Computers & Graphics 000 (2017) 1–9

Contents lists available at ScienceDirect

Computers & Graphics

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

Special Issue on CAD/Graphics 2017

Better initialization for regression-based face alignment

Hengliang Zhu[a,*], Bin Sheng[a], Zhiwen Shao[a], Yangyang Hao[a], Xiaonan Hou[a], Lizhuang Ma[a,b]

[a] Department of Computer Science and Engineering, Shanghai Jiao Tong University, Shanghai, China
[b] Department of Computer Science and Software Engineering, East China Normal University, Shanghai, China

ARTICLE INFO

Article history:
Received 15 June 2017
Revised 27 July 2017
Accepted 29 July 2017
Available online xxx

Keywords:
Neighborhood representation prior
Occlusions
Projected initial shape
Cascade regression

ABSTRACT

Regression-based face alignment algorithms predict facial landmarks by iteratively updating an initial shape, and hence are always limited by the initialization. Usually, the initial shape is obtained from the average face or by randomly picking a face from the training set. In this study, we discuss how to improve initialization by studying a neighborhood representation prior, leveraging neighboring faces to obtain a high-quality initial shape. In order to further improve the estimation precision of each facial landmark, we propose a face-like landmark adjustment algorithm to refine the face shape. Extensive experiments demonstrate our algorithm achieves favorable results compared to the state-of-the-art algorithms. Moreover, our algorithm achieves a smaller normalized mean error than the human performance (5.54% vs. 5.6%) on the challenging dataset the Caltech Occluded Faces in the Wild (COFW).

1. Introduction

Face alignment is defined as the localization of facial landmarks such as eyebrows, eyes, nose tip and mouth corners. Efficient face alignment is critical in multimedia applications and often regarded as a pre-processing step for many vision tasks, such as face recognition [1,2], expression analysis [3–5], face makeup [6,7], and 3D face modeling [8–10]. However, accurate and robust landmark localization remains a big challenge for in-the-wild images that contain severe occlusions and large head rotations. In most regression-based algorithms, the bottleneck of this problem is the initialization, since a poor initial shape may subject face alignment into local optimum. The conventional algorithms trivially make use of the mean shape as initialization during testing, or randomly choose shapes from the training images during training [11–13]. Due to large occlusions and pose variations, poor initialization may cause the failure of face alignment. In this study, we aim to exploit a simple yet effective initialization algorithm that deals with these complex face images.

Recently, some algorithms have been proposed to improve the initial face shape [14–19], but they fail in the case of heavy occlusions and complex variations of poses and expression. These algorithms [14,16] use 3D information or other involved processes to get the initial shape. However, these algorithms cannot meet real-time applications requirements due to complexity and time consumption. Therefore, efficient initialization is still a challenge in cascade regression. Since a face can be approximated by a linear combination of similar faces (Fig. 2(a)), finding a good initialization algorithm means to search for the right facial neighborhood and compute how it should be weighted. However, it is difficult to find the neighborhood of a face, since its landmarks positions are unknown initially. To deal with this problem, we contribute our study, which leverages a sparse set of points to approximately find neighboring faces, which estimate 5 key points: two pupils, a nose tip, and two mouth corners. It is worth noticing that these facial points can approximate the face shape and pose (Fig. 1). A subset of multiple points (e.g. the aforementioned points) is easier to obtain and more robust to occlusions [20]. As shown in Table 2, the mean errors of the 5-point set are lower than the 68-point set. The reason is that these sparse points are the most prominent on a face and can achieve good performance even in variations of occlusion, pose and expressions. Furthermore, we found that the neighboring faces and weight coefficients on the proposed subset can be well applied to the full set of facial landmarks (Fig. 2(b)). Given those experiments, we propose a neighborhood representation prior: *Based on the face similarity computed on the subset, we can approximate the multi-point shape well using its neighboring faces.*

More specifically, the point information of the subset can be used to generate a high-quality initial shape, which reduces the errors of facial landmark detection. Based on the above observations, we propose a simple yet efficient algorithm to produce a desirable initialization, called the projected initial shape (PIS). We also

* Corresponding author.
E-mail addresses: hengliang_zhu@sjtu.edu.cn (H. Zhu), ma-lz@cs.sjtu.edu.c (L. Ma).

http://dx.doi.org/10.1016/j.cag.2017.07.036
0097-8493/

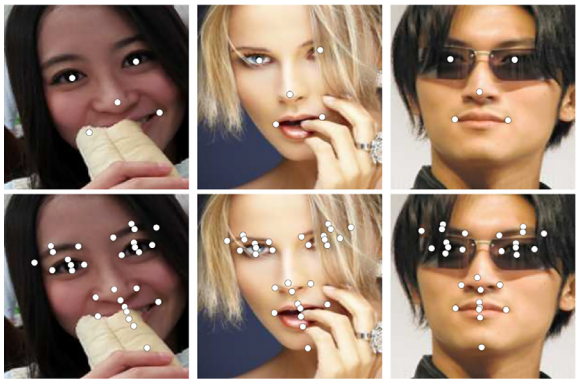

Fig. 1. Some experimental results on the COFW dataset and the images with heavy occlusions.

propose a face-like landmark adjustment algorithm for shape fine-tuning and final initialization. After that, the location of key facial points is improved in detail. The PIS can significantly improve both accuracy and robustness of face alignment. Experimental results show our new strategy is more efficient in dealing with heavy occlusions and large pose variations.

In this paper, we handle the problem of finding initial face for cascade regression-based algorithms. Our main contributions are summarized as follows:

- We propose a neighborhood representation prior to approximate the target face shape. The experimental results demonstrate the effectiveness of our assumptions.
- An efficient face alignment algorithm, utilizing the neighborhood representation prior, and a face-like landmark adjustment algorithm, is used to generate a better initial shape.
- Extensive experiments demonstrate the efficiency and robustness of our initialization scheme.

The paper is organized as follows. Section 2 reviews the related work. Then, the detail of the neighborhood representation prior is presented in Section 3. Section 4 depicts the details of our algorithm, and Section 5 demonstrates the experimental results and discussions. Finally, the conclusion is given in Section 6.

2. Related work

In this section, we mainly review research related to our work. In recent years, a number of regression-based algorithms have been proposed and have become popular for detection of facial landmarks [21–23]. These algorithms learn the feature mapping function from image appearance to the final shape. The classic active appearance model (AAM) [21] uses the difference between the current appearance estimate and the target image to drive an optimization problem. However, AAM is unfeasible for images with occlusions and large pose variations. Later, the fast AAM for face alignment was proposed [22]. Xiong and De la Torre [24] proposed a supervised descent algorithm to solve nonlinear least square problems based on scale-invariant feature transform (SIFT) [25]. Then a global supervised descent algorithm is proposed and performs well in facial tracking [26]. These algorithms may be difficult to handle complex scenarios, and fail to predict the landmarks accurately. Since feature descriptors that are extracted at occluded areas will greatly affect the update of the face shape at each iteration. It might result in a shape that is far away from the true landmarks.

