Cascade of Forests for Face Alignment

Heng Yang, Changqing Zou, Ioannis Patras

Abstract

In this paper we propose a regression forests-based cascaded method for face alignment. We build on the Cascaded Pose Regression (CPR) framework and propose to use Regression Forest as a primitive regressor. The regression forests are easier to train and naturally handle the over-fitting problem via averaging the outputs of the trees at each stage. We address the fact that the CPR approaches are sensitive to the shape initialization, in contrast to using a number of blind initializations and selecting the median values, we propose an intelligent shape initialization scheme. More specifically, a large number of initializations are propagated to a few early stages in the cascade, then only a proportion of them are propagated to the remaining cascade according to their convergence measurement. We evaluate the performance of the proposed approach on the challenging face alignment in the wild database and obtain superior or comparable performance to the state-of-the-art, in spite of the fact that we have utilized only the freely available public training images. More importantly, we show that the intelligent initialization scheme make the CPR framework more robust to unreliable initialization that are typically produced by different face detections.

Index Terms

cascade detection, face alignment, regression forests.

I. Introduction

FACE alignment in an image is an active topic in computer vision. The face shape is typically described by a set of landmarks $\mathcal{S} = \{\mathrm{x}_1, ..., \mathrm{x}_k, ..., \mathrm{x}_K\}$, with x_k the coordinates of the kth landmark in the shape space. This alignment step is very crucial and often the first step for face biometrics [1], [2], [3], [4] and a variety of other applications like facial animation [5], facial expression recognition and face reconstruction [6]. It has been studied extensively and many alignment models [7], [8], [9] have been proposed, for instance the Active Appearance Model (AAM) proposed by Cootes *et al.* [10]. However, accurate face alignment meets great challenges in uncontrolled environment, such as low quality image, object pose variation and partial occlusions.

Cascaded Pose Regression (CPR) [11] method has emerged as an effective and accurate approach for object shape alignment. It starts from a gross estimate of the object pose and progressively refine the pose by shape increments, that are learnt by boosted regression in different stages. It was proposed in [11] for general object pose regression and then was extended for the problem of face alignment in [12] and [8]. They have achieved very promising performance both in terms of alignment accuracy and in terms of computational efficiency. Among those CPR approaches, *fern* is widely used as the primitive regressor.

H. Yang and I. Patras are with the Department of Electrical Engineering and Computer Science, Queen Mary University of London, London, UK,E1 4NS. e-mail: {heng.yang,i.patras}@qmul.ac.uk.

C. Zou is with Shenzhen Institutes of Advanced Technology, Chinese Academy of Sciences, Shenzhen, Guangdong, China. email:aaronzou1125@gmail.com

Manuscript received on 3rd April 2014, revised on 6th June 2014.

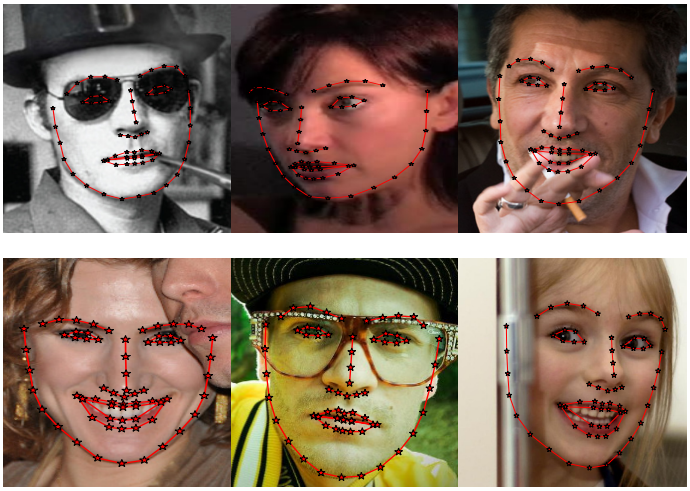


Fig. 1: Example results. We use a cascade of regression forests to estimate face landmark positions in 2D images.

We mainly make the following two contributions in this work.

First, we use regression forest, an ensemble of regression trees, as the primitive regressor in each stage of the CPR. By averaging the outputs of regression trees, we decrease the risk of over-fitting, which is a common problem of boosting regression. In order to compute the shape indexed feature at each internal node of the regression tree, in contrast to a approximate warping like [8], we directly select two random landmark indexes and a random bias value for each. In this way, the feature locations are directly indexed by the current shape and the most computationally expensive similarity transform is avoided. Second, we propose an intelligent initialization scheme. CPR is an initialization-dependent scheme and the outputs are sensitive to bad initializations. Previous methods like [8], [11] deal with this issue by restarting the initialization for several times and then the final output is calculated as the median value of the shapes of all the initializations. This scheme has two problems: first, the relationship among the shapes are not considered until they arrive the final stage; second, there is no theoretical justification of selecting the median value. Our proposed intelligent initialization first selects a large number of initialization shapes and then propagates them into the early stages of the cascade. Then, we calculate the density in the shape space using mean-shift. When the mode is detected, we sample a few shapes around it then propagate them to the remaining cascade. The final output is calculated as the average of the nearest pair of shapes.

Since most of the current facial feature detection approaches are built on top of the face detection, in this work we also investigate the sensitivity of the CPR face alignment method to different face detection initialization. As the initial shape is calculated based on the face detection, in our experiments, we have found that the current state of the art methods are very sensitive to the face detection. On the contrary, our proposed intelligent initialization scheme is more robust.

The remainder of the this paper is organized as follows: In Section 2, we briefly review the existing facial feature detection techniques related to our work. In Section 3, we first describe the general Cascade Pose Regression (CPR), and then present our Regression Forests based CPR and the intelligent initialization scheme. In Section 4, we show the experimental results of our proposed method on the latest 300-W database, that is a comprehensive database with common landmarks annotation cross different facial landmarks detection 'in the wild' databases. We close with concluding remarks in Section 5.

II. Related work

Object alignment is a well-studied problem in computer vision, particularly for the face. Two different sources of information are usually exploited for this task: object appearance and spatial shape. Based on how those two types of information is utilized, we categorize the methods into two groups: part-based deformable models and explicit shape regression.

A. Part-based deformable models

Approaches in the group involve two learning tasks, discriminative local detectors for each individual landmark and the shape prior of the face. The final detection is an optimization of the two terms. We will review the local detection and the shape models separately.

Discriminative Local Detection approaches exploit the discriminative appearance features of different landmarks. They can be classified into two groups, the regression based and the classification based. For instance, in [13], GentleBoost classifier based on Gabor features is proposed to detect 20 facial points separately. The classic Support Vector Machine (SVM) classifier is used as facial point detector in [14], [15], [16] and [17]. Several types of appearance features are utilized for training the discriminative local detector, for instance the most widely used Gabor feature in [18], SIFT feature in [17], [9] and the HoG feature in [19]. In [20], Boddeti *et al.* introduced Correlation Filters to learn the local appearance model. Regression-based approaches to facial point detection have attracted the attention of researchers in recent years. Cristinacce & Cootes [21] presented a regression-based approach to facial point detection. It combines a GentleBoost regressor with an Active Shape Model (ASM) that is used to correct the estimates obtained. Another sequential regression-based approach was presented in [22], where Support Vector Regressors (SVRs) were used. Regression forests in recent years have also proven to be very powerful in detecting facial points [23]. The location of facial point is estimated by accumulating *votes* from nearby regions. Methods in this category can be regarded as an extension of general object detection. Due to no shape constraints are imposed, this type of methods have good generality but suffer heavily from partial occlusions, therefore they are always combined with shape models.

