Image and Vision Computing 47 (2016) 27–35

Contents lists available at ScienceDirect

Image and Vision Computing

journal homepage: www.elsevier.com/locate/imavis

Approaching human level facial landmark localization by deep learning☆

Haoqiang Fan *, Erjin Zhou

Megvii Inc., Beijing, China

ARTICLE INFO

Article history:
Received 3 March 2015
Received in revised form 12 September 2015
Accepted 30 November 2015
Available online 12 December 2015

Keywords:
Facial landmark localization
Deep learning
Convolutional neural network

ABSTRACT

In this paper we present our solution to the 300 Faces in the Wild Facial Landmark Localization Challenge. We demonstrate how to achieve very competitive localization performance with a simple deep learning based system. Human study is conducted to show that the accuracy of our system has been very close to human performance. We discuss how this finding would affect our future direction to improve our system.

1. Introduction

Reliable face recognition crucially depends on accurate and robust face alignment. Good alignment enables the face recognizer to be robust against pose and expression change [1,2]. Facial landmark localization seeks to detect a set of predefined key points on a human face. It attracts intense interest from both the industry and the research community.

Despite rapid progress in this area, facial landmark localization in uncontrolled settings remains an unsolved problem. The major challenge comes from the large variations in the facial images (see Fig. 1 for example of images used in our experiments). The head can have large yaw or pitch angles. The lightening condition can be extreme. In addition, some of the images are of low quality. The resolution can be low. The image can be blurry or corrupted. When viewed locally, some of the facial landmarks are difficult to be recognized. Even humans have to use the global context to identify a point.

The major requirement of face alignment is robustness. The landmark localizer must return a set of reasonable point coordinates however the image is corrupted. Even if a point is not visible, it is typical for face recognizers to demand that the landmark localizer should "guess" a position for the point so that some global measurements (e.g. the rotation of the face) can be made. This poses great challenge to the landmark localizer.

The most straightforward way to solve the landmark localization problem is to view it as an image-based regression problem. The input is the RGB image. The outputs are the coordinate values. Any image-based regressor can be plugged in. This framework is capable of giving very promising result if the regressor is powerful enough. In our solution we use a CNN (Convolutional Neural Network) Cascade. There are two key ingredients in our solution:

- Deep network. To increase robustness, we used significantly deeper networks compared to previous works [3,4].
- Coarse-to-fine prediction. Using multiple CNNs in a coarse-to-fine manner improves accuracy.

However, obtaining good quality training data is difficult and expensive. We conduct human study to investigate human's ability to locate key points in an image. Then we discuss how our findings would influence our future direction to improve the system.

1.1. Related work

Face alignment is an indispensable step in modern face recognition system [5,6,7,and 8]. Generally, there are two schools of methods for facial landmark detection: model-based methods and regression-based methods. Model-based methods try to build models to fit input images. They can take into account the local texture appearance [9] or the part-based structure [10,11,12]. Their advantage is that human's prior knowledge can be easily incorporated into the system.

The regression-based methods are more straightforward. Deep neural networks [4,3,13,14] and boosted regressors [15] have been successfully employed. The regression-based method solely depends on the regressor's capacity to learn the extremely complex relation between pixel values and the appearance of facial features. Great care is needed in training these complex models. In our solution we aim at keeping the conceptual framework of the method as simple as possible. The major difference in our method is that our neural network's depths greatly exceed those used by existing works [4,3,13] which typically only has at most 4 convolutional layers (ours contains 8 convolutional

☆ This paper has been recommended for acceptance by Stefanos Zafeiriou.

* Corresponding author. Tel.: +86 13671270149.
E-mail addresses: fhq@megvii.com (H. Fan), zej@megvii.com (E. Zhou).

http://dx.doi.org/10.1016/j.imavis.2015.11.004
0262-8856/

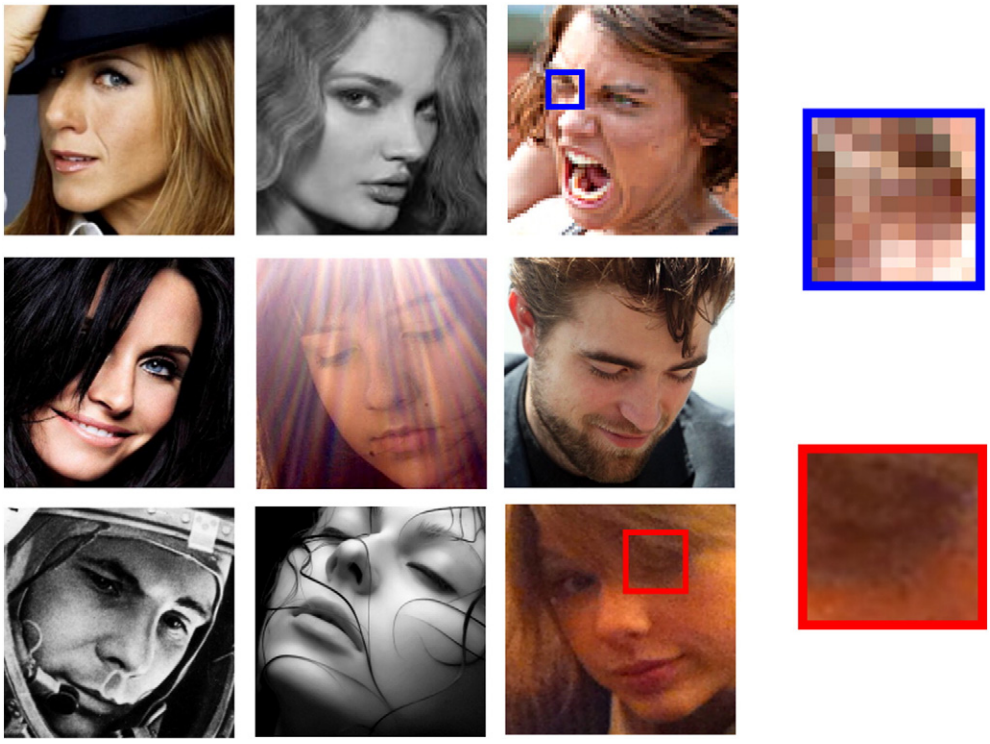

Fig. 1. Examples of pictures in the IBUG dataset which is used as the validation set in our experiments. This dataset contains very difficult pictures. On the right are two zoom-in view of two selected image regions. From the local patches we can hardly recognize the facial landmarks.

layers). Due to the extra power of the network, we can make the whole framework largely simplified.