Table 1
The speed and mean errors by the inter-ocular distance on 300W.

Result	Mean errors	Total time (ms)
Baseline + 5 points	5.58	4.48
Baseline + 8 points	6.12	4.80
Baseline + 14 points	6.10	5.08

A new cascaded shape regression (CSR) in face alignment is highly efficient in both training and testing. CSR algorithms use the image features to estimate the facial points in a cascaded way. For example, an explicit shape regression algorithm for face landmark location contains two-level regressors for shape estimation [11]. Kazemi and Sullivan [12] detected the landmarks by using an ensemble of gradient regression trees. Ren et al. [13] further improved this algorithm [11] and designed local binary features for shape regression. Burgos-Artizzu et al. [27] designed an occlusion-invariant face alignment algorithm that used shape indexed features and detected occlusions explicitly. Lee et al. [28] used cascade Gaussian process regression trees (cGPRT) for face alignment by using shape-indexed difference of Gaussian features to achieve robustness against geometric variances of faces. Wu and Ji [29] proposed a robust cascaded regressor to handle large pose and severe occlusions. Based on explicit head pose estimation, Yang et al. [16] presented a supervised initialization scheme for cascaded face alignment. Deng et al. [30] proposed a multi-view, multi-scale, and multi-component cascade shape regression (M3CSR) model for landmark prediction. Yang et al. [19] developed a spatio-temporal cascade shape regression (STCSR) model for robust facial tracking.

Most of the aforementioned algorithms start from a mean shape, and optimize the face shape iteratively. However, with a poor or wrong initialization, regression-based algorithms usually trap into local optimum [31]. By taking advantages of the neighborhood coherence in face similarity, our algorithm is fast, and can generate a high-quality initial shape closer to the true location. In order to alleviate the sensitivity to initialization, we propose a face

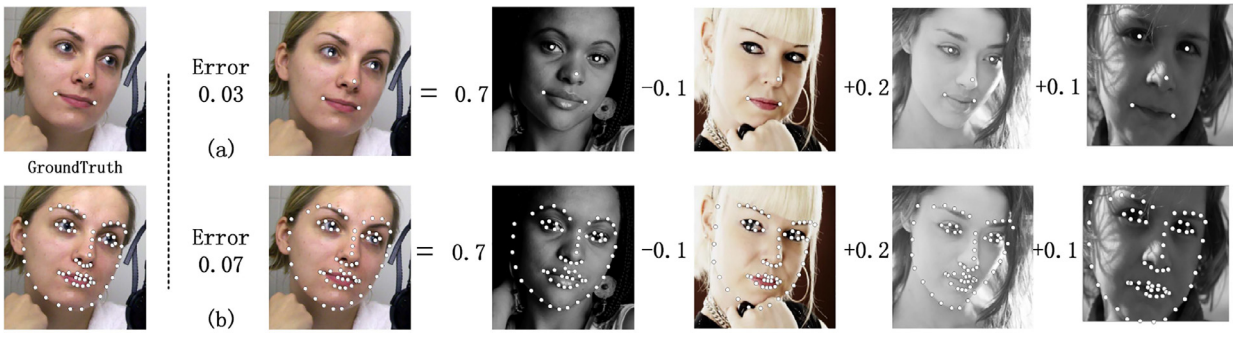

Fig. 2. The information (neighboring faces and weights) attained from the subset can be applied to the full set and still be a good approximation. The front numbers are weight coefficients. The errors are the distance between the predicted points and the ground truth, normalized by the inter-ocular distance.

Please cite this article as: H. Zhu et al., Better initialization for regression-based face alignment, Computers & Graphics (2017), http://dx.doi.org/10.1016/j.cag.2017.07.036

prior to obtain a better initial shape, which can significantly improve the performance of regression-based algorithms. In Section 4, we show our algorithm improves the accuracy and robustness in dealing with heavy occlusions.

3. Neighborhood representation prior

It is well-known that a face is approximately composed of a linear combination of its neighboring faces. However, the full set of key points is unknown in the beginning. Fortunately, we find the subset of points can be accurately obtained by using the cascade regression model (Table 2). Thus, we search neighboring faces in the subset and apply them into the full set to get an initial shape.

Define $\mathcal{V}^k = \{\mathbf{v}_i^k | i = 1, ..., N\}$, where $\mathbf{v}_i^k$ is a k*2 Dim vector representing a k-point set of the ith face in set $\mathcal{V}^k$, and N is the number of training samples. $\mathcal{V}^{68}$ is the full set, since the largest number of points used here is 68.

Given $\mathbf{v}_i^k$, we compute the Cosine similarity distance to search its m nearest neighbors, with the index in $\mathcal{V}^k$ being $\{u_1, \ldots, u_m\}$, where $m < k$*2. Then we compute weight coefficients to represent v_i^k by its nearest neighbors $\mathbf{V} = \{\mathbf{v}_{u_1}^k, \ldots, \mathbf{v}_{u_m}^k\}$:

$$\{w_1^*, \ldots, w_m^*\} = \arg\min_w \sum_{i=1}^{N} \|\mathbf{v}_i^k - \sum_j^m w_j \mathbf{v}_{u_j}^k\|_2^2. \quad (1)$$

We also expect the reconstruction error is small and expressed as

$$e_i(k, m) = \|\mathbf{v}_i^k - \sum_{j=1}^{m} w_j^* \mathbf{v}_{u_j}^k\|_2^2. \quad (2)$$

We apply the selected neighboring faces and weight coefficients obtained in the subset into the full subset. The average reconstruction error can be calculated as

$$e(k, m) = \frac{1}{N} \sum_i^N \frac{1}{\|v_i^k\|_2^2} e_i(k, m), \quad (3)$$

where $\frac{1}{\|v_i^k\|_2^2}$ is used for energy normalization, k is the dimension of the subset, and m is the number of neighboring faces .

In order to verify the neighborhood representation prior, we carry out experiments on the dataset 300 Faces in-the-Wild (300W). In this prior, a face can be represented as a linear combination of its neighboring faces, and we use parameters k and m to largely reduce the reconstruction error of Eq. (3). In complex scenarios, a larger number of facial landmarks further complicates the prediction. The human visual system suggests that the most important parts on a face are eyes, nose and mouth, which depict a face and always first attract attention from other people. Therefore, we select the most representative points to form a subset, and achieve subset of 68 points through regression [12]. Also, a larger number of test points would decelerate the prediction. In order to keep the fast speed and attain high accuracy at the same time, we design six groups of key subset points, as red points shown in Fig. 3. From top to bottom, they are 5, 6, 7, 8, 12 and 14 points, respectively. We can see that eyes, nose tip and mouth corners are all included in each group. For each group, the subset points are calculated directly from the multiple points (ground truth). We then calculate the average error for each group on 300W. The final results of $e(k, m)$ is illustrated in Fig. 5.

