2013 IEEE International Conference on Computer Vision Workshops

Extensive Facial Landmark Localization with Coarse-to-fine Convolutional Network Cascade

Erjin Zhou Haoqiang Fan Zhimin Cao Yuning Jiang Qi Yin
Megvii Inc.
{zej,fhq,czm,jyn,yq}@megvii.com

Abstract

We present a new approach to localize extensive facial landmarks with a coarse-to-fine convolutional network cascade. Deep convolutional neural networks (DCNN) have been successfully utilized in facial landmark localization for two-fold advantages: 1) geometric constraints among facial points are implicitly utilized; 2) huge amount of training data can be leveraged. However, in the task of extensive facial landmark localization, a large number of facial landmarks (more than 50 points) are required to be located in a unified system, which poses great difficulty in the structure design and training process of traditional convolutional networks. In this paper, we design a four-level convolutional network cascade, which tackles the problem in a coarse-to-fine manner. In our system, each network level is trained to locally refine a subset of facial landmarks generated by previous network levels. In addition, each level predicts explicit geometric constraints (the position and rotation angles of a specific facial component) to rectify the inputs of the current network level. The combination of coarse-to-fine cascade and geometric refinement enables our system to locate extensive facial landmarks (68 points) accurately in the 300-W facial landmark localization challenge.

1. Introduction

Facial landmark localization plays a critical role in the systems of face recognition and face analysis. In a recent paper of Chen's [4], it is shown that simple features can achieve leading performance on face recognition if accurate facial landmarks can be utilized. For this reason, the problem of facial landmark localization has attracted extensive interests in the past years. In general, there are three main methods to locate the facial landmarks from a face image: the first category performs a sliding window search based on local-patch classifiers, which encounters the problems of the ambiguity or corruption in local features. Besides, it is difficult to incorporate the global contextual information into the local search framework; the second category of methods is the well-known framework of the Active Shape Model (ASM) [2] and the Active Appearance Model (AAM) [5]. These methods fit a generative model for the global facial appearance and hence are robust to local corruptions. However, to estimate the parameters in the generative models, expensive iterative steps are required.

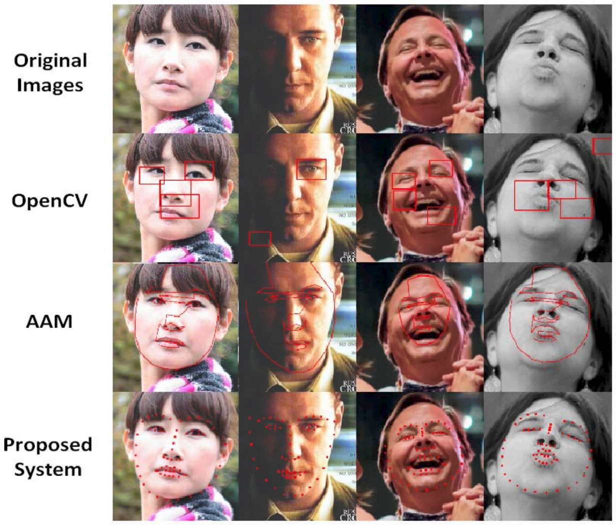

Figure 1. **Comparison of landmark localization systems**. The first row is the original facial image. The second row is produced by local-patch detectors included in OpenCV [3]. The third row is produced by Stasm [9], an open source AAM implementation. Our result is shown in the fourth row, which outperforms the rest significantly.

Recently, a new framework based on explicitly regression methods [10, 11] has been proposed. In this framework, the problem of landmark localization is considered directly as a regression task, and a holistic regressor is used to compute the landmark coordinates. Compared to the aforementioned methods, this framework is more robust and stable since the global contextual information is incorpo-