Deformable Shape Models focus on building shape prior to regularize the local part detections. Typical shape models like the Constrained Local Models (CLMs) [7] first align the images using the landmarks annotation with Procrustes, then learn the shape prior by using Principal Component Analysis (PCA). Other shape models include the probabilistic MRF model in [18] and tree structured shape models proposed by Zhu and Ramanan [19], which have shown promising results both in capturing global elastic deformation and in finding the global optimal solutions efficiently. In addition, Amberg et al. [24] proposed to find an optimal set of local detections using a Branch & Bound method. Recently, instead of using parametric shape models, [17] propose more flexible, non-parametric representations of shape constraints from training data. Saragih *et al.* [25] also proposed a non-parametric representation to model the posterior likelihood and using mean-shift to optimize the result. [26] proposed to impose pairwise shape constraints within the regression forests. There are some other shape models based on facial points such as the Pictorial Structure[27], Markov Random Fields [22], Restricted Boltzmann Machines [28], graph matching [29], and Regression Forests votes sieving [30].

The advantages of part-based methods are two folds: first, any good object detector can be adapted for face alignment

problem like [23], since each individual landmark is treated separately; second, as a the local detection and shape model are learned separately thus it is easy to be extended. The shortcomings of the methods in this group are also two folds: first, the local detection is very sensitive to partial occlusion consequently it results in unreliable detection; second, the computational complexity of the methods are often exponentially related to the number of the landmarks to be detected, which limits the application of face alignment with large number of landmarks in real time.

B. Explicit shape regression

Approaches in this group jointly model the shape and appearance, and learn directly a mapping from image features to shape space (locations of landmarks). Since a direct mapping function is difficult to learn, many approaches in this group work in a boosting/cascade way. The typical method is the Active Appearance Models (AAMs) proposed by Cootes *et al.* [10]. The AAMs is fit by learning a linear regression between the increment of motion parameters and the appearance differences. Since very simple linear regression method is applied, the original fitting method suffers from occlusion and is very difficult to deal with unseen images. Saragih and Gocke [31] and Tresadern *et al.* [32] showed that using boosted regression for AAM discriminative fitting significantly improved over the original linear formulation. Instead of using the Newton's gradient descent method for optimization in the original AAM method, Xiong and De la Torre [9] proposed a supervised descent method for minimizing the non-linear least squares function, which has achieved very good result in face alignment application. [33] studied the optimization problem of AAM in detail.

In recent years, a framework called Cascaded Pose Regression was proposed for the problem of general object alignment which has been successfully applied in face alignment [11], [12], [8], [34]. They learn a set of weak regressors (random ferns) to model the relation between the image feature and the update in the parameter space. In the cascade sequence, they proposed using the re-sampled features based on the current shape state for the next regressor. Due to the simple feature evaluation procedure in the fern regressor, it is highly efficient in both training and testing.

Since most of the current Explicit Shape Regression methods work in a cascade way, they are all initialization dependent. In order to be less sensitive to the initialization, the CPR methods [8], [34] proposed to restart the initialization for several times and select the median value of the final estimations. But how to set the initialization is still an open question in the CPR.

III. METHOD

In this section we will present our method of cascaded forests for face alignment. First we give a brief summary of Cascaded Pose Regression (CPR) and then present how regression forest is used as regression primitive. Finally we present our proposed intelligent initialization scheme that can be used for the general CPR.

A. Cascaded Pose Regression (CPR)

The shape of an object is often represented as a vector of landmark locations, i.e., $S = (\mathrm{x}_1, ..., \mathrm{x}_k, ..., \mathrm{x}_K) \in \mathbf{R}^{2K}$, where K is the number of landmarks. $\mathrm{x}_k \in \mathbf{R}^2$ is the 2D coordinates of the k-th landmark. CPR is formed by a cascade of T regressors, $R^{1...T}$. Shape estimation starts from an initial shape S^0 and progressively refines the pose. Each regressor refines the pose by

producing an update, ΔS, which is added up to the current shape estimate, that is,

$$S^t = S^{t-1} + \Delta S. \tag{1}$$

The update ΔS returned by the regressor that takes the previous pose estimation and the image feature I as inputs:

$$\Delta S = R^t(S^{t-1}, I) \tag{2}$$

An important aspect that differentiates this CPR framework from the classic boosted approaches is the feature re-sampling process. More specifically, instead of using the fixed features, the input feature for regressor R^t is calculated relative to the current pose estimation. This is often called pose-indexed feature as in [11]. This introduces weak geometric invariance into the cascade process and shows good performance in practice. The CPR is summarized in Algorithm 1 [11].

Algorithm 1 Cascaded Pose Regression

Input: Image I, initial pose S^0
Output: Estimated pose S^T

1: **for** t=1 to T **do**
2: $\quad f^t = h^t(I, S^{t-1})$ ▷ Shaped-indexed features
3: $\quad \Delta S = R^t(f^t)$ ▷ Apply regressor R^t
4: $\quad S^t = S^{t-1} + \Delta S$ ▷ update pose
5: **end for**

The above scheme holds several advantages. First, though for each stage, the pose-indexed feature is re-calculated, the original image feature, that can be more than image gray scale values, requires only one computation as a preprocessing step. Thus the feature re-calculation in practice is highly efficient. Second, the number of the landmarks representing the object shape has little impact on the testing efficiency since it only involves a vector addition operation, while other methods like [19], [23], [31], the computational complexity is linearly or exponentially related to the number of landmarks. Thus, besides the effectiveness in real application, the CPR is very popular due to its computational efficiency.

The essential part of the CPR is the primitive regressor. We follow the main scheme of CPR and use Regression Forest as our primitive regressor in each stage, as described in the following.

B. CPR training

In order to train a cascade of forests, let us assume we are given a set of n training samples $\{(I_i, S_i)\}_{i=1}^n$. I_i represents the image of the i sample and S_i, the ground truth shape. We assume here that the image only contains the face or has the bounding box of the face, since our algorithm is built on top of the face detection. For each training sample, we randomly select 20 ground truth poses from the training set except its own. We treat an individual training sample with a different initialization as a new sample. Each training sample is now represented by a triplet, that is $(I_i, S_i, \bar{S}_i)$, with $\bar{S}_i$ the initial pose. The augmented number of training samples is therefore $N = 20 \times n$.