Fig. 5 shows that, in ideal conditions, a larger number of k facial points in the prior means a lower average error. But in practical scenarios, if k is too large, it is hard to predict point positions under the condition of large pose and occlusions. The average error minimizes when the number of similar faces is within [8, 10]. It is noteworthy that the error of the 6-point group, which has one point on the face contour, is lower than the 7-point group,

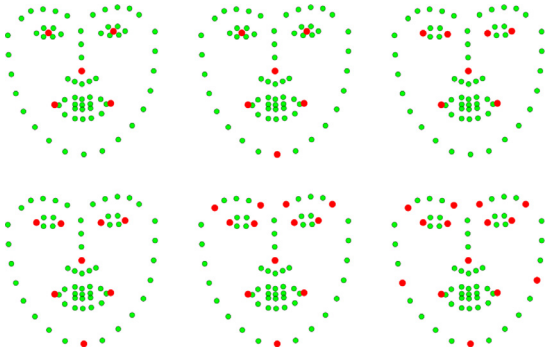

Fig. 3. Six groups of the key subset points. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

Table 2
Normalized mean errors by the inter-ocular distance on 300W. Our re-implemented results with 5-point set and the 68-point set.

Algorithm (ERT)	5 points	68 points
Original paper	–	6.40
Our re-implementation (baseline)	4.31	6.55

indicating the point on the face contour is also very important for shape estimating. Moreover, the computational cost to predict key points increases when k increases, and so we are presented with a trade off. Further, for complex face databases, mean error can actually increase as k increases. Table 1shows the performance of our algorithm when 5, 8 and 14 landmarks are included in the prior on the complex 300W dataset. The baseline algorithm is ERT [12]. Therefore, a more accurate key point detector helps us to generate a high-quality initial shape.

In addition, there are many algorithms for accurate location of 5 points [20,32]. These algorithms have successfully applied deep convolution networks to locate the key points, but they are not fast enough. In practical applications, fast speed is crucial in real-time landmark tracking. In this study, we take speed and accuracy into consideration, and focus on the 5-point group. As shown in Table 5, we can see that the five point detection takes less than 0.6 ms per image (about 1850 fps). In the future, we will utilize more points, such as 6 or 8 points, to generate a better initialization and acquire more accurate results.

4. Our algorithm

4.1. Initial prior

Based on the neighborhood representation prior, we propose an efficient and robust algorithm to approximate the facial landmarks. First, a 5-point detector is trained by using a cascade regression model [12]. Then this detector is used to locate the five key points for a face, including two left–right pupils, a nose tip and two mouth corners. The shape is denoted as $S_{\text{test}} = (\hat{Y}_1, \hat{Y}_2, ..., \hat{Y}_N)$, where $\hat{Y}_i$ denotes the ith face shape.

It should be noticed that the 5 points of the training images can be acquired directly by the ground truth. For the 68-point dataset, the pupils are not offered, so the average of the points along eye contour is regarded as the pupils. The set is denoted as $S_{\text{train}} = (\hat{X}_1, \hat{X}_2, ..., \hat{X}_M)$.

Training step: Most of regression-based algorithms utilize a random shape from other training images as the initial shape, which is very useful and effective in achieving better results for the frontal face. However, large pose and occlusions are ubiquitous

Please cite this article as: H. Zhu et al., Better initialization for regression-based face alignment, Computers & Graphics (2017), http://dx.doi.org/10.1016/j.cag.2017.07.036

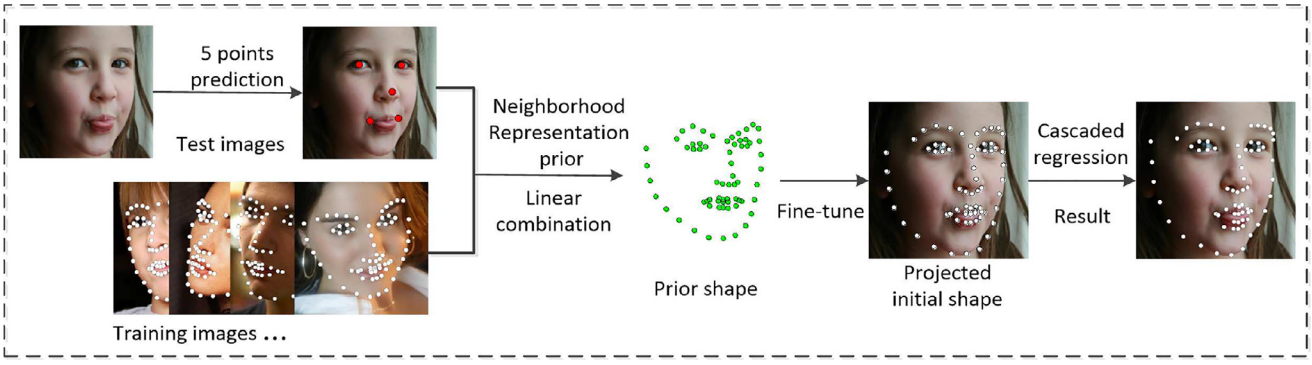

Fig. 4. Framework of our face alignment algorithm. Instead of using the mean shape as an initial shape, we use neighborhood representation prior to produce a projected initial shape (PIS) and get fine results.

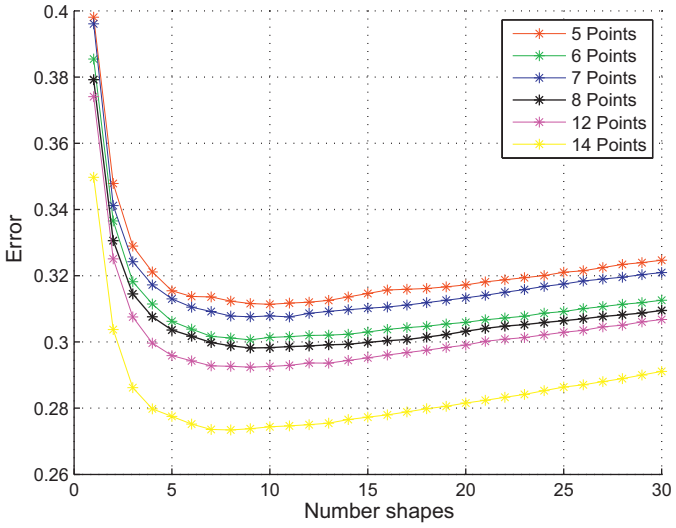

Fig. 5. Validation of the power of neighborhood representation prior on 300W.