978-1-4799-3022-7/13 $31.00 © 2013 IEEE
DOI 10.1109/ICCVW.2013.58

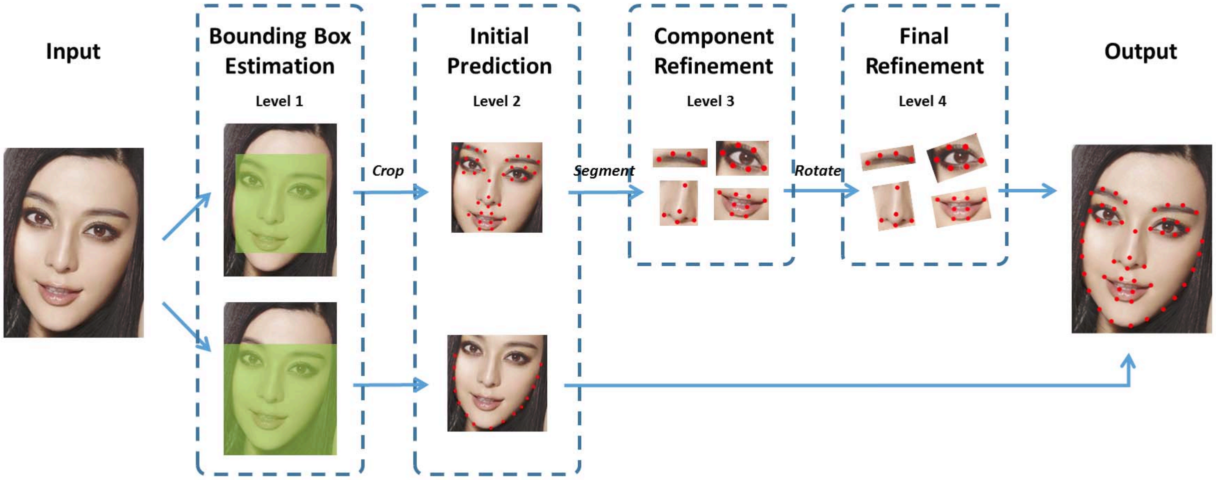

Figure 2. **System overview**. The first-level network predicts the bounding boxes for the inner points and contour points separately. For the inner points, the second level predicts an initial estimation of the positions which are refined by the third level for each component. The fourth level is used to improve the predictions of mouth and eyes by taking the rotated image patch as new input. Two levels of separate networks are used for contour points. For clarity reason, not all of the 68 points are rendered in the figure.

rated at the very beginning; it is also more efficient since no iterative fitting step or sliding window search is required. Instead of the random ferns used in [10], Sun [11] applies more powerful deep convolutional neural network (DCNN) in the regression framework and achieves the state-of-the-art performance.

However, facial landmark localization remains a very challenging problem. The challenge comes from the large variations of facial appearance due to the changes in pose, lightening, expression and etc. The task is even more challenging when a large number of landmark points is required. The nature of the challenge varies dramatically across different facial points, so a single-model method would probably fail. On the other hand, employing individual systems for each point sharply increases computational time. However, the large number of points is a two-edge sword: valuable information pertaining to the inner structure of the relative position of the landmarks becomes present. The geometric constraints on the global arrangement of facial components and the interaction of points inside a component provides hope for improvement in accuracy and robustness if the system amply exploits them.

To address the challenge, we carefully design a multi-level convolutional network cascade, which tackles the task of extensive facial landmark localization with a coarse-to-fine network cascade. Our contributions are three-fold: 1) unlike [11] predicts sparse facial landmarks (5 points) with network cascade, we validate the effectiveness of convolutional network cascade for the problem of extensive facial landmark localization; 2) we design a coarse-to-fine network cascade to spread the network complexity and training burden of traditional convolutional networks; 3) we show that explicit geometric refinement (estimate the position/rotation of facial components and rectify the inputs of each network level) can improve the accuracy and robustness significantly. Extensive experiments show that our system is accurate and robust.

2. Overview

Figure 2 gives a brief illustration of our multi-level facial landmark localization system. We use the term *inner points* to denote the 51 points for eyes, eyebrows, mouth and nose, and *contour points* for the 17 points on the contour. The subsystems for the inner points and contour points are separated from the first level. In the first level, two neural networks are trained to estimate the bounding boxes (the maximum and minimum value of the x-y coordinates) for the inner points and contour points independently. The boxes are fed into the rest of the system respectively.

Inner points. For the inner 51 points, three levels of convolutional neural networks are trained in addition. After obtaining the bounding box of inner points, the 51 inner landmarks are initially estimated by the second level. Based on the initial estimation, the regions for 6 facial components (i.e., eyebrows, eyes, mouth and nose) are computed in separate. The third level is trained to refine the landmarks of each facial component independently. The rotation angle of each component is estimated and corrected to upright, and the rotated patches are fed to the fourth level network for the final results.

Contour points. A simpler network cascade is utilized for the localization of contour points. Given the bounding box covering the cheek, the second level takes the cropped image as input and computes the coordinates of the contour points from the raw pixels. Third and fourth level networks are not utilized due to the limited time, and we leave the further exploitation of deeper network cascade to future work.