2. Deep CNN cascaded for facial landmark localization

We formulate the landmark localization problem as learning a function that maps image pixel arrays to point coordinates. The input image $I_{h\times w}$ is a $h\times w$ three channel (RGB) image. It contains the face area found by the face detector. The output $P_{n\times 2}$ is landmark coordinates relative to the face's bounding box.

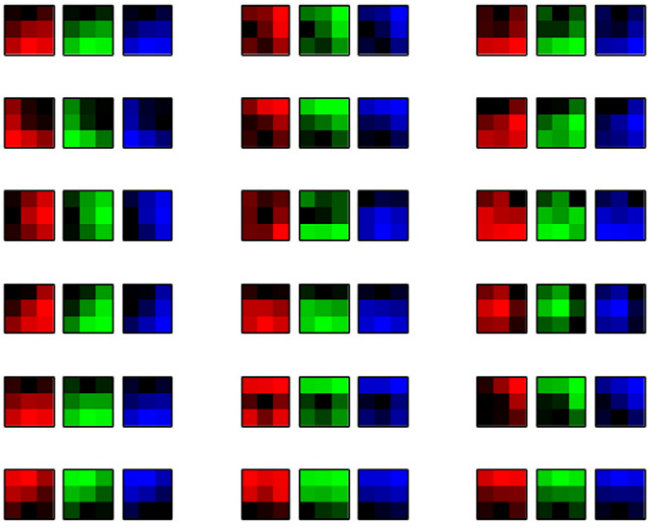

Fig. 2. Example of learned first layer filters. Each filter is a three channel weight map. Some of the filters can be recognized as line or corner detectors. The functions of others are not easily explainable.

In the following two subsections we present detailed descriptions of the two components of our framework: Convolutional Neural Network and coarse-to-find prediction.

2.1. Convolutional neural network

Mathematically, a Convolutional Neural Network is the composition of a series of non-linear function maps that operate on 3 dimensional arrays (height × width × # channels). We denote the input image as $I^0_{i,j,k}$ which is a three channel colored image. There are three types of functions: convolution, pooling and non-linear activation.

The convolution operation slides multi-channel "filters" on the image and computes the dot product of the filter's weights and the image pixels (See Fig. 2). In one network layer, many filters are applied to the image and their outputs are stacked to obtain the next layer's input channels. Mathematically, the function is defined as follows:

$$\mathrm{conv}_{W,B}\left(I^t\right)_{i,j,k} = \sum_{x=0}^{p\text{-}1}\sum_{y=0}^{q\text{-}1}\sum_{u=0}^{c\text{-}1} \mathrm{I}^t_{i\text{-}x,j\text{-}y,u} W^t_{k,u,x,y} + B^t_k.$$

The filter's size is $p\times q$ and the input image contains c channels. The four-dimensional array W stores the filter's weights, and B is a "bias" term which additively shifts the output for each output channel. The number of output channels does not need to match the input channels. We can think of the filters as pattern detectors that selectively response to certain input configurations. Fig. 1 visualizes the learned first layer filters in a network. Some of them are edge or corner detectors while the function of others is not easily explainable.

One special case of "convolution" is when the input image and output image are of size 1×1. In this case the operation reduces a matrix–vector multiplication. We call this kind of layers "fully connected" layers because each input value contributes to all output values.

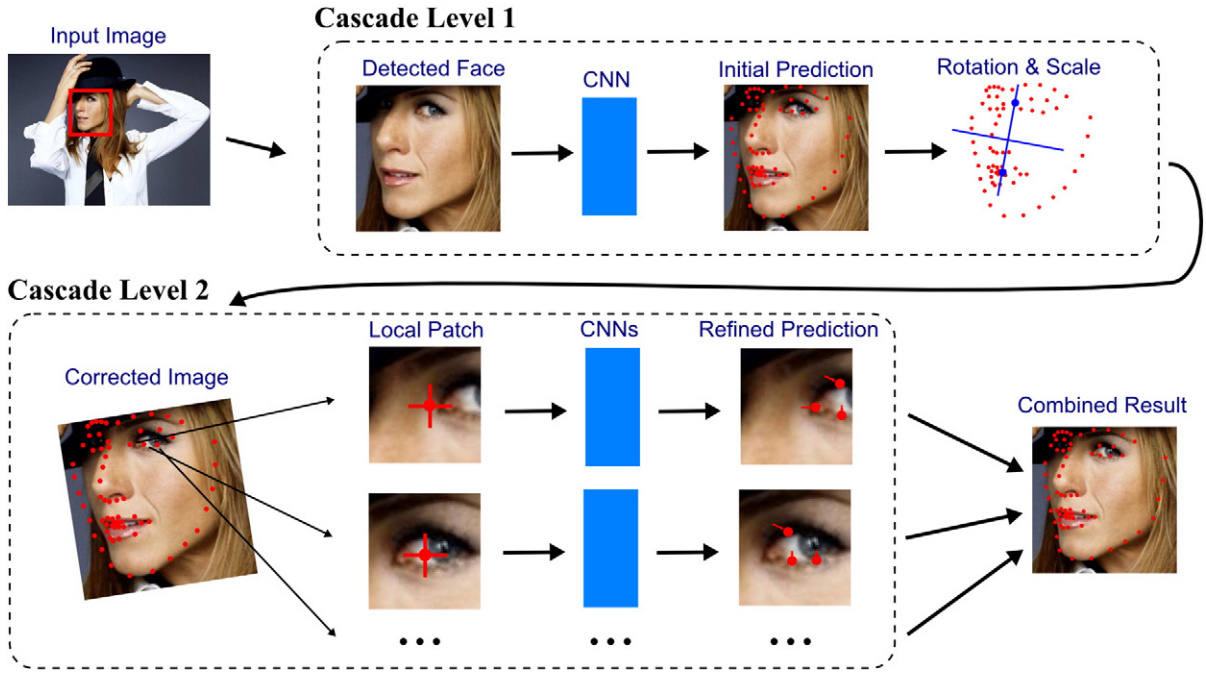

Fig. 3. The CNN Cascade. Two levels of CNNs work in a coarse-to-fine manner. The first level CNN provides a rough initial prediction. From the initial prediction the rotation and scale of the face are found. Then the image is rectified and sent to the second level. The second level CNNs work on image patches centered at the predicted landmarks. Each of them refines the predictions of points in the input patch. $K=3$ in this figure. The output from the second level networks is voted to produce the final result.