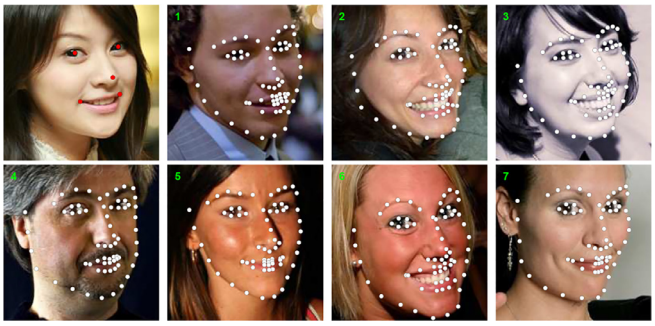

Fig. 6. Similar faces are selected using the neighborhood representation prior. The top left image is the testing image and its five landmarks, and the images with numerical symbols are the selected faces. Here, we only illustrate the first seven faces, and the rank of similarity from high (no. 1) to low (no. 7).

in the real world, so this rough way is not robust enough to handle the faces in complex scenes.

For a training image, we first get its 5 points, and then use Cosine similarity distance to calculate its similarity with the remaining training images, defined as $D(\hat{X}_i, \hat{X}_j), i \neq j, j = 1, 2, ..., N$, where $\hat{X}_i \in S_{\text{train}}$ is a 10-dimensional vector for the 5 facial points, and $D(\hat{X}_i, \hat{X}_j)$ is the shape error between the current face and other training samples. We can see that a smaller $D(\hat{X}_i, \hat{X}_j)$ indicates a higher similarity. This process of searching similar faces is efficient and accurate. Then, based on the similarity, the m most similar faces are selected. Following a previous work [12], we set $m = 20$ in the training step. This procedure contributes to dealing with large variations of occlusions and avoiding trapping into local optimum. As shown in Fig. 6, the selected faces are very similar to the current face in the terms of facial contour and head pose.

Testing step: In the testing step, we use a better initial shape generated from the proposed prior instead of the mean shape. Like the training step, we use Cosine similarity distance to measure the similarity between the testing face $\hat{Y}_i$ and the training images $\hat{X}_j$. The similar faces are denoted as $C = \{\text{sim}_1, \text{sim}_2, ..., \text{sim}_m\}$. Finally, based on Eq. (1), the selected neighborhood faces and weight coefficients are used to generate the prior shape, defined as:

$$S_{prior} = CW^T = \sum_{j=1}^{m} w_j^* sim_j \tag{4}$$

where S_{prior} is a prior shape, $W = \{w_1^*, \ldots, w_m^*\}$. Experiments show the result of using the simple average of similar faces is slightly better than using the weighted average. For simplicity, we use the average of similar faces and get $S_{\text{prior}} = \frac{1}{m}\sum_{i=1}^{m} \text{sim}_i$. The details of this situation are talked about in Section 5.4. Based on Fig. 5, the 10 most similar faces are selected from the training images in the testing step.

4.2. Face-like landmark adjustment

In order to get a high-quality initial shape closer to the target face, we propose a new algorithm to further optimize the prior shape S_{prior} to get a projected initial shape (PIS). This adjustment is only applied in the testing step. For a testing face, we have already obtained its prior shape, denoted as $S = \{S_1, S_2, ..., S_N\}$.

Then, we only choose from the prior shape the five key points, defined as $\hat{Z}_i$, including two left–right pupils, a nose tip and two mouth corners. Now a similarity transformation between $\hat{Z}_i$ and $\hat{Y}_i$, where $\hat{Y}_i \in S_{\text{test}}$ is created. This problem can be easily solved by using a reported algorithm [33]. Then, S_{prior} is transformed into a new shape by using the transform relationship, denoted as S_{tran}. As shown in Fig. 7, after shape fine-tuning, the location of key facial landmarks (eyes, nose and mouth) is improved in detail.

The final PIS for a testing face is defined as follows:

$$S_{projected} = \lambda S_{prior} + (1 - \lambda) S_{tran} \tag{5}$$

where the balance weight λ controls the level of shape fine-turning and is empirically set within [0.8, 0.9]. In our experiments, we set $\lambda = 0.8$. The whole framework of prediction is illustrated in Fig. 4. It should be noted that the computation of the initial shape is normalized based on a provided face bounding box.

Please cite this article as: H. Zhu et al., Better initialization for regression-based face alignment, Computers & Graphics (2017), http://dx.doi.org/10.1016/j.cag.2017.07.036

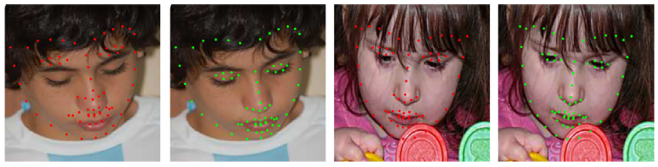

Fig. 7. Some visual examples of improvement for face-like landmark adjustment. The red and green colors depict the landmarks before and adjustment, respectively. It is noteworthy that the locations of landmarks in eyes, nose and mouth are improved. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

4.3. Implementation details

We have re-implemented the regression-based algorithm ERT [12] as our baseline for both the 5-point set and the 68-point set. The number of cascade stages is $T = 15$, each cascade stage consists of $K = 500$ weak regression trees, feature pool size is 400, and the depth of each tree is $D = 5$. In order to find the best split, we set the number of node split test to be 500. In the training step, different from the algorithm in Ref. [12] that uses 20 different shapes randomly sampled from training images, we choose the 20 most similar faces for initialization based on similarity values. This way can be found to achieve robust performance.

Following a previous work [12], we use the mean shape as the initial shape, and report the mean errors of the 5-point set and the 68-point set. All the experiments are conducted on 300W. As shown in Table 2, the result of our re-implementation is very close to the original paper and is used as the baseline in the experiments. It is worth noting that except lack of similarity transformation, the training step is similar to the testing step. Therefore, we only describe the details of the testing algorithm, as illustrated in Algorithm 1 .

Algorithm 1 Testing algorithm by using the neighborhood representation prior.

Input:
Testing images I_i, training images $\hat{S}_i$;
a 5 point set S_{test}, S_{train} and other parameters;
Output:
Shape estimations $\mathcal{S}^t$;

```
Initialize
Calculate shape similarity D(i, j)
Based on D(i, j), get the m most similar shapes sim_i from training images
Generate prior shape S_prior
Adjust face-like landmark S_tran ← S_prior
Get the projected initial shape S = λS_prior + (1 − λ)S_tran
for t = 1 to T do
    for k = 1 to K do
        Learn ΔS = R^t(I, S)
        Update S^t = S^{t−1} + ΔS
    end for
end for
```

5. Experimental results

Datasets: All the experiments are conducted on two public datasets: the Caltech Occluded Faces in the Wild (COFW) and the 300 Faces in-the-Wild (300W). The images of these datasets are very challenging owing to the occlusions and large variations of pose and expressions. For a fair comparison, following previous algorithms [12,27], we only use the training images from these two datasets and evaluate the testing images.

Table 3
Normalized mean errors on COFW (29 points). The results are marked with * from Ref. [29], others are cited from the original articles. Lower values are better, bold is the best score.