3. Coarse-to-fine DCNN cascade

The central idea of our framework is the design of coarse-to-fine cascade. Each network level refines a subset of the landmarks inside a region computed by previous levels. In the first level, the face is divided into two parts : inner and contour. After the second level, the facial components of inner part are further separated. We do not train individual networks for each facial landmark to reduce computational cost. There are multiple advantages of the coarse-to-fine framework.

3.1. Separation of the loss function

The hardness of localization is unbalanced across different landmarks. Particularly, the contour is significantly more difficult than inner points for two reasons. First, the facial image provides less local texture information for contour points compared to the inner landmarks, but the irrelevant information from the background near these points is noticeably more. Additionally, the ground truth for these points is by nature more noisy, because the definition of the exact position of each point is more ambiguous. These factors result in the heavy imbalance between the training errors of the two parts, hence the L2 loss function will be dominated by the contour if all 68 points are trained together. So training two independent subsystems give the whole system a chance to learn the detailed structure of inner points instead of devoting most of its capacity to fitting the "difficult" contour. This argument is supported by our experiment.

Among the inner points, the relative difficulties of the facial components are still not uniform. As shown in Section 5, eyebrows are notably harder whilst the system's prediction on eyes is more accurate.

3.2. Multi-level refinement

The localization task is decomposed into multiple stages at each of which the interaction between the points or components is considered. In the first level, the relative position of the face contour, which is closely related to the pose of the face, is computed. In higher levels, more detailed information is revealed step by step. The second level network learns the relative location of the facial components, and the task of recognizing the shape inside a component is handled by succeeding levels. It is possible that the third level network is compromised by local corruption. However, since global information is taken account in the second level, the final output still makes sense.

The bounding box carries the information of the position and range of the group of points to the next level. Thus the image inside the box is generally well aligned in terms of translation and scaling. In contrast, the rectangle generated by the face detector is far from satisfactory. In some cases, it contains too much irrelevant background information that confuses the neural network. Moreover, the face is not always centered in the rectangle, which further complicates the localization task for the system.

DCNN is generally considered to be powerful enough to handle great variation in the input image, but the capacity of a single network is still limited by its size. Given insufficient prior knowledge, the network will devote a considerable part of its power to finding where the face is. To tackle the problem, the "divide-and-conquer" strategy is adopted, which divides the task into two steps: first to find the overall position, then to compute the relative position inside the region. For the whole face, the first step is performed by the first level networks whose supervision signal does not include the detailed structure of the points inside the bounding box, and the rest of the task is left to succeeding levels. In this way, the burden is shared across networks in different levels, and good performance is achieved by networks of only moderate size.

The idea is extended further in the third and fourth level where the orientation is canonicalized by means of a rotation of the image patch. Rotation is considered only after the third level since the consequence caused by failure to predict a robust rotation angle in the early levels is serious. Experimental results show that the fourth level gives a performance gain that is not as dramatic as the previous levels but absolutely non-negligible.

4. Implementation Details

Deep convolutional neural network. We use DCNN as the basic building block of the system. The network takes the raw pixels as input and performs regression on the coordinates of the desired points. Figure 3 is an illustration of the deep architecture. Three convolutional layers are stacked after the input nodes. Each convolutional layer applies several filters to the multichannel input image and output the responses. Let the input to the t-th convolutional layer be I^t, then the output is computed according to

$$C^t_{i,j,k} = |\tanh(\sum_{x=0}^{h_t-1} \sum_{y=0}^{w_t-1} \sum_{z=0}^{c_t-1} I^{t-1}_{i-x,j-y,z} \cdot F^t_{x,y,k,z} + B_k)|$$

where I represents the input to the convolutional layer, F and B are tunable parameters. Following the standard practice, hyper-tangent and absolute value function are applied to the filter responses to bring non-linearity to the system.

