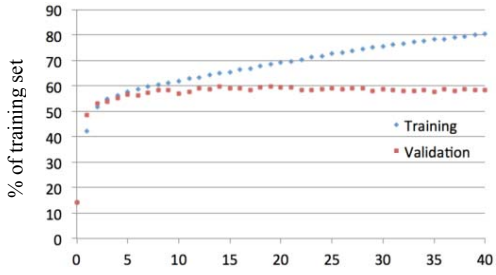

Figure 3. Training and validation accuracy over 40 epochs for a typical network we trained.

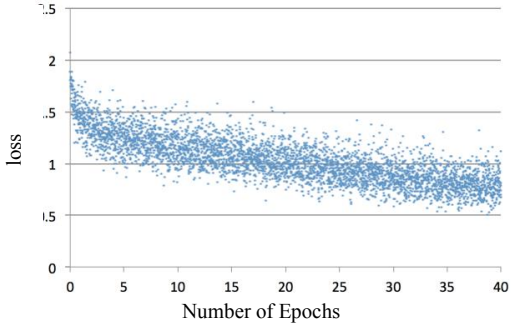

Figure 4. Loss History

TABLE I. VALIDATION SET CONFUSION MATRX

	Anger	Disgust	Fear	Happy	Sad	Surprise	Neutral
Anger	60	0	9.	5	14.	2	9.
Disgust	40.	21	10	6	19	0	4
Fear	14	0	33	7	28	7	12
Happy	5.	0	3	81	4	2	6
Sad	14	0	8	7	52	3	17
Surprise	3	0	9	6	4	74	5
Neutral	11.	0	7	8	19	0	54

Some of the difficulties with improving this is that the images are very small and in some cases it is very hard to distinguish which emotion is on each image, even for humans. To understand how the neural net classified different images we used saliency maps, to detect important regions in the images according to the neural net. Even though most results where quite noisy, some images showed convincing results.

V. CONCLUSION

In this work, we proposed a new method for face detection and facial expression recognition using deep learning techniques. Specifically, we inherent correlation them. We conducted an extensive set of experiments on the well-known FER2013 tested for FER work. Even though our validation accuracy is low, we believe that adding more layers and more filters would further improve the network. We will further improve the accuracy, and address the efficiency of the proposed method for other devices.

REFERENCES

[1] P. Ekman and W. V. Friesen. Emotional facial action codingsystem. Unpublished manuscript. University of California at San Francisco, 1983.

[2] Pramerdorfer, Christopher, and M. Kampel. "Facial Expression Recognition using Convolutional Neural Networks: State of the Art." (2016).

[3] D.Chen, S.Ren, Y.Wei, X.Cao, and J.Sun, "Joint cascade face detection and alignment", ECCV, 2014, pp.109-122.

[4] Zhang, Kaipeng, et al. "Joint Face Detection and Alignment Using Multitask Cascaded Convolutional Networks." IEEE Signal Processing Letters 23.99(2016):1499-1503.

[5] Viola, Paul, and M. Jones. "Robust object detection using a boosted cascade of simple features." Proc Cvpr 1(2001):I-511-I-518 vol.1.

[6] Xiangxin Zhu and Deva Ramanan. "Face detection, pose estimation, and landmark localizationin the wild". CVPR, 2012 IEEE Conference on, pages 2879–2886. IEEE, 2012.

[7] Li H, Lin Z, Shen X, et al. "A convolutional neural network cascade for face detection", CVPR, 2015: 5325-5334.

[8] Huaizu Jiang and Erik Learned-Miller. "Face detection with the faster r-cnn". arXiv preprint arXiv:1606.03473, 2016.

[9] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. In Advances in neural information processing systems, pages 91–99, 2015.

[10] Dong Chen, Shaoqing Ren, Yichen Wei, Xudong Cao, and Jian Sun. Joint cascade face detection and alignment. In European Conference on Computer Vision, pages 109–122. Springer, 2014.

[11] Matthew N Dailey, Garrison W Cottrell, Curtis Padgett, and Ralph Adolphs. Empath: A neural network that categorizes facial expressions. Journal of cognitive neuroscience, 14(8):1158–1173, 2002. 2.

[12] Z. Yu and C. Zhang, "Image based static facial expression recognition with multiple deep network learning," in ACM International Conference on Multimodal Interaction (MMI), 2015, pp. 435–442.

[13] Z. Zhang, P. Luo, C.-C. Loy, and X. Tang,"Learning Social Relation Traits from Face Images," in Proc. IEEE Int. Conference on Computer Vision (ICCV), 2015, pp. 3631–3639.

[14] Goodfellow, I. J., et al. "Challenges in representation learning: a report on three machine learning contests. " Neural Networks 64(2015):59-63.