The pooling operation reduces the image's size. It divides the image into $b\times b$ blocks and selects the maximum value from each block. Each channel is considered independently. The mathematical expression is

$$\mathrm{pool}_b\left(I^t\right)_{i,j,k} = \max_{0\le x<b, 0\le y<b} I^t_{ib+x,jb+y,k}.$$

The function of the pooling operation is to acquire invariance against image deformation. When the pattern is translated, the pooled value remains the same so long as the maximum stays in the same block. This makes the network robust against slight local translation or rotation. However, spatial information is dropped by the pooling operation.

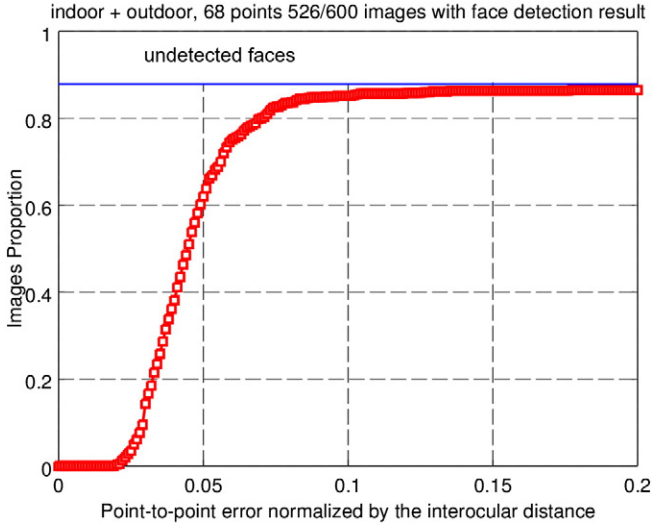

Fig. 4. Result in the 300 W challenge. We are among the best results. From the figure we can see our performance is largely affected by the poor face detector which misses 12% of all faces. Notice that in 300 W the interocular distance is computed from the distance between the two outter eye corners.

Though in landmark localization the spatial information is important, we still find that using max-pooling is beneficial.

The third type of operation is non-linear activation. It applies a non-linear function to each value in the image. This is a purely element-wise operation:

$$s\left(I^t\right)_{i,j,k} = g\left(I^t_{i,j,k}\right)$$

where g is some non-linear function. In our experiment we use the hyper-tangent function though other functions are also possible.

We consider the output as a 1×1 "image" with $2n$ channels, where n is the number of landmarks. The input data goes through a series of transformations

$$I^1 = f_0\left(I^0\right)$$

$$I^2 = f_1\left(I^1\right)$$

$$\ldots$$

$$I^t = f_{t-1}\left(I^{t-1}\right)$$

Each function f_i is either conv_{W_i,B_i}, pool_b or s. I^0 is the RGB input image and I^t is the output landmark coordinates. Conventionally, only convolution operations count as "layers" so when we call a network a ten-layer CNN, we mean ten W and B pairs are included in this network.

The network is trained by Empirical Risk Minimization. Input and output pairs are collected. A loss function is defined to measure the difference between the network's output and the ground truth. We use the L2 loss function which sums the squared differences:

$$\mathcal{L} = \sum_{i=0}^{N-1} \|Y_i - f(X_i)\|_2^2.$$

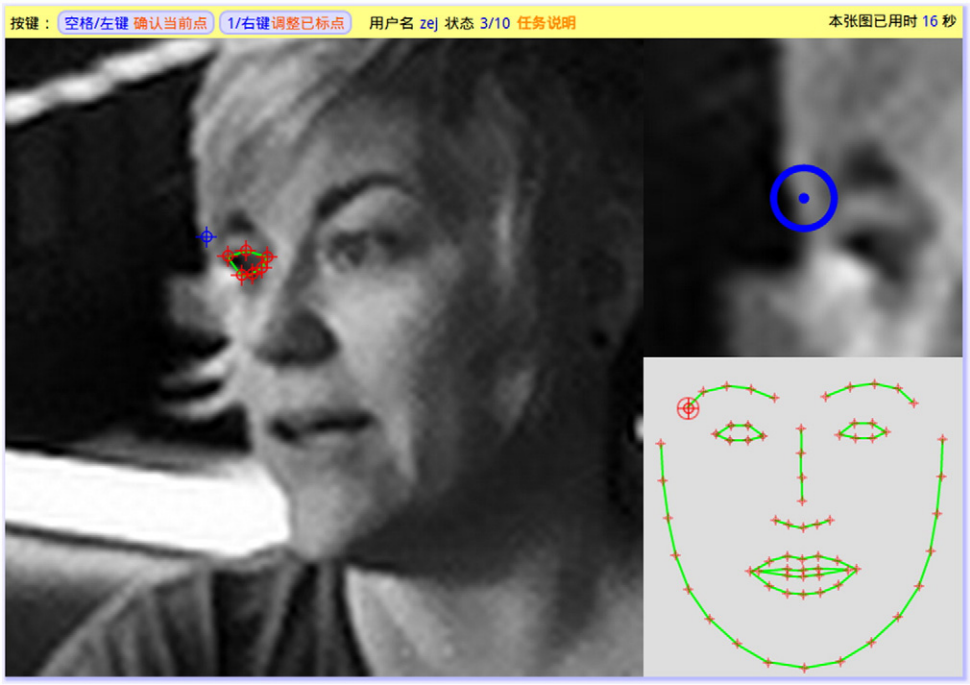

Fig. 5. The GUI tool used in our human study. The subject can use the mouse to choose the point's position and press the keyboard to advance to the next point. They can also revise the point's positions by switching to "edit mode". The instructions are written in Chinese as most of our subjects are Chinese speaking.

f represents the calculations in the CNN. X and Y are data pairs. This objective function is optimized by Stochastic Gradient Descent which randomly samples a batch of data pairs and adjusts the network's parameters according to the gradient estimated on this batch.