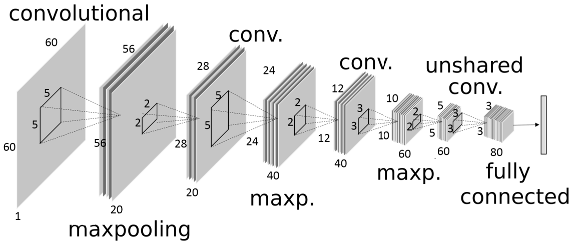

Figure 3. **Typical structure of networks in our system**. The network consists of convolutional layers, unshared convolutional layers and fully-connected layers. Max-pooling is performed after convolutional layers. In unshared convolutional layers, the weights used in different positions are different. Tanh and absolute value non-linearity is inserted between the layers. The architectures of other networks are similar to this.

network	N1	N2	N3
input	60x60	40x40	40x40
conv. 1	5x5x20	5x5x20	5x5x20
conv. 2	5x5x40	3x3x40	3x3x40
conv. 3	3x3x60	3x3x60	3x3x60
unshared	3x3x80	2x2x80	2x2x80
hidden			120

Table 1. Resolution, filter size and number of channels of the networks. N1 is used for inner points in the second level. N2 is used for contour points. N3 is used for others. Two fully connected layers are used in N3 and there are 120 hidden units between them. In N1 and N2, one fully connected layer directly connects the output units and the unshared convolutional layer.

Max-pooling with non-overlapping pooling regions is used after convolution

$$I^t_{i,j,k} = \max_{0 \le x < d, 0 \le y < d} (C^t_{i \cdot d+x, j \cdot d+y, k})$$

It seems unnatural to use max-pooling layers in a localization task that seeks pixel level accuracy. However, these layers are still adopted in the belief that the robustness of the whole system induced by these layers well compensates for the loss of information in the pooling operation, and the overall shape and relative position of the landmarks are more important than the pixel level detail in the input image. After the convolutional layers is an unshared-convolutional layer. The filter applied is not the same across different positions, so the layer is local-receptive rather than convolutional.

$$C_{i,j,k} = |\tanh(\sum_{x=0}^{h-1} \sum_{y=0}^{w-1} \sum_{z=0}^{c-1} I_{i-x,j-y,z} \cdot F_{i,j,x,y,k,z} + B_{i,j,k})|$$

The final prediction is produced by one or two fully connected layers. Parameters are adjusted to minimize the L2 loss:

$$\sum_{I^0} |layer_m \circ layer_{m-1} \circ \cdots \circ layer_1(I^0) - label(I^0)|^2_2$$

Network size The architecture of our DCNN is motivated by the work of [11]. Table 1 gives a summary of the network architectures. We employ three kinds of networks in different parts of the system. The network used in the second level, N1, has higher resolution since its input covers a range of the whole face.

Training. The neural networks are trained by stochastic gradient descent with hand-tuned hyper-parameters. To avoid severe over-fitting, the image is randomly altered by slight similarity transformation (rotating, translating and scaling) before feeding into the network. This step creates virtually infinite number of training samples and keeps the training error close to the error on our validation set. Also, we flip the image to reuse the left eye's model for the right eye, and left eye-brow for right eye-brow.

Image Processing. Image patch is normalized to zero-mean and unit-variance, then a hyper-tangent function is applied so that the pixel values fall in the range of $[-1, 1]$. When cropping the image inside a bounding box, the box is enlarged by 10% to 20%. More context information is retained by the enlargement, and it allows the system the tolerate small failures in the bounding box estimation step. In the fourth level, the rotation angle is computed from the position of two corner points in the facial component.

5. Experiment

We conducted our experiments on a dataset containing 3837 images provided by the 300-Faces in the Wild Challenge. The images and annotations come from AFW, LFPW, HELEN, and IBUG [6, 1, 12, 7, 8]. A subset of 500 images are randomly selected as our validation set. Two performance metrics are used on the validation set: the first one is the average distance between the predicted landmark positions and the ground truth normalized by inter-ocular distances

$$err = \frac{1}{N} \sum_{i=1}^{N} \frac{\frac{1}{M} \sum_{j=1}^{M} |\mathbf{p}_{i,j} - \mathbf{g}_{i,j}|_2}{|l_i - r_i|_2}$$

where M is the number of landmarks, p is the prediction, g is ground truth, l and r are the positions of the left eye corner and right eye corner. The second one is the cumulative error curve that plots the percentage of points against the normalized distance.