For each training sample, with the current pose $\bar{S}$ and the ground truth pose S, the target update vector the regressor aims to estimate is

$$\Delta S = S - \bar{S}. \tag{3}$$

Thus at each stage we train a regressor at each stage that minimizes the square error loss, given the features f_i^t calculated using the previous pose state.

$$R^t = \arg\min_R \sum_i |R(f_i^t) - \Delta S_i^t| \tag{4}$$

The training procedure of the CPR is summarized in Algorithm 2.

Algorithm 2 Cascaded Pose Regression Training

Input: training data $(I_i, S_i, \bar{S}_i)$ for $i = 1...N$
Output: $R = (R^1, ...R^T)$

for t=1 to T **do**
  for all $i \in (1...N)$ **do**
    $\Delta S_i^t = S_i^t - \bar{S}_i^t$ ▷ Calculate ΔS_i^t
    $f_i^t = h^t(I_i, \bar{S}_i^{t-1})$ ▷ Shaped-indexed features
  end for
  $R^t = \arg\min_R \sum_i |R(f_i^t) - \Delta S_i^t|$
  for all $i \in (1...N)$ **do**
    $\bar{S}_i^t := \bar{S}_i^t + R(f_i^t)$ ▷ Update current pose
  end for
end for

C. Forest-based regressor

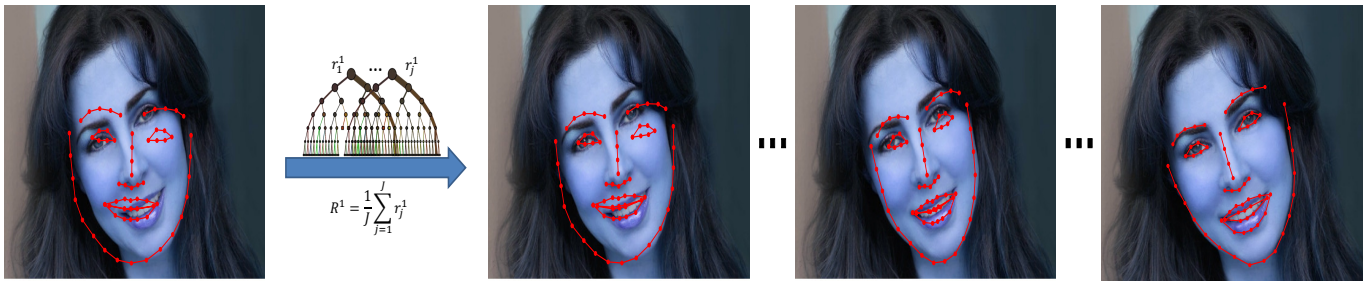


Fig. 2: Starting from a raw pose, our method refines the face shape recursively by using different stages of regression forests, organised in a cascade.

In this section we discuss how a primitive regressor, a forest is trained. A forest is an ensemble of regression trees. The simplest version of a forest consists of one tree. Thus we first discuss how a regression tree is trained and then discuss the ensemble method.

Let $\mathcal{X}$ denote the input space, $\mathcal{Y}$ the output space. Each tree is induced based on a randomly selected subset of the training data $\mathcal{D} \subseteq \mathcal{X} \times \mathcal{Y}$. An empty tree starts with only one root node. Then a number of *split test function* candidates $\phi : \mathcal{X} \to \{0, 1\}$ are generated, which determines whether to route a data sample $x \in \mathcal{X}$ reaching it to go left or right child. $\mathcal{P}_{left}(\phi)$ and $\mathcal{P}_{right}(\phi)$. According to one specific split function ϕ, the set of data, denoted by $\mathcal{P}$ ($\mathcal{P} \subseteq \mathcal{D}$), at the node will be partitioned into two, $\mathcal{P}_{left}(\phi)$ and $\mathcal{P}_{right}(\phi)$. Based on the partition, each candidate split function is evaluated according to a certain loss function, so that the best split function, that is the one with the minimum value of the loss function, is selected, i.e. $\phi^* = \arg\min_\phi \mathcal{L}(\phi)$. The node is parametrized by the selected split function ϕ^*. Then, the training set is partitioned according

to this split function into two subsets that are propagated to the two child nodes. The same procedure is recursively applied at each subsequent child node. The procedure stops and a leaf node is created when certain criteria is met, typically, when there are fewer than a minimum number of training data or a maximum tree depth is reached. At each leaf node, a regression model is learned and stored.

According to the above description of tree construction, aside from the macro parameters of the tree, there are two tasks involved: specifying the split test function at each internal node and learning the regression model at each leaf node. As discussed before, in order the keep the high efficiency of the algorithm, we focus on very simple test functions, that is to compare the feature values at two pixel locations. Besides gray scale, other pixel-wise features can also be used such as the Gabor features, with an additional cost of feature computation. So as to generate a pool of split testing functions, we randomly select two landmark numbers, l_1 and l_2. Then we generate an random offset to each of the two landmark locations, δ_1 and δ_2. Thus for the training sample i the first location feature indexed by the current pose is:

$$x_i^1 = I_i^{\bar{S}_i(l_1)+\delta_1} \tag{5}$$

where $\bar{S}_i(l_1)$ denotes the image location of the l_1-th landmark, deduced by the current pose estimate $\bar{S}_i$. The second location feature is $x_i^2 = I_i^{\bar{S}_i(l_2)+\delta_2}$. The split function consists of five parameters, $\phi = (l_1, l_2, \delta_1, \delta_2, \tau)$, where τ is a threshold variable. Formally the split function ϕ is written as:

$$\phi_{(l_1,l_2,\delta_1,\delta_2,\tau)}(I_i, \bar{S}_i) = \begin{cases} 0 & \text{if} \quad x_i^1 - x_i^2 > \tau \\ 1 & \text{otherwise} \end{cases} \tag{6}$$

In order to select the best split function candidate at each node, based on the loss function in Eq. 4, we rewrite the objective function as:

$$\mathcal{L}(\mathcal{P}, \phi) = \sum_{c \in \{left, right\}} \sum_{i \in \mathcal{P}_c} |\Delta S_i - \mu_c| \tag{7}$$

where

$$\mu_c = \frac{1}{|\mathcal{P}_c|} \sum_{i \in \mathcal{P}_c} \Delta S_i \tag{8}$$

is the mean value of the update vectors. The optimal split candidate is selected as the one which has minimized the above loss function, i.e.,

$$\phi^* = \arg\min_\phi \mathcal{L}(\mathcal{P}, \phi) \tag{9}$$

When training samples arrive the leaf node, the regression model is calculated as the average pose update vector of all the training samples in question, similar to Eq. (8).

Instead of using one tree as weak regressor at each stage of the cascade as described above, we train a forest consisting of a set of trees, that is $R = \{r_j\}_{j=1}^J$. The output of the forest is the average of the predictions of all the trees, that is,

$$R^t(S^{t-1}, I) = \frac{1}{J} \sum_{j=1}^{J} r_j^t(S^{t-1}, I). \tag{10}$$

The averaging regularization is able to deal with the general over-fitting problem in boosting regression. This will be demonstrated in the experiments.