2.2. Coarse-to-fine prediction

In our system multiple CNNs are used to form a CNN cascade. The overview of the system is shown in Fig. 3. The neural networks work in a coarse-to-fine manner. Two levels are used in our cascade. The first level aims at capturing the global structure of the face and gives a rough but robust initial estimate of the landmark positions. The following level refines the prediction by taking a closer look at each local area surrounding the landmark points. The two levels are described in detail in the following subsections.

2.2.1. First level CNN

The first level network takes the whole face image as input. Before feeding the image to the network, the face is detected and cropped. In the output layer the network generates the predicted point coordinates relative to its input region.

After obtaining the initial predictions we perform a global similarity transform to turn the face to a canonical direction. We calculate the average of the landmark coordinates that correspond to eyes and mouth. From the center of the eyes and mouth we estimate

Fig. 6. Example of our randomly generated curves. The true equal division points of the first curve are rendered.

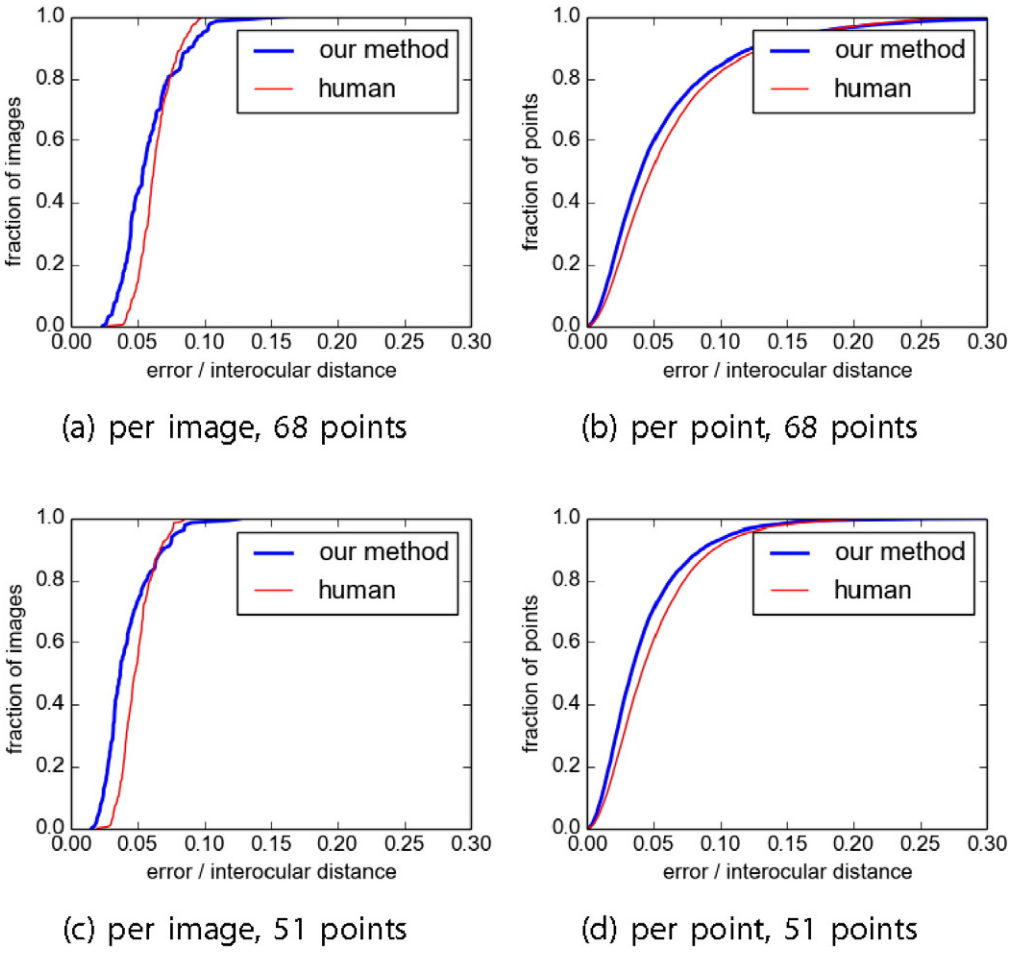

Fig. 7. Comparison between human and our neural networks in the IBUG dataset. CED curves are plotted for (a) average distance per image, all 68 points; (b) distance per point, all 68 points; (c) average distance per image, 51 inner points only; and (d) distance per point, 51 inner points only. The red solid line is human. The blue thick line is our neural network cascade. From the figures we can see that the neural network's performance is very close to human.

the scale and in-plane rotation angle of the face. The inverse transformation is applied to the image so that the face is upright and properly scaled.

Because localizing landmarks is a very difficult task, we do not expect that the landmarks returned from the first level network are satisfactory. Instead, we only need the predictions to be robust enough. Because the whole face area is given in the input, the first level network is capable of giving reasonable prediction even when the face is partially occluded. The position of invisible points can be inferred from the face's global structure.