Method	Normalized mean error
Human [27]	**5.6**
ESR [11]	11.2*
RCPR [27]	8.5
SDM [24]	11.14*
HPM [40]	7.46
RPP [41]	7.52
TCDCN [42]	8.05
RAR [15]	6.03
Wu et al. [29]	5.93
Baseline	9.29
Baseline + PIS (ours)	**5.54**

COFW (29 landmarks) [27] contains about 1852 images with large occlusions, and has many images from MultiPIE [34]. COFW is designed to present faces in real-world conditions, and has an average occlusion of over 23%. The training set consists of 500 COFW images and 845 LFPW [35] images (1345 total), and the testing set includes the remaining 507 COFW images.

300W (68 landmarks) [36] is very challenging due to large changes in shape, pose, illumination and occlusions. It contains six famous datasets with 68 landmarks, including LFPW, AFW, HELEN, XM2VTS and IBUG. Following the work [11], we use AFW [37], the training images of LFPW and HELEN [38] as training sets (3148 images in total), and LFPW, HELEN and IBUG as full set for testing (689 images). In addition, the testing images from LFPW and HELEN are used as the common subset (554 images), and those from IBUG as the challenging set (135 images).

Evaluation metric: Following previous algorithms [11,39], we use standard normalized mean error to evaluate face alignment performance, and normalize all errors by using the inter-ocular distance. In this paper, all the results are the simplified form without '%' symbol. For comprehensive comparison, we also plot the cumulative errors distribution (CED) curves.

5.1. Comparison with other algorithms

Results on COFW: To verify the effectiveness and robustness of our algorithm, we compare it with other eight state-of-the-art methods on COFW, including ESR [11], RCPR [27], SDM [24], HPM [40], RPP [41], TCDCN [42], RAR [15], and Wu and Ji's [29]. As shown in Table 3, the performance of our algorithm is greatly improved on the challenging dataset COFW, which has large occlusions by different objects, such as hands, hair and glasses. Moreover, our result outperforms all reported results on COFW (3). Our mean error is even better than the human performance (5.54 vs. 5.6) [27], and is distinctly reduced by about 6.58% compared with the best result from Wu and Ji [29].

For validation of our algorithm, we first use the mean shape as the initial shape for testing (see the baseline result), and the mean error on COFW is 9.29. When using PIS for training and testing, the result is 5.54, with a reduction about 40.27%.

Results on 300W: We compare our algorithm with eight state-of-the-art algorithms, including DRMF [43], ESR [11], RCPR [27], SDM [24], CFAN [44], ERT [12], LBF [13], and CFSS [39]. As shown in Table 4, our algorithm outperforms most of other state-of-the-art methods. The results on 300W (common set or full set) performs well against LBF and CFSS. Moreover, our algorithm reduces the mean error dramatically. Compared with the baseline (i.e. ERT), our algorithm reduces the mean error by up to 8.1%, 24.7% and 14.8% on the common set, the challenging set and the full set, respectively (Fig. 4). Such large reduction rates indicate our algorithm is efficient and robust for the in-the-wild images.

Please cite this article as: H. Zhu et al., Better initialization for regression-based face alignment, Computers & Graphics (2017), http://dx.doi.org/10.1016/j.cag.2017.07.036

Table 4
Normalized mean errors on 300W dataset (68 points). The results are marked with * from Ref. [41], and others are cited from the original papers. Lower values are better, bold is the best score.

Method	Common	Challenging	Full set
DRMF [43]	6.65	19.79	9.22
ESR [11]	5.28	17.00	7.58*
RCPR [27]	5.67	15.50	7.54*
SDM [24]	5.60	15.40	7.52*
CFAN [44]	5.50	–	–
ERT [12]	–	–	6.40
LBF [13]	4.95	11.98	6.32
CFSS [39]	4.73	**9.98**	5.76
Baseline	4.81	13.7	6.55
Baseline + PIS (ours)	**4.42**	10.32	**5.58**

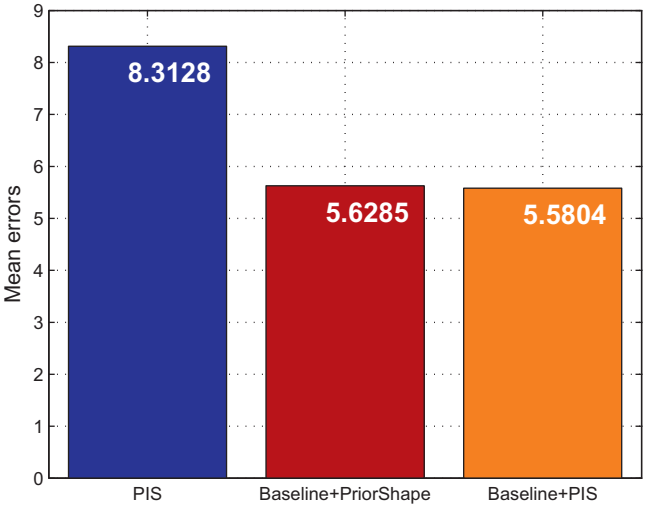

Fig. 8. The normalized mean errors of each component in our algorithm on 300W.

Table 5
The running time of each part on 300W. Part 1 is 5-point detection, part 2 is face similarity searching, and part 3 is 68-point regression.

Time	Part 1	Part 2	Part 3	Total
ms	0.54	1.10	2.84	4.48

In addition, the role of each component is also analyzed through computing the corresponding mean errors on 300W. These components include three parts: PIS, baseline + prior shape, and baseline + PIS.

Experiment results show the accuracy of face alignment is greatly affected by the initial shape (Fig. 8). By using our neighborhood presentation prior, we get a high-quality PIS, and improve the performance of the regression-based algorithms.

5.2. Comparisons of CED curves

Compared with four representative and prominent algorithms (ESR [11], ERT [12], RCPR [27], Wu and Ji [29]), we plot the CED curves on COFW (Fig. 9). Also, we plot the CED curves for the three typically algorithms ERT [12], LBF [13], CFSS [39] on 300W (Fig. 10). It is observed that our algorithm achieves the best performance on both two benchmarks.

As shown in Fig. 9, we can see that with the error smaller than 0.1, our algorithm and RCPR can localize the face landmarks for about 95% and 80% images, respectively. Therefore, our algorithm is strongly robust in getting better results under occlusion environments. Also, our algorithm can detect the face landmarks

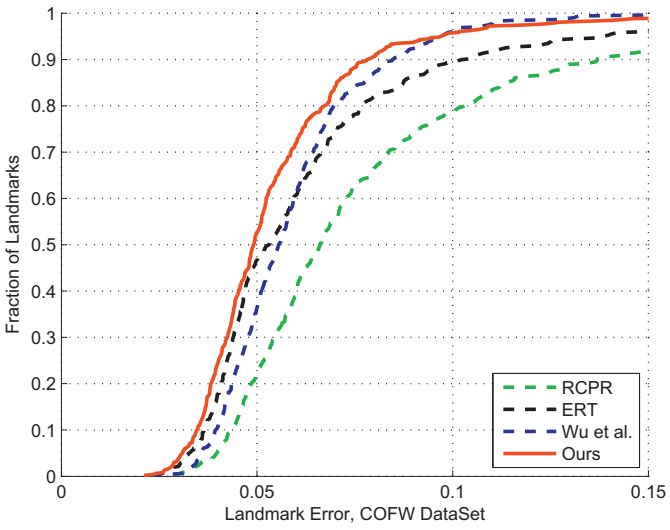

Fig. 9. Experimental results of CED curves on COFW.