D. Intelligent initialization

The output of CPR is initialization dependent and very sensitive to bad initializations. Previous approaches such as [8], [11] propose to run multiple different initializations and pick up the median of all the predictions as the final output. Each initialization is treated in a completely independent way until the output is calculated. The theoretical support of selecting the median value is not well understood. Also there is no guidance on how to choose the multiple initializations.

We propose an intelligent initialization scheme, which works in a coarse-to-fine manner. We build an initialization pose dataset with M instances, each with a unique pose consisting of K landmark locations. Given a testing image, we randomly select m initializations, $m \leq M$. The number of m is set to a large number, around ten times larger than the number of initializations used in the previous approaches [8], [11]. Instead of applying the whole cascade on the m initializations, we apply only a few top stages of the cascade and analyse their results. Specifically, we apply the mean-shift algorithm to find the mode of the estimated shapes using the small number of top stages, that is the shape with highest density in the shape space. Then the remaining cascade is applied on m' poses, which are closest to the shape mode. $m' \ll m$ is a very small number that can be even smaller than the restart number in [8], [11].

We now discuss the theoretical support of this scheme. As discussed in [8], at the early stage, the regressors in the cascade aim at adjusting the global shape updates such as yaw, roll and scaling. In later stages, the regressors are dominated by the subtle variations such as motions on eyes and lips. Therefore, we assume that a good initialization aligns the rough shape in a few stages while a bad initialization progresses towards a wrong position. Also we assume that in most of the cases, there are more good initializations than bad initializations given the fact that we have augmented multiple random initializations during the training stage. The first assumption is validated by the Principal Components analysis in [8] and the second assumption was implicitly used in the previous approaches. Given these two assumptions, we believe that the m' initializations we selected as discussed above are more reliable and are more likely to converge towards the correct pose position.

Since we only apply a very small number of stages in the cascade on the m raw initializations thus we can still expect very high evaluation efficiency. When m' initializations arrive the end of the cascade, since the number of $m's$ is very small, we calculate the distance between each pair and then select the the pair with minimum distance. The final output is calculated as the mean value of the selected pair, this is different from selecting the median value.

IV. Experiments Setting

To evaluate the efficacy of the proposed approach, we conduct the experiments on face alignment from a single image. The face images are collected in uncontrolled environments, and taken from various viewpoints and often present in cluttered backgrounds, with severe partial occlusion.

A. Datasets

There are several datasets are collected for the problem of facial feature detection, including those collected in the laboratory, such as BioID [35], XM2VTS [36] and PUT [37] and those collected from the Internet such as LFPW, LFW. As the state of the art has already reached very similar to human performance on datasets in the laboratory, we only list the publicly available datasets recorded 'in the wild' as shown in Table I. These datasets are all collected from the Internet, from search engine results or from Flickr. Most of those images exhibit a very large variability in pose, lighting, expression as well as general imaging conditions. Many images exhibit partial occlusions that are caused by head pose, objects (e.g., glasses, scarf, food), body parts (hair, hands) and shadows. We make a brief review of the characteristics of all the recent publicly available datasets below, so as to assist people working in this topic to select proper datasets to evaluate their methods.

LFW is a dataset recorded for face recognition problem and annotated in [23], in which close-to-human facial feature detection performance has been reported. The main challenge of this dataset is the low resolution of images. LFPW is one of the most used datasets to benchmark facial feature detection in uncontrolled conditions in recent two years. But as it only provides the urls of the images, it is difficult to make very fair comparison. The very recent methods have reported close-to-human performance on it [9], [34]. ALFW is a dataset with up to 21 facial landmarks were annotated, of which no annotation was present if the point was invisible caused by heavy occlusion or large head pose changes. Some annotations are of big error. ALW is a dataset with side-view face images. Zhu and Ramanan [19] created it for testing their joint method of face detection, pose estimation and facial landmarks detection. HELEN consists of very high resolution face images. The images were hand-annotated using Amazon Mechanical Turk to precisely locate the eyes, nose, mouth, eyebrows, and jawline. Then review and post-processing were carried out by the authors to ensure the facial landmarks annotations are also highly accurate and reliable. Sun et al. [38] created the Sun-CNN dataset. It consists of 5,590 LFW images and 7,876 other images downloaded from the web for training and 1,521 BioID images and 1030 LFPW images for testing. Only 5 salient points, the two eye centres, nose tip and two month corners were annotated, which are relatively easy to detect. 300-W was created for Automatic Facial Landmark Detection in-the-Wild Challenge using a semi-automatic annotation methodology [39]. Landmark locations for four popular data sets, LFPW, AFW, HELEN and XM2VTS, are re-annotated with 68 points mark-up, as illustrated in Fig. 3. COFW is a newly created dataset by [34], of which images exhibit very heavy partial occlusion by sunglasses, scarf, hair, hands and other objects. Additional occlusion state (occluded/unoccluded) annotations are provided for each landmark. Their images have an average occlusion of over 23%, which further introduces the difficulty of the task. The method in [34] is an occlusion centred approach that leverage occlusion information to improve the robustness of shape updates in cascaded pose regression.

In terms of facial landmarks number and face image number, the datasets vary a lot from each other. The landmarks number in the above described databases varies from 5 to 194. How many landmarks are needed in a specific applications remains an open question. Generally speaking, 5 or 10 facial landmarks can only assist very rough face alignment. In order to do subtle analysis for instance facial expression recognition, large number of facial landmarks are often required.

In this work we mainly focus the comparison on the LFPW and HELEN datasets, with the annotation from 300-W, as it provides annotations of large number of common landmarks for several widely used datasets. Since 300-W has not made

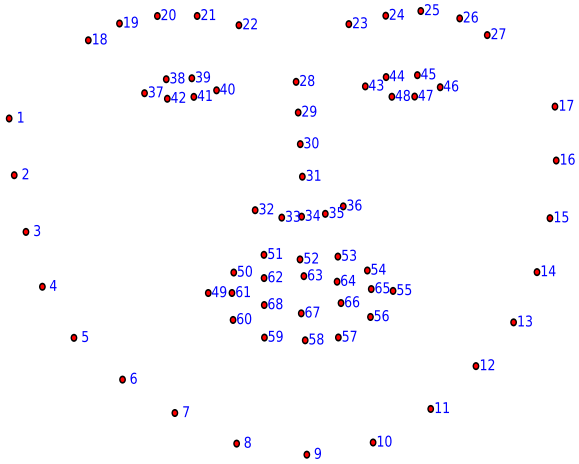


Fig. 3: The 68 points mark-up used for annotations in 300-W.

its test images publicly available, we follow the experimental setting (training/testing partition) of LFPW and HELEN when comparing to other methods. We compare the Regression Forests related methods on LFW dataset and follow the experiment setting of [23].

TABLE I: Description of datasets for face alignment in the wild.