We rotate and scale the image to rectify the second level network's input. This reduces the second level networks' burden of handling

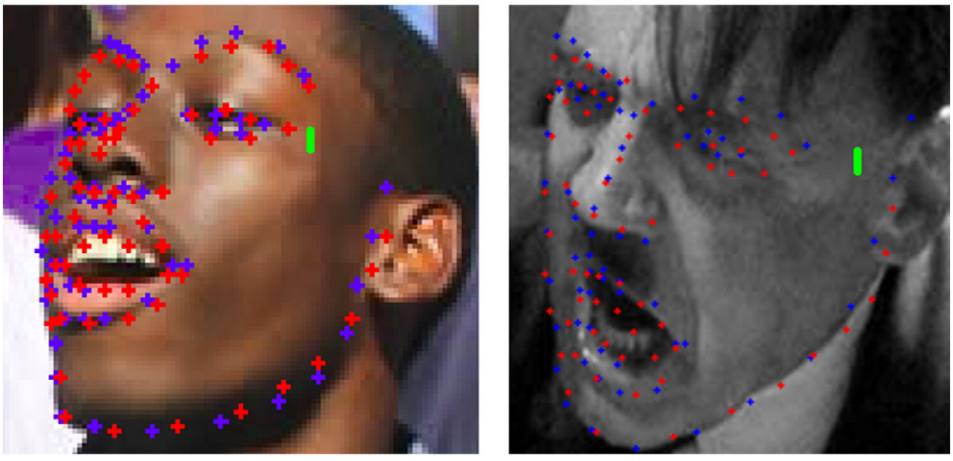

Fig. 8. Example of human's errors. The red dots are produced by our human subjects. The blue ones are IBUG's "ground truth". The green line indicates the length of 0.1 interocular distance. As can be seen from the figure, due to the image's quality and the inherent ambiguity of the landmarks, it is sometimes really difficult to locate landmarks like the eyebrow and contour.

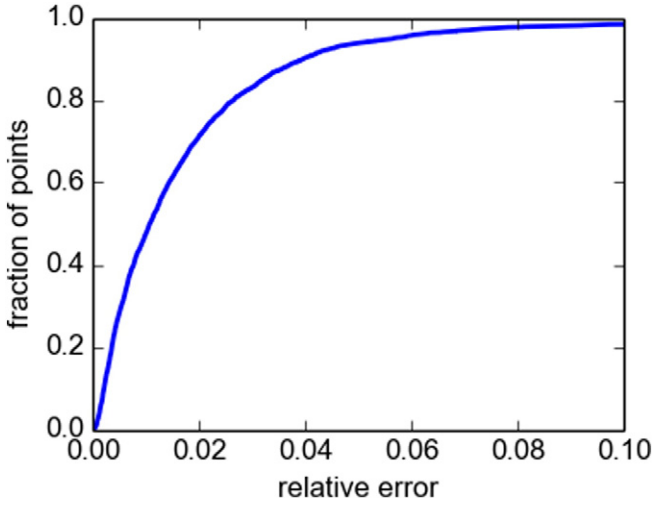

Fig. 9. Human performance to find equal division points. The errors are normalized by the length of the curve. Recall that the length of the face's contour line is greater than 3 interocular distance and the average eyebrow length is about 0.6 interodular distance. So this amount of error is highly non-negligible.

rotation and scale changes so that they can use their capacity to learn more detailed and local information of the image.

2.2.2. Second level CNNs

The task of finding the exact locations of the landmark points is left to the second level networks. Totally n CNNs are used in the second level where n is the number of landmarks. The input to the ith CNN is an image patch centered at the ith landmark's position predicted by the first level network. The CNN is asked to predict the positions of K landmarks that are closest to the ith landmark in the mean face. So totally nK positions are computed by the second level networks. Each landmark position is predicted by at least one second level network while some landmarks can be predicted by multiple networks. We average the predictions associated with the same landmark to obtain the final output.

The scope of the input of each second level network is much smaller than the first level network. In this way it is given a chance to look at the local detail of the image. However, the small input scope makes the second level network vulnerable to image corruption. So we combine the predictions from multiple second level networks to improve robustness. The value of K is chosen so that the second level networks can see all of the K points in their input regions. Due to time limit, when we submit our solution to 300 W challenge we only use $K = 1$. In the validation dataset we find using $K = 3$ improves the performance. Further experimental results are shown in Section 3.1.

2.3. Implementation details

We used an off-the-shelf CNN training tool called CUDA-Convnet [16]. It is capable of using the GPU to accelerate the training procedure.

One issue in training the second level networks is that their input depends on the first level network's prediction. However, if we use the same training set for both levels, the second level network will see an overly optimistic input distribution because the first level has fitted to the training set before providing its prediction to the second level networks. This will make the second level networks tend to make smaller changes to the first level's prediction. We resolve this issue by using a bagging-like training strategy. The training set is divided into 4 fold. We left out each fold in turn and train the first level network on the other three fold. Totally 4 first level networks are trained. When the second level networks are trained, the first level prediction comes from the first level network that does not see the corresponding data fold. In this way we mimic the error distribution of the first level network that is seen by the second level during testing. On the test set we run all of the 4 first level networks and use the average of their predictions to generate the initial estimates.

Table 1 summarizes the network architecture used in our system. Because the first level network has a greater input region, we assign it with more layers. Totally 10 layers are used in this network which is significantly more than what we used in our last year's solution [3] (which uses only 4 convolutional layers).

3. Experiment

We conducted three experiments. The first experiment evaluates the number of points seen for each of networks in the second level.