Table 6
The speed compared with state-of-the-art algorithms on 300W.

Methods	FPS	Programming	CPU (Intel)
SDM [24]	70*	C++	i7-2600
CFAN [44]	44*	Matlab	i7-3770
ESR [11]	245	C++	i5-3470
ERT [12]	533	C++	i5-3470
LBF [13]	320*	C++	i7-2600
CFSS [39]	25	Matlab	i5-3470
Ours	223	C++ and Matlab	i5-3470

accurately on the full set of 300W, with error smaller than 0.1 in over 90% images (Fig. 10(a)).

We also illustrate some visible results of our algorithm on COFW and 300W (Figs. 11 and 12). Clearly, our algorithm locates landmarks accurately in the condition of large pose, complex occlusions and expression variations.

5.3. Running time

The running time of our algorithm can be divided into three parts, including five-point detection, face similarity searching, and multi-point regression. With 300W as an example, the time complexity for five-point detection is $O(TKDP)$, where $P = 5$ is the number of landmarks. The time complexity of the second part is $O(N)$, where $N = 3148$ is the number of training images. This similarity is obtained by computing Cosine similarity distance between a sample and the training dataset, and its computation takes little time. The time complexity of multi-point regression is $O(TKDP')$, where $P' = 68$. The part two is implemented in Matlab, and others in C++. The time of each part is demonstrated in Table 5. Clearly, our algorithm has a low computational complexity.

Table 6 shows the running time (frame per second or FPS) of different face alignment algorithms. The experiments are conducted with 300W on an Intel Core i5-3470 3.2 GHz CPU. Our algorithm is implemented on C++ and Matlab R2013a hybrid programming, and it takes less than 5 ms (223 FPS) to predict an image. We re-implement the representative ESR and ERT on C++. Both algorithms are very fast for face landmark detection, but their accuracies are lower than ours. For CFSS, we used the Matlab codes provided by the authors for comparison. Since neither CFAN nor LBF shares their codes, we choose the best published performance marked with '*'. CFAN takes about 22.84 ms per image, which runs

Please cite this article as: H. Zhu et al., Better initialization for regression-based face alignment, Computers & Graphics (2017), http://dx.doi.org/10.1016/j.cag.2017.07.036

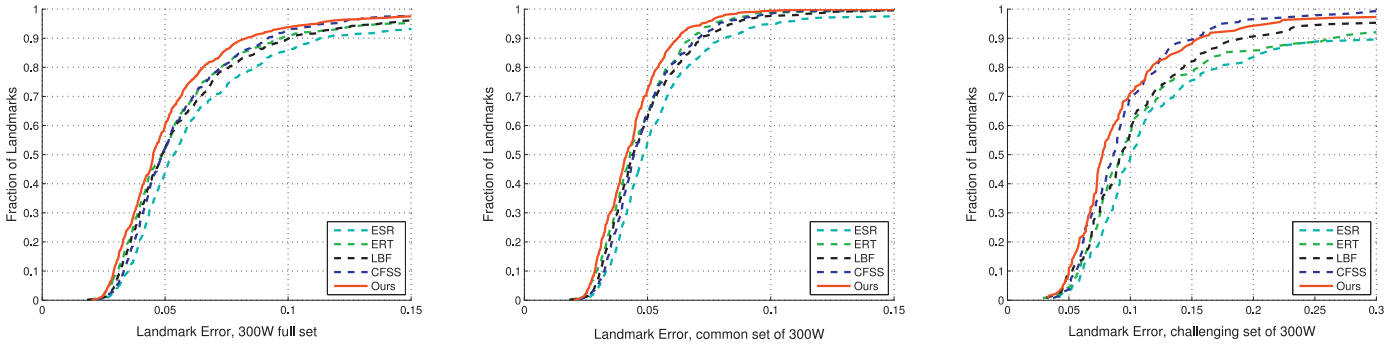

Fig. 10. Experiment results of CED curves on 300W. (a) Full set. (b) Common set. (c) Challenging set.

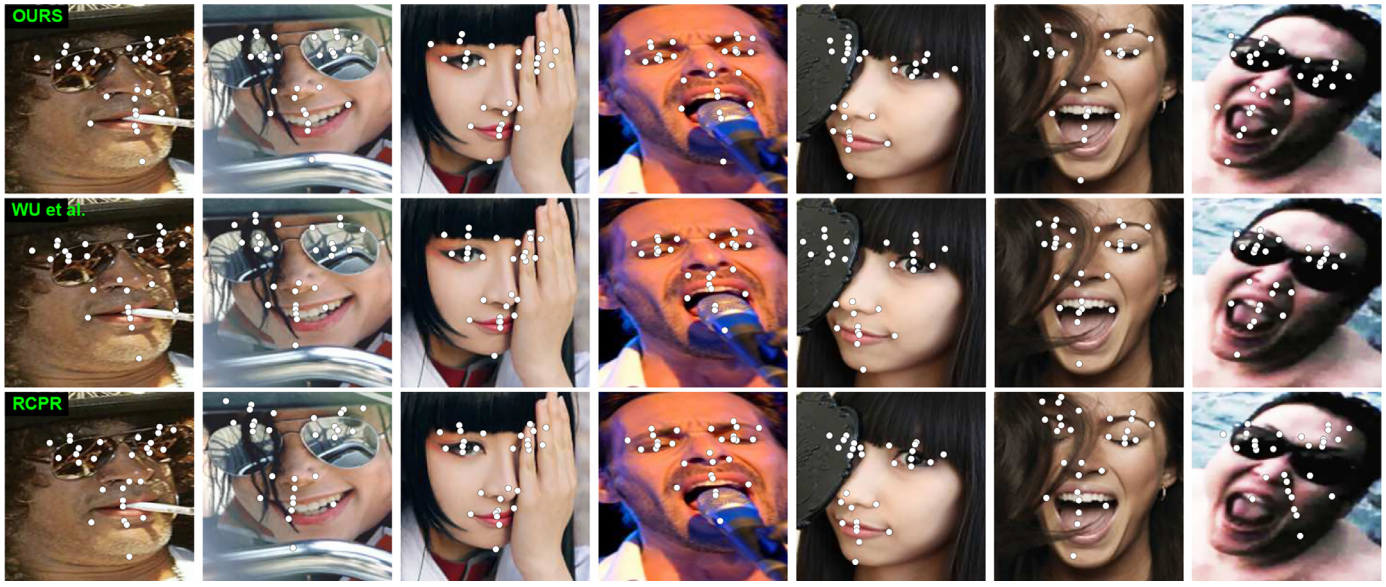

Fig. 11. Some images from COFW where our algorithm outperforms Wu et al.'s algorithm and RCPR. These images suffer from extreme occlusions.