Datasets	# landmarks	Resolution	# images	Ref. methods	Pub. code	Notes
LFW	10	low	13233	[23], [26], [9], [34]	[23], [26]	Multiple images for one person
LFPW	29	high	871+239	[17], [8], [34], [9]	[9]	Only urls are provided
AFLW	up to 21	diverse	25993	[40], [41], [30]	[30]	Not fully annotated
AFW	39 or 68	high	436	[19]	[19]	Side-view faces annotated differently
HELEN	194	very high	2000+330	[42], [34]	[34]	N/A
Sun-CNN	5	diverse	13466+2302	[38]	[38]	Incl. LFW, BioID and LFPW
300-W	68	high	135+300	TBD	TBD	Incl. LFPW, HELEN, AFW, XM2VTS
COFW	29	diverse	1007	[34]	[34]	Landmarks visibility annotated

B. Implementation details

We train our model using the training partition of LFPW and HELEN with 68 landmark annotations provided by 300-W. As mentioned in Section III-B, we augment the training data with 20 training poses for each training sample. For each tree in the forest, we keep the same parameter setting. The depth of the tree is set to 5. At each internal node, in order to select the best split function, we generate 400 candidate split functions that consists of a pair of locations, the corresponding offsets as well as a threshold. In the cascade, at each stage, i.e. for each forest we use 5 weak tree regressors and in total we have trained $T = 500$ stages of forests.

During testing, we create an initialization set with 500 pose instances, i.e. $M = 500$. In order to generate intelligent initializations, we set $m = 100$, i.e. randomly select 100 pose instances from M. We apply the top $\frac{1}{10}$ cascade on the m initialization instances and then select the best $m' = 5$ pose instances, as discussed in Section III-D, that are allowed to go through remaining cascade and generate the final output.

C. Evaluation measurements

In the literature, it is commonly accepted that the individual detection error is measured as the distance between the detected landmark location and the ground truth, normalized as a fraction of the inter-ocular distance [23], [8], [34], [18] (or the face size [19]). In order to measure the performance on a dataset, there are several measurements were proposed, including overall average landmarks error [34], landmark-wise average error[17], [43], [23], cumulative distribution function (CDF) of landmark-wise error [8], [23], CDF of face-wise error [7] and failure rate [23], [34]. As most of the current methods have achieved very high accuracy, within an error level of 10 (as a fraction), it is difficult to evaluate the algorithm using the CDF as most of the errors are within small values. We for comparison report the the overall and landmark-wise average error as well as the failure rate of the algorithm. The failure is determined if the average error is larger than 10, as defined in [23].

V. Results

A. Method evaluation

1) Cascade stages: It is an open question that how many stages in the cascade should be set for a specific problem. Since the testing time just depends linearly on the number of stages in the cascade, increasing the number of the stages does not influence the testing much. We have tried in our experiments by increasing the number from 100 to 450, with a step size of 50. The performances on the LFPW and HELEN are shown in Fig. 5 and Fig. 6 respectively. Note that the failure proportion here is calculated as when the face wise average error over all 68 landmarks is larger than 10 (as a fraction), that is different from Table IV, where the failure is calculated over the common 49 points. On testing images from both datasets, the mean error and failure proportions decrease gradually while the number of stages increases from 100 to 400. On the HELEN dataset, the failure percentage decreases from around 11.0% to 7.7% and the mean error decreases from 5.2 to 4.75. On the LFPW dataset, The failure percentage decreases from 6.8% to 4.78% and the mean error decreases from 5.5 to 4.78. When the stage number keeps increasing, on the HELEN dataset, the performance decreases while on the LFPW dataset, the performance has slight change. Thus we will set the $T = 400$ as the optimized stage number in the cascade. The overall performance of landmark-wise mean error using 400 cascade stages regression on the HELEN is shown in Fig. 4, where the landmark IDs are defined in Fig. 3. All the landmarks mean errors are smaller than 10, and the error of the landmarks along the face contours (from 1 to 17) are bigger than the internal landmarks.

2) Intelligent initialization: In this section we evaluate the effectiveness of the proposed intelligent initialization scheme. We compare it to the blind initialization scheme that is used in the traditional CPR method, i.e. to propagate a set of initializations till the final stage of the cascade. To make a fair comparison, instead of selecting the median values, we also apply our proposed method to calculate the final shape pose. The comparison is shown in Table II. As can be seen, using the intelligent initialization just slightly reduces the mean error, but greatly reduces the failures. Note that the failures and average landmark error is calculated over the all 68 landmarks.

3) Image feature: As we have discussed in Section III-C, in the primitive regressor, not only the gray scale gray scale image feature can be used. We also evaluate other high level grid based features like the Gobor feature, image edges, etc. In order to train the model with the compact features, we set the training parameters the same as that is used to train the model with

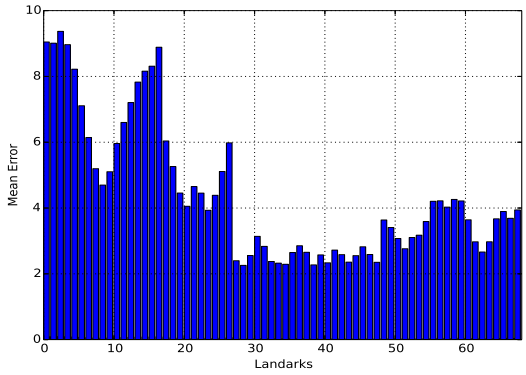


Fig. 4: Mean error of individual landmarks on the HELEN.

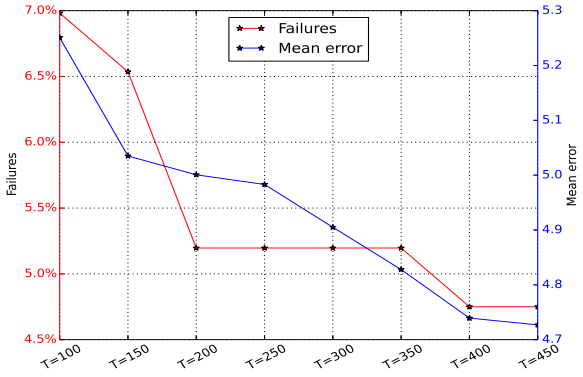


Fig. 5: Performance against cascade levels on LFPW.

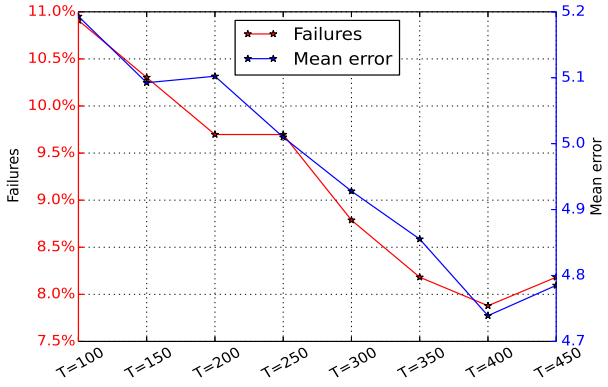
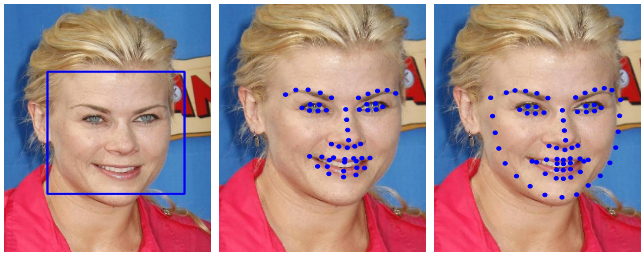


Fig. 6: Performance against cascade levels on HELEN.