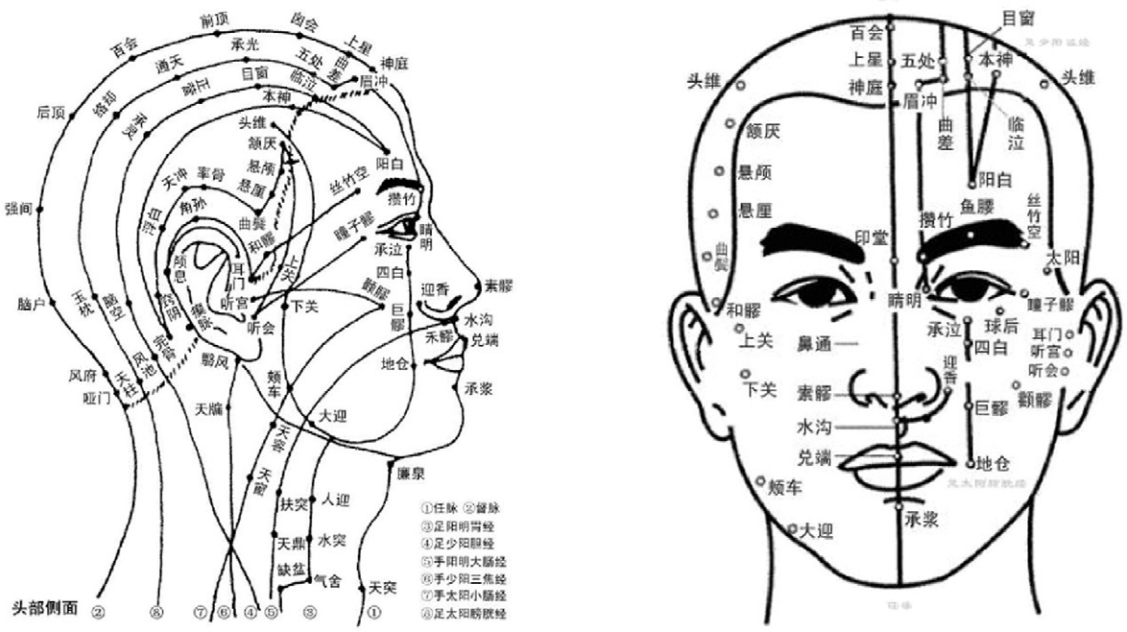

Fig. 10. Traditional Chinese medicine practitioners have long been defining physiologically important points on human skin. These points are often the targets of acupuncture, which must be located with great precision. However these points are not suitable for landmark localization due to lack of visual saliency.

Table 1
Network architecture used in our experiment. Conv $a \times b \times c$ denotes a convolution layer with c filters of sizes a × b. Pool b stands for a pooling layer with grid $b \times b$. Full c represents a fully connected layer with c output channels. We omit the non-linear activations which are inserted after each Conv and Full (except the last fully connected layer, which is the output layer). n and K are the number of output landmarks in the two levels. In our experiment $n = 68$ and K is set to 1 or 3.

(a) level 1	(b) level 2
input 128 × 128 × 3	input 24 × 24 × 3
conv 3 × 3 × 32	conv 3 × 3 × 32
pool 2	pool 2
conv 3 × 3 × 64	conv 3 × 3 × 64
conv 3 × 3 × 64	conv 3 × 3 × 64
pool 2	pool 2
conv 3 × 3 × 64	conv 3 × 3 × 64
conv 3 × 3 × 64	conv 3 × 3 × 128
pool 2	full 256
conv 3 × 3 × 128	full 2K
conv 3 × 3 × 128	
pool 2	
conv 3 × 3 × 256	
full 1024	
full 2n	

The second experiment validates the effectiveness of our method. In the last experiment we study human's ability to locate points on images.

Our training set consists of the datasets provided by 300 W [17,18,19,20] and our own manually labeled web images. The IBUG dataset as well as "test" partition of HELEN and LFPW is left out for testing. The point-to-point Euclidean distance normalized by interocular distance is used to as the error measurement.

3.1. Choosing the number of points for the second level networks

We evaluate the network performance according to different numbers of points predicted by the each of networks in the second level. We choose $K = 1, 3, 5$ and Table 2 shows the average error results on the IBUG dataset. The best performance is achieved when $K = 3$. We argue that this is because of an inevitable trade-off between the image scope and image details existed in a fixed size of network input. When $K = 1$, the network loses the structural relationship between the predicted point and the points around it. When K is large, in order to cover all the predicted points in the network's input, image details are lost due to enlarged input scope. Therefore, we use $K = 3$ in the following experiments.

3.2. Evaluating the effectiveness of method

3.2.1. Evaluation on 300-W benchmark

In this section we compare our methods with recently published results [21,11,22,23,14,15,24,25,26,27,13]. The evaluation dataset is divided into common subset (test set of LFPW, HELEN) and challenge subset (135 images collected by IBUG) [26,13] are cited from Zhang's work). Our method reduces the error significantly compared with previous methods (See Table 3).

Our implementation takes 0.5 s to process one image on a single CPU core (Intel Core i7-4870HQ @ 2.50 GHz).

Table 2
Average normalized distance on the IBUG dataset with different numbers of points predicted in each of the second level networks.

	$K = 1$	$K = 3$	$K = 5$
IBUG	5.95	5.72	6.45

Table 3
Average normalized distance on the 300-W benchmark. Numbers for some of the methods are taken from [13].

Method	Common subset	Challenge subset	Full set
CDM [21]	10.10	19.54	11.94
DRMF [22]	6.65	19.79	9.22
RCPR [23]	6.18	17.26	8.35
GN-DPM [11]	5.78	–	–
CFAN [14]	5.50	16.78	7.69
ESR [15]	5.28	17.00	7.58
SDM [24]	5.57	15.40	7.50
ERT [25]	–	–	6.40
LBF [26]	4.95	11.98	6.32
CFSS [27]	4.73	9.98	5.76
TCDCN [13]	4.80	8.60	5.54
Our Method	4.76	8.25	5.45

3.2.2. Evaluation on 300-W challenge results

The 300 Faces in the Wild Challenge (300 W) is a facial landmark localization benchmark. A total of 600 pictures including 300 indoor pictures and 300 outdoor pictures are provided. The participants need to submit runnable binaries to the evaluation server. They are asked to include the face detector in the program. Due to commercial constraints, we are only able to submit a Free Trial version of Face++'s face detector, which is Viola-Jones [28] based in our system.

The cumulative error curve is plotted in Fig. 4.

Notice that a large portion of our error can be attributed to the poor face detector included in our binary. We believe the result will be significantly improved if we are able to use a more competent face detector.

3.3. Human study

In this study[1] we aim at investigating human's ability to find point locations on image. Because in the framework human labels are our only source of supervising signal, it is important to understand whether our dependence on human has become the bottleneck.

We first study the level of disagreement among human labelers about the landmark points' positions. To do this we recruit 76 Asian subjects (25 male, 53 female, age from 17 to 34) to label the IBUG dataset which contains 135 images. This image dataset is used as the validation set in our learning procedure. The subjects are instructed to use a web based GUI tool (see Fig. 5) to label the landmarks. We put no time limit on the human subjects, but we manually reject all labeling results that are done faster than 3 min per image. The remaining results have a median labeling time of 4.4 min per image. We also remove apparently erroneous results caused by labeler's misunderstanding of how the GUI work (e.g. they label the points in a wrong order). Each subject labels an independently randomly selected subset of all images. In total 443 labeled images are collected, which means on average each image in the dataset is labeled by 3.3 human subjects.