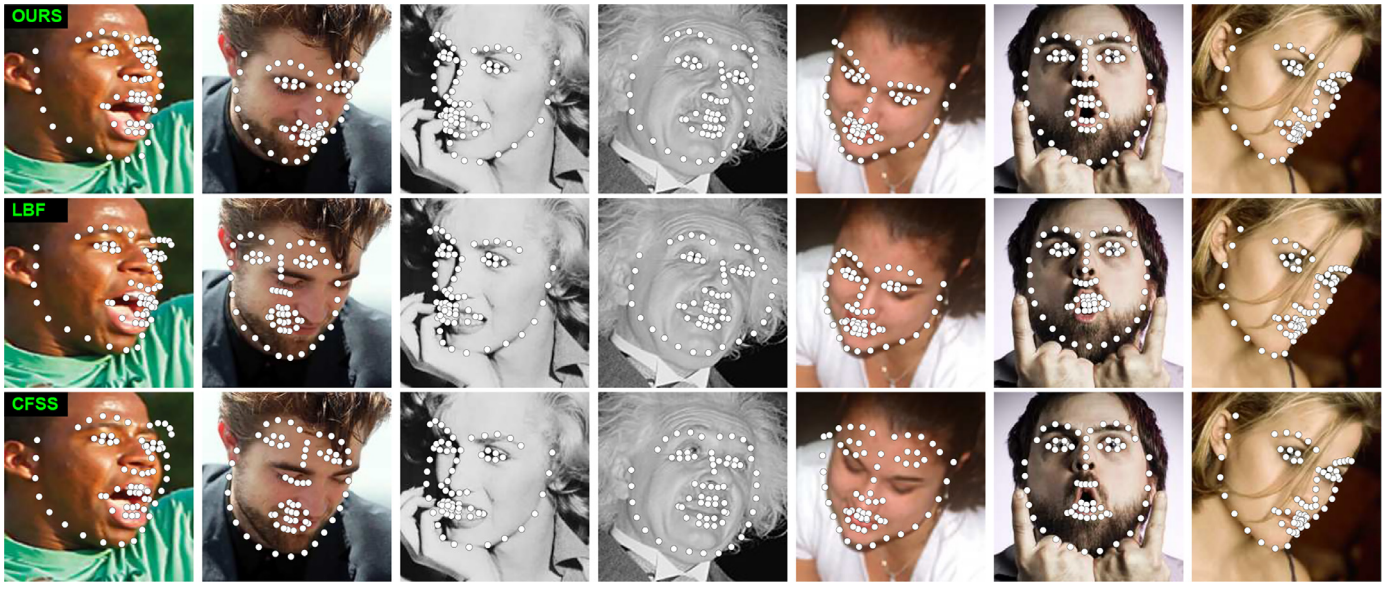

Fig. 12. Some results from 300W where our algorithm predicts more accurately than LBF and CFSS. These samples are challenging due to large variations of pose and expression.

Please cite this article as: H. Zhu et al., Better initialization for regression-based face alignment, Computers & Graphics (2017), http://dx.doi.org/10.1016/j.cag.2017.07.036

by Matlab 2012 on Intel i7-3770 3.4 GHz CPU. LBF spends 3.1 ms per image on a single core i7-2600 CPU. Although SDM is provided with executable code, it is only used to detect 49 landmarks. For a fair comparison, the speed of SDM in detecting 68 landmarks is cited from LBF [13].

5.4. Further analysis

In Section 4.1, we use the simple average of similar faces instead of the weighted average. We have two reasons. First, the optimized weight parameter in Eq. (1) is mainly used to verify our neighborhood representation prior: a face can be composed of a linear combination of other similar faces. The current solution of weights may be not optimal owing to the lack of effective constraints. In addition, the weight of a more similar face should be bigger. We could add constraints to the weight coefficients in Eq. (1), such as $w_1^* >= w_2^* >= \ldots >= w_m^*$, which will be exploited in the future.

Second, when we apply the weighted average on 300W, the mean error is slightly less than that of simple average (5.65 vs. 5.58) due to the under-fitting problem. Specifically, the algorithmic parameters are too simple to capture the underlying trend of the training data. For example, the testing samples contain many images with large pose variations, such as the head pose over 40°, but the training samples have few such images. Also, a higher diversity of training samples helps us to generate a better weighted average. Taking high accuracy into consideration, we use the simple average instead of the weighted average.

6. Conclusion

Previous works that usually start with the mean shape or use complex algorithms for initialization often fail to deal with face alignment under occlusions, large pose and expressions. In this paper, we use a neighborhood representation prior to generate a projected initial shape, which greatly improves the performance of cascade regression-based algorithms. In the condition of heavy occlusions, our initial scheme is efficient and the final result is better. Because of the faster speed, our algorithm can be used to track the facial landmarks in realtime. Our algorithm has some limitations, such as the low location precision of the five landmarks detector, which may greatly reduce the accuracy of similar face searching. Since a better key point localization helps to generate a better initial shape, we will further improve the performance of key point detection in future.

Acknowledgments

The authors would like to thank all reviewers for their helpful suggestions and constructive comments. This work is supported by the National Natural Science Foundation of China (nos. 61472245 and 61572316), National high and new technology research and development Program of China (863 Program) (no. 2015AA015904), and the Science and Technology Commission of Shanghai Municipality Program (no. 16511101300).