TABLE II: Intelligent initialization vs. blind initialization.

	Blind Initialization	Intelligent Initialization
LFPW	8.1%/4.95	4.8%/4.73
HELEN	10.2%/5.19	7.7%/4.78



(a) Viola-Jones detector (b) SDM (c) Our method

(d) IBug Detector (e) SDM (f) Our method

Fig. 7: With different face detection initialization.

single channel gray scale feature. When testing on the same images, the model with compact features (gray scale, 32 channels of Gabor features and two channels of gradient) performs slightly better in terms of the failures. More specifically, it reduces the failures by 1.7% on the LFPW and 1.3% on the HELEN. In terms of alignment accuracy, there is no significant difference by using the compact features. However, due to the Gabor feature computation is very time consuming, the speed (FPS) is 3 times slower. Therefore in order to keep the highly computational efficiency we will just keep the gray scale feature in our experiments.

4) Different face detection: Most approaches in the face alignment (facial feature detection) assume the face detection (face bounding box) is available. Only a few methods like [19] integrates the face detection and landmarks detection. However, there is no standard definition of the face bounding box. It varies from methods. The most commonly used face detection method is the Viola-Jones face detector [44]. In other face databases, different face bounding boxes are provided like the 300-W. Since all boosting method starts from an initial shape, the face bounding box affects the initialization shape, in return affects the final shape regression. We compare our method to Xiong and De la Torre's Supervised Descent Method (**SDM**) [9], the state-of-the-art method in literature, with different face detection. An example face image from the LFPW is shown in Fig. 7. The face bounding box returned from the Viola-Jones detector is Fig. 7a and the face bounding box in 300-W is shown in Fig. 7d. The facial landmarks detected from the SDM and our method with the two different face detections are shown in the second the third column respectively. Fig. 7b shows the landmarks detection based on the Viola-Jones face bounding box while Fig. 7e shows the landmarks detection based on face bounding box in the 300-W. Fig. 7c and Fig. 7f are the landmarks detection results of our method based on the Viola-Jones detector and 300-w face detector. Since SDM is trained on the face images with bounding boxes returned from Viola-Jones detector, the landmarks localization in Fig. 7b is much more accurate than that in Fig. 7e. On the testing images, by using a different face detector, the failure rate of SDM increase 21% while that of our method increases 7%, 1/3 of SDM. This validates our method is more robust to different face detection initialization.

TABLE III: Comparison to RF methods on LFW.

Methods	C-RF	RF-CLM	RF-S	Our method
Mean Error	7.1	6.5	6.2	5.3
Speed (FPS)	25	12	10	35

B. Comparison to Regression Forest methods

Since our method uses Regression Forest (RF) as the primitive regressor, we first compare the related methods that use RF for facial feature detection including the Conditional Regression Forests (C-RF) method in [23], the Regression Forest based Constrained Local Model (RF-CLM) in [41] and the recent Regression Forests votes sieving (RF-S) in [30]. We note that the RF in these methods is used in a different way from our proposed method. While in their RF framework, local patches are used to cast votes for individual landmarks, in our method, RF is used as a holistic regressor for the update of the whole shape. The comparison is made on the LFW dataset on which the related methods reported results. We follow the experiment setting of [23] for all these methods. The results of the mean error of the 10 facial landmarks and the test run-time performance (It is measured on a standard 3.3GHz four-core machine) is shown in Table III. Our RF-based outperforms the counterparts significantly in both accuracy and efficiency, despite the fact that the other RF methods use the four cores for parallel computation but our method uses only one core. The other RF methods work in a sliding window fashion and cast votes for each individual landmark separately, therefore the computational complexity grows exponentially when the number of landmarks increases. On the contrary, our method treats the shape as whole, thus the number of landmarks will not affect the run-time performance. Our method can also detect 68 landmarks on other datasets at a speed of 35FPS.

C. Comparison to other methods

TABLE IV: Comparison with the existing methods. C. represents the common 49 facial landmarks that SDM and other methods can detect while 66P represents the 66 common landmarks the methods except SDM can detect.

Method Description			LFPW			HELEN		
Method	Model trained on	# of points	C. ME	C. Fails	66P ME	C. ME	C. Fails	66P ME
Mix.Tree [19]	Multi-PIE	68	11.4	27.3%	15.2	12.6	26%	14.7
DRMF [45]	Multi-PIE+LFPW	66	4.4	7%	5.8	4.6	**4.8**%	5.4
SDM [9]	Multi-PIE and LFW-A_C	49	4.27	**2.7%**	N/A	3.67	5.33%	N/A
CPR [8]	LFPW/HELEN	68	5.1	6.5%	5.7	4.8	7.5%	5.8
RCPR [34]	LFPW/HELEN	68	4.9	4.2%	5.2	4.5	6.1%	5.2
Our method	LFPW/HELEN	68	**3.92**	3.5%	**4.91**	**3.65**	6.37%	**4.78**

Closely related to our work are the CPR-based methods [11], [12], [8], [34]. The current one with the best performance is [34], that has used additional occlusion annotation for model training. We use the code that is provided by [34], which also contains a re-implementation of [8]. We use the same training/testing setting as our model on the LFPW and HELEN dataset. The comparison is shown in Table IV. As can be seen, the proposed approach outperforms the baseline CPR model as well as the RCPR method. Note that, since there is no occlusion annotation on HELEN and LFPW, we only use their feature extraction and their proposed smart restart components for a fair comparison. The superior performance validates the efficacy of our proposed strategy.

We also compare the performance of our approach with the state of the art methods with publicly available code. We compare with the following methods, 1) Xiong and De la Torre's Supervised Descent Method (**SDM**) [9], 2) Asthana et al.'s Discriminative Response Map Fitting (**DRMF**) method [45] running on the best performing tree-based model, 3) Zhu and Ramanan's Mixture of Trees (**Mix.Tree**) model [19].

We apply the publicly available code of SDM, DRMF and Mix.Tree on the testing images from LFPW and HELEN in 300-W. From the description of the papers, the model of SDM detector is trained on Multi-PIE [46] and LFW-A&C datasets, DRMF, trained on Multi-PIE and the LFPW training set while Mix.Tree is trained on Multi-PIE. CMU Multi-PIE face database contains more then 750,000 images of 337 people under various view points (15) and different illumination conditions while displaying a range of facial expressions. However it is not freely available to the public. Therefore, our model is trained on the freely available database in order to make the future comparison more convenient. All methods except the Mix.Tree and DRMF are built on top of the face detection. SDM is based on the Viola-Jones face detector while our method is based on the face detector in [39]. Thus for fair comparison, in case a face detector fails, we will manually set a proper bounding box for a face.