Before labeling the human subject is provided with a detailed description of the definitions of the landmark points. After labeling an image the subjects receive the image's "ground truth" labels as feedback. They are instructed to try to follow IBUG's labeling style.

In addition to labeling landmark, we also ask the subjects to perform another task of labeling equal division points along a curve. Because the definition of many landmarks depends on equal division (e.g. contour and eyebrow), we want to study how this way of definition affects human's ability of labeling. In the study a curve is presented to the subject, and the subject is asked to equally divide the points into 8 segments (Fig. 6). They are asked to use the GUI interface only, without the help of external tools [29]. We record the Euclidean distance between the labeled point and the true equal division points.

[1] Note that a larger scale experiment was conducted after the first submission of the paper. The figures here reflect the latest results.

As shown in Fig. 7, the "human performance" is very close, and even sometimes inferior, to our best machine landmark localizer. We want to point out that the "ground truth" in IBUG dataset is far from perfect. So we suggest that this result only implies that machine is comparable with human in terms of "fitting" to the ground truth labeler's favor.

Fig. 9 plots the human's error in locating equal division points. The errors are normalized by the length of the curve. The median is above 0.01 while the 90% quantile is around 0.04. This is a highly non-negligible amount of error. From the dataset we estimate that the average length of the eyebrow is 0.6 interocular distance and the length of the contour line is greater than 3 interocular distance. This can partially explain why the disagreements between human labels are so large. We suggest that the current equal division based definitions of the contour points and the point-by-point labeling tools are not the most suitable for human labelers to use. Using spline based annotations may be a better choice. However, this will complicate the design of the CNN's loss function.

Due to limited time and budget, our human study is of small scale and the constitution of participants is probably biased. But we believe our experiment is able to reveal the fact that humans are not always reliable in labeling facial landmarks. We list two factors that contribute to human's errors:

- Ambiguity. People can have very different interpretations of the terms like "eyebrow" (See Fig. 8). It is hard to enforce a uniform and consistent labeling rule that can handle all kinds of special cases.
- Human's ability. Human's eyes are not designed for making precise length measurements. They are also not good at imagining image transformations like where a point should be if the curve is straightened. The user interface must be carefully improved if we want human labelers to achieve higher precision.

4. Discussion and future direction

4.1. End of the free lunch

One of the "free lunches" in Deep Learning is that the system's performance almost improves monotonically when we increase the neural network's depth and size. This is only made possible by the availability of abundant training data and ever increasing computing power. The combination of Amazon Turk and Nvidia's high-end GPUs witness researchers' impressive progress in image classification and object detection [30,31], which is one of the most exciting stories in the current deep learning wave.

In many vision fields including image classification and face recognition deep learning methods have claimed "human level" performance [8,31]. Researchers argue that the "accuracy" obtained by their classifier and human are comparable. However, in order to measure the "accuracy" they need the "ground truth" which is nothing but the labels generated by another set of human. This problem is partially addressed by "voting" the human labels to find consensus. But conceptually, the truth is not always in the minds of the majority, especially for vision tasks like face recognition where there is a genetically or legally defined "truth". It is known that in the LFW face recognition benchmark there are erroneous labels and the error could only be found after tracing the source of the image.

As for landmark localization there are indeed ways to define unambiguous facial points. We can resort to medical or anatomical definitions (e.g. Fig. 10). However, it is difficult for untrained human to follow these definitions. Furthermore, they are very hard to use when the only available information is a single image of the face.

In principle the neural network is able to learn from noisy data labels. However, the label quality will still eventually pose obstacle to the system's performance. For the research community the "human performance" has been a good enough target, but for many real world applications (e.g. access control in sensitive area) the human performance is only a start. It will be necessary to find ways to go beyond human or even consensus of human.

4.2. Future direction of face alignment

As stated above, we cannot totally depend on human to provide "ground truth" landmark positions. One possible way to bypass this obstacle is to use unsupervised or semi-supervised learning. We can hope to learn person-specific landmarks through these learning procedures. The person-specific landmarks are facial points that can be consistently defined only for each person. The points must be in good correspondence on facial images of the same person while the requirement on the relation between corresponding landmarks on different people can be relaxed. These landmarks are still useful for face recognition while they can better capture the person's own characteristics. We describe two possible ways to do this.

4.2.1. 3D consistency

When we could collect pictures of the same person that contain little change in expression (e.g. shots from different cameras during one event), we can enforce 3D consistency restriction on the landmark positions. The change in coordinates should be explainable by the camera's projection matrix (focus, view point and direction). Because there are totally 136 degrees of freedom in the 68 point coordinates, which is much more than the number of free parameters in the fundamental matrix this will give an overly determined system of equations and hence a non-trivial constraint on the point's positions. In this way we can even hope to obtain the 3D coordinates of the landmark points which are even more useful in face analysis.

4.2.2. Temporal continuity

We can also learn landmark positions from face video. The coordinates should change smoothly with time. Also, the movement of the landmark points should be in agreement with the image motion computed from optical flow or other motion estimates. This can make the landmark localizer robust against illumination change. Also, it helps the system to model people's expressions or actions.

One common problem in unsupervised learning is "drifting" which means the learned definition of points slowly degrades to some trivial configuration (e.g. all 68 landmark collapses at one point). So we believe a certain amount of supervised signal is still necessary. Our hope is that a small scale labeled dataset could leverage large scale or even web scale unsupervised data.

5. Conclusion

In this paper we showed how to build an accurate and robust facial landmark localizer using deep learning tools. We point out the limitation of our current supervised learning framework and describe directions to further improve our system.