5.1. Validation of our method

The degree of difficulty in localizing the 68 landmarks varies dramatically. Figure 4 shows the validation error

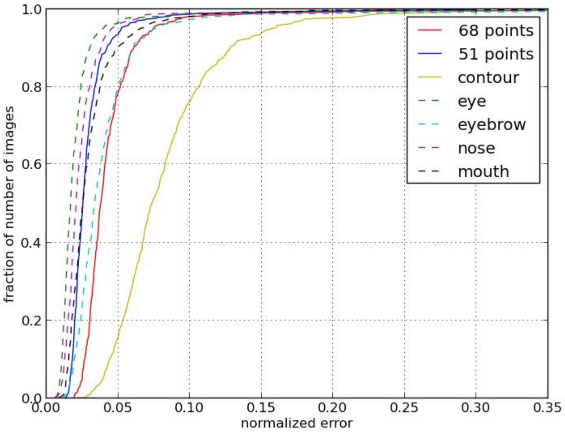

Figure 4. **Cumulative error curves on the validation set**. The errors of the whole face, contour points, inner points and different facial components are compared. It is shown that the hardness of different facial landmark points is heavily unbalanced.

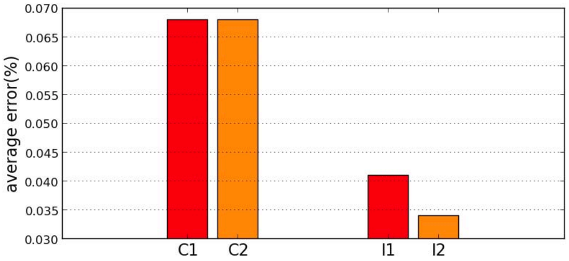

Figure 5. **Effect of separation of contour and inner points**. One network predicts the 68 points together, and its errors on the inner points and contour points is C1 and I1 respectively. C2 and I2 are achieved by two networks that predict those points independently.

for the different facial components. The performance on contour points is noticeably worse. This observation is the motivation of our idea to separate the contour from inner points.

Separating the contour from inner points is essential to our performance. We conducted a experiment in which three networks are trained. One of them, which as a larger size, predicts the 68 points together. The other two learned the contour and inner points respectively, and the sum of amount of computation involved in training and testing of the two networks roughly matches the big network. To eliminate other influence factors, the input region of the networks is computed from the ground truth value. Figure 5 shows that separation improves performance on inner points while the performance on the contour points is not worse.

In our system, the rectangle given by the face detector is not directly used to compute the input region of the network which produces actual facial landmarks. In contrast, the image for the second level is cropped according to the first level's prediction. If the face detector's box is used directly instead, performance deteriorates. Table 2 lists the validation errors achieved under various conditions. It indicates that performance is improved on the inner points by the bounding box estimated at the first level.

output	error value
(51 points) detector box	0.0662
(51 points) level 1 box	0.0401
level 2	0.0510
level 3	0.0438
level 4	0.0431

Table 2. **Validation errors achieved under various conditions**. The error value is average normalized distance between prediction and ground truth. The first two rows shows the error calculated on the inner points only, while other rows correspond to the average error on all of the 68 points.

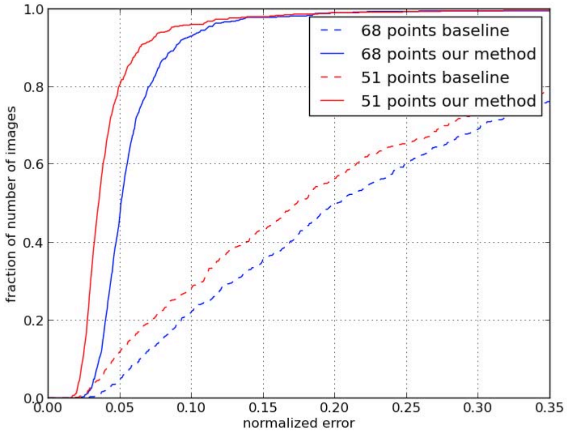

Figure 6. **Result comparison in the 300-W Challenge**. Our method outperforms the baseline significantly.

To quantitatively investigate the effect of the third and fourth level, we calculated the validation error achieved at each network level. Training separate networks for each component allows the third level network to improve the performance by 14%. Performance gain is still obtained by the fourth level in which rotation is rectified.