References

[1] Chen D, Cao X, Wen F, Sun J. Blessing of dimensionality: high-dimensional feature and its efficient compression for face verification. In: Computer vision and pattern recognition; 2013. p. 3025–32.
[2] Ding C, Choi J, Tao D, Davis L. Multi-directional multi-level dual-cross patterns for robust face recognition. IEEE Trans Pattern Anal Mach Intell 2014;38(3):518–31.
[3] Martinez A, Du S. A model of the perception of facial expressions of emotion by humans: research overview and perspectives. J Mach Learn Res 2012;13(1):1589–608.
[4] Baltrusaitis T, Robinson P, Morency LP. OpenFace: an open source facial behavior analysis toolkit. In: IEEE winter conference on applications of computer vision; 2016. p. 1–10.
[5] Haar FBT, Veltkamp RC. Expression modeling for expression-invariant face recognition. Comput Graph 2010;34(3):231–41.
[6] Li C, Zhou K, Lin S. Simulating makeup through physics-based manipulation of intrinsic image layers. In: Computer vision and pattern recognition; 2015. p. 4621–9.
[7] Qian K, Wang B, Chen H. Automatic flexible face replacement with no auxiliary data. Comput Graph 2014;45:64–74.
[8] Cao C, Weng Y, Lin S, Zhou K. 3D shape regression for real-time facial animation. TOG 2013;32(4):41.
[9] Jeni LA, Cohn JF, Kanade T. Dense 3D face alignment from 2D videos in real–time. In: IEEE international conference on automatic face and gesture recognition; 2015. p. 1–8.
[10] Hernandez M, Hassner T, Choi J, Medioni G. Accurate 3D face reconstruction via prior constrained structure from motion. Comput Graph 2017;66:14–22.
[11] Cao X, Wei Y, Wen F, Sun J. Face alignment by explicit shape regression. Int J Comput Vis 2014;107(2):177–90.
[12] Kazemi V, Sullivan J. One millisecond face alignment with an ensemble of regression trees. In: Computer vision and pattern recognition. IEEE; 2014. p. 1867–74.
[13] Ren S, Cao X, Wei Y, Sun J. Face alignment at 3000 fps via regressing local binary features. In: Computer vision and pattern recognition. IEEE; 2014. p. 1685–92.
[14] Zhu X, Lei Z, Liu X, Shi H, Li SZ. Face alignment across large poses: a 3D solution. In: Computer vision and pattern recognition; 2016.
[15] Xiao S, Feng J, Xing J, Lai H, Yan S, Kassim A. Robust facial landmark detection via recurrent attentive-refinement networks. In: European conference on computer vision. IEEE; 2016. p. 57–72.
[16] Yang H, Mou W, Zhang Y, Patras I, Gunes H, Robinson P. Face alignment assisted by head pose estimation. In: BMVC; 2015.
[17] Yang H, Zou C, Patras I. Cascade of forests for face alignment. IET Comput Vis 2014;9(3):321–30.
[18] Xiao S, Yan S, Kassim AA. Facial landmark detection via progressive initialization. In: IEEE international conference on computer vision workshop; 2015. p. 986–93.
[19] Yang J, Deng J, Zhang K, Liu Q. Facial shape tracking via spatio-temporal cascade shape regression. In: IEEE international conference on computer vision workshop; 2015. p. 994–1002.
[20] Zhang Z, Luo P, Loy CC, Tang X. Facial landmark detection by deep multi-task learning. In: European conference on computer vision, 2014. Springer; 2014. p. 94–108.
[21] Cootes TF, Edwards GJ, Taylor CJ. Active appearance models. IEEE Trans Pattern Anal Mach Intell 2001;23(6):681–5.
[22] Tzimiropoulos G, Pantic M. Optimization problems for fast AAM fitting in-the-wild. In: IEEE international conference on computer vision; 2013. p. 593–600.
[23] Saragih J, Goecke R. A nonlinear discriminative approach to AAM fitting. In: IEEE international conference on computer vision, ICCV 2007, Rio De Janeiro, Brazil, October; 2007. p. 1–8.
[24] Xiong X, De la Torre F. Supervised descent method and its applications to face alignment. In: Computer vision and pattern recognition. IEEE; 2013. p. 532–9.
[25] Lowe DG. Distinctive image features from scale-invariant keypoints. Kluwer Academic Publishers; 2004.
[26] Xiong X, De la Torre F. Global supervised descent method. Computer vision and pattern recognition; 2015. p. 2664–73.
[27] Burgos-Artizzu XP, Perona P, Dollár P. Robust face landmark estimation under occlusion. In: IEEE international conference on computer vision. IEEE; 2013. p. 1513–20.
[28] Lee D, Park H, Yoo CD. Face alignment using cascade gaussian process regression trees. In: Computer vision and pattern recognition; 2015. p. 4204–12.
[29] Wu Y, Ji Q. Robust facial landmark detection under significant head poses and occlusion. In: IEEE international conference on computer vision; 2015. p. 3658–66.
[30] Deng J, Liu Q, Yang J, Tao D. M3CSR: multi-view, multi-scale and multi-component cascade shape regression. Image Vis Comput 2016;47:19–26.
[31] Smith BM, Brandt J, Lin Z, Zhang L. Nonparametric context modeling of local appearance for pose- and expression-robust facial landmark localization. In: IEEE conference on computer vision and pattern recognition; 2014. p. 1741–8.
[32] Sun Y, Wang X, Tang X. Deep convolutional network cascade for facial point detection. In: Computer vision and pattern recognition; 2013. p. 3476–83.
[33] Goshtasby A. Piecewise linear mapping functions for image registration. Pattern Recognit 1986;19(6):459–66.
[34] Gross R, Matthews I, Cohn J, Kanade T, Baker S. Multi-pie. Image Vis Comput 2010;28(5):807–13.
[35] Belhumeur PN, Jacobs DW, Kriegman DJ, Kumar N. Localizing parts of faces using a consensus of exemplars. IEEE Trans Pattern Anal Mach Intell 2013;35(12):2930–40.
[36] Sagonas C, Tzimiropoulos G, Zafeiriou S, Pantic M. 300 faces in-the-wild challenge: the first facial landmark localization challenge. In: ICCVW. IEEE; 2013. p. 397–403.
[37] Zhu X, Ramanan D. Face detection, pose estimation, and landmark localization in the wild. In: Computer vision and pattern recognition, 2012. IEEE; 2012. p. 2879–86.
[38] Le V, Brandt J, Lin Z, Bourdev L, Huang TS. Interactive facial feature localization. In: ECCV European conference on computer vision, 2012. Springer; 2012. p. 679–92.

Please cite this article as: H. Zhu et al., Better initialization for regression-based face alignment, Computers & Graphics (2017), http://dx.doi.org/10.1016/j.cag.2017.07.036

[39] Zhu S, Li C, Loy CC, Tang X. Face alignment by coarse-to-fine shape searching. In: Computer vision and pattern recognition; 2015. p. 4998–5006.
[40] Ghiasi G, Fowlkes CC. Occlusion coherence: localizing occluded faces with a hierarchical deformable part model. In: Computer vision and pattern recognition; 2014. p. 1899–906.
[41] Yang H, He X, Jia X, Patras I. Robust face alignment under occlusion via regional predictive power estimation. IEEE Trans Image Process 2015;24(8):2393–403.
[42] Zhang Z, Luo P, Chen CL, Tang X. Learning deep representation for face alignment with auxiliary attributes. IEEE Trans Pattern Anal Mach Intell 2016;38(5):918–30.
[43] Asthana A, Zafeiriou S, Cheng S, Pantic M. Robust discriminative response map fitting with constrained local models. In: Computer vision and pattern recognition. IEEE; 2013. p. 3444–51.
[44] Zhang J, Shan S, Kan M, Chen X. Coarse-to-fine auto-encoder networks (CFAN) for real-time face alignment. In: ECCV European conference on computer vision. Springer; 2014. p. 1–16.

Please cite this article as: H. Zhu et al., Better initialization for regression-based face alignment, Computers & Graphics (2017), http://dx.doi.org/10.1016/j.cag.2017.07.036