The comparison on the testing images from LFPW and HELEN is shown in Table IV. By comparing the mean error of the common 49 landmarks (C. ME) and the failures of common landmarks (C. Fails), we can clearly see the superior performance of our method over the Mix.Tree and the DRMF method, except the DRMF method performs best in terms of C. Fails on the HELEN. The failures of our method on the LFPW are just 10% of Mix.Tree and half of DRMF, which is a very significant improvement. Our method has comparable performance to the SDM [9]. The SDM has fewer failures while our method performs slightly better in terms of mean error on both databases. We also note the models used by these three existing methods were all trained on a large number of highly reliable face images from the Muli-PIE face database while the model of our method was trained on only a few thousands of face images. It shows superior or comparable performance when compared to the existing state of the art methods. One example image from each dataset is shown in Fig. 8. As can be seem, the Mix.Tree and DRMF both have difficulty to deal with the subtle variations caused by the facial expression (in the second row) and abnormal appearance (the eyes in the first row). On the contrary, SDM and our method localize the eye corners and mouth corners very accurately despite facial expression and occlusion is presented. Our method also localizes the even more difficult landmarks along the face contour very accurately.

VI. Conclusion

In this work we propose a Regression Forests based Cascaded Pose Regression method for face alignment. We show the efficiency of using regression forests as the primitive regressor in the CPR framework. We propose an intelligent initialization scheme that is able to select a few reliable pose estimations in a few stages in the cascade and aggregate them to the remaining cascade to calculate the final pose. By using the proposed method, we have achieved performance superior or close to the state of the art. Furthermore, we have shown that through using different features, there is a slight improvement of performance at a cost of feature computation. Also, as the boosting method is sensitive to the face detection, our method to some degree is capable of decreasing the risk by using the intelligent initialization scheme. The initialization of CPR is an interesting problem

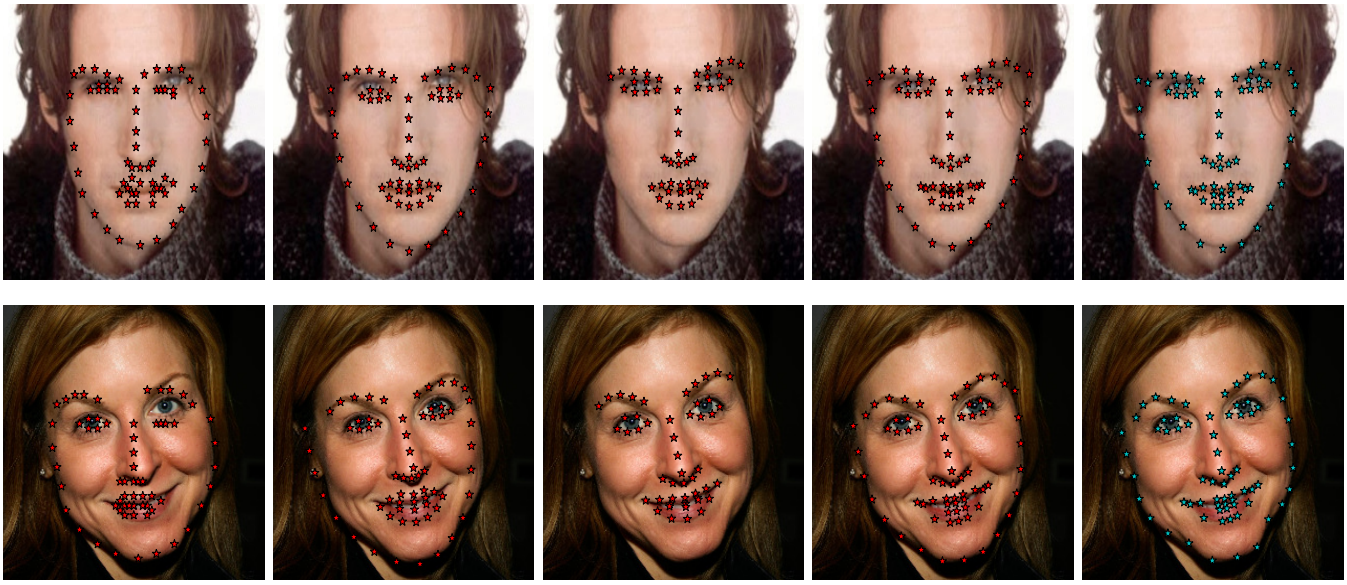


Fig. 8: Example results of different methods on LFPW (the first row) and on HELEN (the second row).

needs to be further investigated in our future work.

REFERENCES

[1] Li, S.Z., Jain, A.K.: 'Handbook of face recognition'. (Springer, 2011, 1st edn.)

[2] He, Z., Tan, T., Sun, Z., Qiu, X.: 'Toward accurate and fast iris segmentation for iris biometrics', IEEE Trans. Pattern Analysis and Machine Intelligence, 2009, 31, (9), pp. 1670–1684

[3] Suhr, J.K., Eum, S., Jung, H.G., Li, G., Kim, G., Kim, J.: 'Recognizability assessment of facial images for automated teller machine applications'. Pattern Recognition, 2012, 45, (5), pp. 1899–1914

[4] Jiang, X., Mandal, B., Kot, A.: 'Eigenfeature regularization and extraction in face recognition'. IEEE Trans. Pattern Analysis and Machine Intelligence, 2008, 30, (3), pp. 383–394

[5] Li, H., Yu, J., Ye, Y., Bregler, C.: 'Realtime facial animation with on-the-fly correctives'. ACM Transactions on Graphics, 2013, 32, (4), pp. 35–42

[6] Lee, Y.J., Lee, S.J., Park, K.R., Jo, J., Kim, J.: 'Single view-based 3d face reconstruction robust to self-occlusion'. EURASIP Journal on Advances in Signal Processing, 2012, pp. 1–20

[7] Cristinacce, D., Cootes, T.: 'Feature detection and tracking with constrained local models'. Proc. British Machine Vision Conference, 2006.

[8] Cao, X., Wei, Y., Wen, F., Sun, J.: 'Face alignment by explicit shape regression'. Proc. IEEE Conf. Computer Vision and Pattern Recognition, 2012.

[9] Xiong, X., De la Torre, F.: 'Supervised descent method and its applications to face alignment'. Proc. IEEE Conf. Computer Vision and Pattern Recognition, 2013.

[10] Cootes, T.F., Edwards, G.J., Taylor, C.J.: 'Active appearance models'. IEEE Trans. Pattern Analysis and Machine Intelligence, 2001, 23, (6) , pp. 681–685

[11] Dollár, P., Welinder, P., Perona, P.: 'Cascaded pose regression'. Proc. IEEE Conf. Computer Vision and Pattern Recognition, 2010.