5.2. Comparison with other methods

Figure 6 is the result of our system in the 300 Faces in the Wild Challenge. Our result is far better than the AAM-based baseline. We also compared our performance on the validation set with other systems. Since those detectors produced different sets of facial landmarks, we only show the relative improvement on the common landmark points when comparing with them. Table 3 lists the results. Our system outperforms those public available or commercial landmark

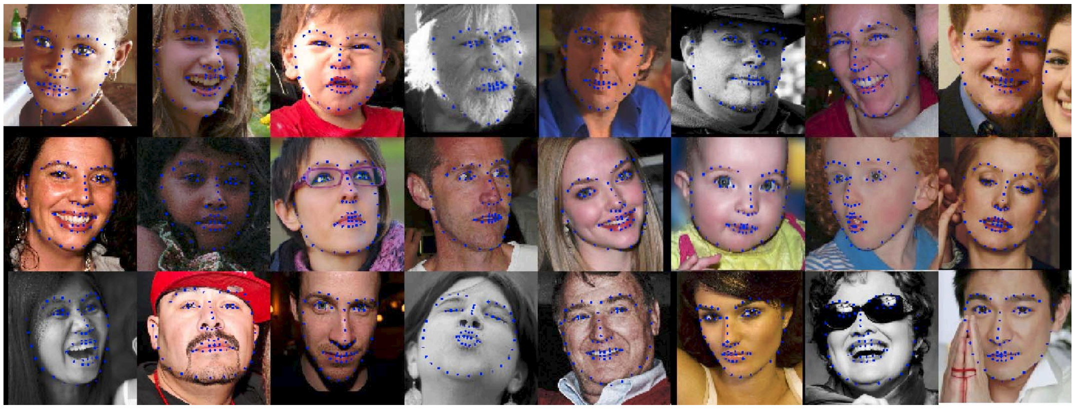

Figure 7. **Some examples from the validation set**. The data set contains great variations in pose and lightening condition, but our system is still able to give good result.

system	# points	error	ours	improvement
Intraface[1]	37	0.046	0.029	37%
FACE^{++}(1.0)[2]	3	0.075	0.029	61%
FACE^{++}	11	0.034	0.026	25%
Lambda Lab[3]	3	0.097	0.026	73%

Table 3. **Comparison with other public systems on the validation set**. The error values are average Euclidean distances normalized by inter-ocular distance. Our system outperforms other systems.

detection systems.

Figure 7 gives some examples taken from the validation set. Our system is able to handle images that contain great variation in pose and lightening condition. It can predict the shape of the face even in the presence of occlusion. Despite the success, chance for further improvement still exists, especially for the points on the eyebrow or face contour.

6. Conclusion

We propose a new automatic system for facial landmark localization. In our method, four DCNN levels are carefully designed to form a coarse-to-fine network cascade. To validate the effectiveness of our design, we show that our system can achieve leading performance in the 300-W facial landmark localization challenge.

References

[1] P. Belhumeur, D. Jacobs, D. Kriegman, and N. Kumar. Localizing parts of faces using a consensus of exemplars. *Computer Vision and Pattern Recognition*, 2011. 4

[2] A. Blake and M. Isard. *Active shape models*. Springer, 1998. 1

[3] G. Bradski. The opencv library. *Dr. Dobb's Journal of Software Tools*, 2000. 1

[4] D. Chen, X. Cao, F. Wen, , and J. Sun. Blessing of dimensionality: High-dimensional feature and its efficient compression for face verification. *Computer Vision Pattern Recognition*, 2013. 1

[5] Cootes, G. J. E. Timothy F., and C. J. Taylor. Active appearance models. *Pattern Analysis and Machine Intelligence*, 2001. 1

[6] R. Gross, I. Matthews, J. Cohn, T. Kanade, and S. Baker. Multi-pie. *Image and Vision Computing*, 2010. 4

[7] V. Le, J. Brandt, Z. Lin, L. Boudev, and T. S. Huang. Interactive facial feature localization. *European Conference on Computer Vision*, 2012. 4

[8] K. Messer, J. Matas, J. Kittler, J. Luettin, and G. Maitre. Xm2vtsdb: The extended m2vts database. *2nd international conference on audio and video-based biometric person authentication*, 1999. 4

[9] S. Milborrow and F. Nicolls. Locating facial features with an extended active shape model. *European Conference on Computer Vision*, 2008. 1

[10] C. Xudong, Y. Wei, F. Wen, , and J. Sun. Face alignment by explicit shape regression. *Computer Vision and Pattern Recognition*, 2012. 1, 2

[11] S. Yi, X. Wang, and X. Tang. Deep convolutional network cascade for facial point detection. *Computer Vision Pattern Recognition*, 2013. 1, 2, 4

[12] X. Zhu and D. Ramanan. Face detection, pose estimation and landmark localization in the wild. *Computer Vision and Pattern Recognition*, 2012. 4

[1] http://www.humansensing.cs.cmu.edu/intraface/

[2] http://en.faceplusplus.com/

[3] http://www.lambdal.com/