References

[1] T. Berg, P.N. Belhumeur, Tom-vs-Pete classifiers and identity-preserving alignment for face verification, BMVC, Citeseer, vol. 2 2012, p. 7.
[2] D. Chen, X. Cao, F. Wen, J. Sun, Blessing of dimensionality: high-dimensional feature and its efficient compression for face verification, Computer Vision and Pattern Recognition (CVPR), 2013 IEEE conference on, IEEE 2013, pp. 3025–3032.
[3] E. Zhou, H. Fan, Z. Cao, Y. Jiang, Q. Yin, Extensive facial landmark localization with coarse-to-fine convolutional network cascade, Computer Vision Workshops (ICCVW), 2013 IEEE international conference on, IEEE 2013, pp. 386–391.
[4] Y. Sun, X. Wang, X. Tang, Deep convolutional network cascade for facial point detection, Computer Vision and Pattern Recognition (CVPR), 2013 IEEE conference on, IEEE 2013, pp. 3476–3483.
[5] E. Zhou, Z. Cao, Q. Yin, Naive-deep face recognition: touching the limit of lfw benchmark or not?, arXiv preprint arXiv:1501.04690.
[6] H. Fan, M. Yang, Z. Cao, Y. Jiang, Q. Yin, Learning compact face representation: packing a face into an int32, Proceedings of the ACM International Conference on Multimedia, ACM 2014, pp. 933–936.

[7] Y. Sun, X. Wang, X. Tang, Deeply learned face representations are sparse, selective, and robust, arXiv preprint arXiv:1412.1265.
[8] Y. Taigman, M. Yang, M. Ranzato, L. Wolf, Deepface: closing the gap to human-level performance in face verification, Computer Vision and Pattern Recognition (CVPR), 2014 IEEE conference on, IEEE 2014, pp. 1701–1708.
[9] T.F. Cootes, G.J. Edwards, C.J. Taylor, Active appearance models, Computer Vision—ECCV'98, Springer 1998, pp. 484–498.
[10] P.F. Felzenszwalb, R.B. Girshick, D. McAllester, D. Ramanan, Object detection with discriminatively trained part-based models, IEEE Trans. Pattern Anal. Mach. Intell. 32 (9) (2010) 1627–1645.
[11] G. Tzimiropoulos, M. Pantic, Gauss–Newton deformable part models for face alignment in-the-wild, Computer Vision and Pattern Recognition (CVPR), 2014 IEEE conference on, IEEE 2014, pp. 1851–1858.
[12] P.F. Felzenszwalb, D.P. Huttenlocher, Pictorial structures for object recognition, Int. J. Comput. Vis. 61 (1) (2005) 55–79.
[13] Z. Zhang, P. Luo, C. Loy, X. Tang, Learning deep representation for face alignment with auxiliary attributes, IEEE Trans. Pattern Anal. Mach. Intell. PP (99) (2015) 1.
[14] J. Zhang, S. Shan, M. Kan, X. Chen, Coarse-to-fine auto-encoder networks (CFAN) for real-time face alignment, Computer Vision—ECCV 2014, Springer 2014, pp. 1–16.
[15] X. Cao, Y. Wei, F. Wen, J. Sun, Face alignment by explicit shape regression, Int. J. Comput. Vis. 107 (2) (2014) 177–190.
[16] A. Krizhevsky, I. Sutskever, G.E. Hinton, Imagenet classification with deep convolutional neural networks, Advances in Neural Information Processing Systems 2012, pp. 1097–1105.
[17] V. Le, J. Brandt, Z. Lin, L. Bourdev, T.S. Huang, Interactive facial feature localization, Computer Vision—ECCV 2012, Springer 2012, pp. 679–692.
[18] C. Sagonas, G. Tzimiropoulos, S. Zafeiriou, M. Pantic, A semi-automatic methodology for facial landmark annotation, Computer Vision and Pattern Recognition Workshops (CVPRW), 2013 IEEE conference on, IEEE 2013, pp. 896–903.
[19] P.N. Belhumeur, D.W. Jacobs, D. Kriegman, N. Kumar, Localizing parts of faces using a consensus of exemplars, Computer Vision and Pattern Recognition (CVPR), 2011 IEEE conference on, IEEE 2011, pp. 545–552.
[20] X. Zhu, D. Ramanan, Face detection, pose estimation, and landmark localization in the wild, Computer Vision and Pattern Recognition (CVPR), 2012 IEEE conference on, IEEE 2012, pp. 2879–2886.
[21] X. Yu, J. Huang, S. Zhang, W. Yan, D.N. Metaxas, Pose-free facial landmark fitting via optimized part mixtures and cascaded deformable shape model, Computer Vision (ICCV), 2013 IEEE international conference on, IEEE 2013, pp. 1944–1951.
[22] A. Asthana, S. Zafeiriou, S. Cheng, M. Pantic, Robust discriminative response map fitting with constrained local models, Computer Vision and Pattern Recognition (CVPR), 2013 IEEE conference on, IEEE 2013, pp. 3444–3451.
[23] X.P. Burgos-Artizzu, P. Perona, P. Dollár, Robust face landmark estimation under occlusion, Computer Vision (ICCV), 2013 IEEE international conference on, IEEE 2013, pp. 1513–1520.
[24] X. Xiong, F. De la Torre, Supervised descent method and its applications to face alignment, Computer Vision and Pattern Recognition (CVPR), 2013 IEEE conference on, IEEE 2013, pp. 532–539.
[25] V. Kazemi, J. Sullivan, One millisecond face alignment with an ensemble of regression trees, Computer Vision and Pattern Recognition (CVPR), 2014 IEEE conference on, IEEE 2014, pp. 1867–1874.
[26] S. Ren, X. Cao, Y. Wei, J. Sun, Face alignment at 3000 fps via regressing local binary features, Computer Vision and Pattern Recognition (CVPR), 2014 IEEE conference on, IEEE 2014, pp. 1685–1692.
[27] S. Zhu, C. Li, C.C. Loy, X. Tang, Face alignment by coarse-to-fine shape searching, Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition 2015, pp. 4998–5006.
[28] P. Viola, M.J. Jones, Robust real-time face detection, Int. J. Comput. Vis. 57 (2) (2004) 137–154.
[29] A. Chandra, Dividing a circular arc into equal number of divisions.
[30] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, L. Fei-Fei, Imagenet: A large-scale hierarchical image database, Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE conference on, IEEE 2009, pp. 248–255.
[31] K. He, X. Zhang, S. Ren, J. Sun, Delving deep into rectifiers: surpassing human-level performance on imagenet classification, arXiv preprint arXiv:1502.01852.