[12] Efraty, B., Huang, C., Shah, S.K., Kakadiaris, I.A.: 'Facial landmark detection in uncontrolled conditions'. Proc. Int'l Joint Conference on Biometrics, 2011.

[13] Vukadinovic, D., Pantic, M.: 'Fully automatic facial feature point detection using Gabor feature based boosted classifiers'. Proc. IEEE Int'l Conf. Systems, Man and Cybernetics, 2005.

[14] Liao, C.T., Wu, Y.K., Lai, S.H.: 'Locating facial feature points using support vector machines'. International Workshop on Cellular Neural Networks and Their Applications, 2005.

[15] Rapp, V., Senechal, T., Bailly, K., Prevost, L.: 'Multiple kernel learning svm and statistical validation for facial landmark detection'. Proc. IEEE Int'l Conf. on Automatic Face and Gesture Recognition, 2011.

[16] Du, C., Wu, Q., Yang, J., Wu, Z.: 'SVM based ASM for facial landmarks location'. Proc. IEEE Int'l Conf. Computer and Information Technology, 2008.

[17] Belhumeur, P.N., Jacobs, D.W., Kriegman, D.J., Kumar, N.: 'Localizing parts of faces using a consensus of exemplars'. Proc. IEEE Conf. Computer Vision and Pattern Recognition, 2011.

[18] Valstar, M., Martinez, B., Binefa, X., Pantic, M.: 'Facial point detection using boosted regression and graph models'. Proc. IEEE Conf. Computer Vision and Pattern Recognition, 2010.

[19] Zhu, X., Ramanan, D.: 'Face detection, pose estimation and landmark localization in the wild'. Proc. IEEE Conf. Computer Vision and Pattern Recognition, 2012.

[20] Boddeti, V.N., Kanade, T., Kumar, B.V.: 'Correlation filters for object alignment'. Proc. IEEE Conf. Computer Vision and Pattern Recognition, 2013.

[21] Cristinacce, D., Cootes, T.: 'Boosted regression active shape models'. Proc. British Machine Vision Conference, 2007, pp. 880–889

[22] Martinez, B., Valstar, M., Binefa, X., Pantic, M.: 'Local Evidence Aggregation for Regression Based Facial Point Detection'. IEEE Trans. Pattern Analysis and Machine Intelligence, 2012.

[23] Dantone, M., Gall, J., Fanelli, G., Van Gool, L.: 'Real-time facial feature detection using conditional regression forests'. Proc. IEEE Conf. Computer Vision and Pattern Recognition, 2012.

[24] Amberg, B., Vetter, T.: 'Optimal landmark detection using shape models and branch and bound'. Proc. IEEE Int'l Conf. Computer Vision, 2011.

[25] Saragih, J.M., Lucey, S., Cohn, J.F.: 'Face alignment through subspace constrained mean-shifts'. Proc. IEEE Int'l Conf. Computer Vision, 2009.

[26] Yang, H., Patras, I.: 'Face parts localization using structured-output regression forests'. Proc. Asian Conf. Computer Vision, 2012.

[27] Tan, X., Song, F., Zhou, Z.H., Chen, S.: 'Enhanced pictorial structures for precise eye localization under incontrolled conditions'. Proc. IEEE Conf. Computer Vision and Pattern Recognition, 2009.

[28] Wu, Y., Wang, Z., Ji, Q.: 'Facial feature tracking under varying facial expressions and face poses based on restricted boltzmann machines'. Proc. IEEE Conf. Computer Vision and Pattern Recognition, 2013.

[29] Zhou, F., Brandt, J., Lin, Z.: 'Exemplar-based graph matching for robust facial landmark localization'. Proc. IEEE Intl Conf. Computer Vision, 2013.

[30] Yang, H., Patras, I.: 'Sieving regression forests votes for facial feature detection in the wild'. Proc. Int'l Conf. Computer Vision, 2013.

[31] Saragih, J., Goecke, R.: 'A nonlinear discriminative approach to AAM fitting'. Proc. IEEE Conf. Computer Vision, 2007.

[32] Tresadern, P.A., Sauer, P., Cootes, T.F.: 'Additive update predictors in active appearance models'. Proc. British Machine Vision Conference, 2010.

[33] Tzimiropoulos, G., Pantic, M.: 'Optimization problems for fast AAM fitting in-the-wild'. Proc. IEEE Intl Conf. Computer Vision, 2013.

[34] Burgos-Artizzu, X.P., Perona, P., Dollár, P.: 'Robust face landmark estimation under occlusion'. Proc. IEEE Conf. Computer Vision, 2013.

[35] Jesorsky, O., Kirchberg, K., Frischholz, R.: 'Robust face detection using the hausdorff distance'. Proc. Audio-and Video-Based Biometric Person Authentication, Springer, 2001.

[36] Messer, K., Matas, J., Kittler, J., Luettin, J., Maitre, G.: 'Xm2vtsdb: The extended m2vts database'. Proc. Second international conference on audio and video-based biometric person authentication, 1999, pp. 965–966

[37] Kasinski, A., Florek, A., Schmidt, A.: 'The PUT face database'. Image Processing and Communications, 2008, 13, (3-4), pp. 59–64

[38] Sun, Y., Wang, X., Tang, X.: 'Deep convolutional network cascade for facial point detection'. Proc. IEEE Conf. Computer Vision and Pattern Recognition, 2013.

[39] Sagonas, C., Tzimiropoulos, G., Zafeiriou, S., Pantic, M.: 'A semi-automatic methodology for facial landmark annotation'. Proc. IEEE Conf. Computer Vision and Pattern Recognition Workshops (CVPRW), 2013.

[40] Kostinger, M., Wohlhart, P., Roth, P.M., Bischof, H.: 'Annotated Facial Landmarks in the Wild: A Large-scale, Real-world Database for Facial Landmark Localization'. Proc. IEEE Int'l Conf. Computer Vision Workshops. 2011, pp. 2144–2151

[41] Cootes, T.F., Ionita, M.C., P., S.: 'Robust and Accurate Shape Model Fitting using Random Forest Regression Voting'. Proc. European Conf. Computer Vision, 2012.

[42] Le, V., Brandt, J., Lin, Z., Bourdev, L., Huang, T.S.: 'Interactive facial feature localization'. Proc. European Conf. Computer Vision, 2012.

[43] Yang, H., Patras, I.: 'Privileged information-based conditional regression forests for facial feature detection'. Proc. IEEE Int'l Conf. on Automatic Face and Gesture Recognition, 2013.

[44] Viola, P., Jones, M.: 'Rapid object detection using a boosted cascade of simple features'. Proc. IEEE Conf. Computer Vision and Pattern Recognition, 2001.

[45] Asthana, A., Zafeiriou, S., Cheng, S., Pantic, M.: 'Robust discriminative response map fitting with constrained local models'. Proc. IEEE Conf. Computer Vision and Pattern Recognition, 2013.

[46] Gross, R., Matthews, I., Cohn, J., Kanade, T., Baker, S.: 'Multi-PIE'. Image and Vision Computing, 2010, 28, (5), pp. 807–813