Facial Landmark Detection via Progressive Initialization

Shengtao Xiao Shuicheng Yan Ashraf A. Kassim
Department of Electrical and Computer Engineering, National University of Singapore
Singapore 117576
xiao_shengtao@u.nus.edu, eleyans@nus.edu.sg, ashraf@nus.edu.sg

Abstract

In this paper, we present a multi-stage regression-based approach for the 300 Videos in-the-Wild (300-VW) Challenge, which progressively initializes the shape from obvious landmarks with strong semantic meanings, e.g. eyes and mouth corners, to landmarks on face contour, eyebrows and nose bridge which have more challenging features. Compared with initialization based on mean shape and multiple random shapes, our proposed progressive initialization can very robustly handle challenging poses. It also guarantees an accurate landmark localization result and shows smooth tracking performance in real-time.

1. Introduction

Face alignment plays a very important role in many computer vision research topics and applications, such as face recognition [20], face synthesis/morphing [11, 9], face detection [6], and 3D face modeling [4]. The accuracy of landmark position localization can directly influence the final performance of these applications. Though many efforts [5, 18, 8] have been devoted during the past decades, facial landmark localization is still very challenging in the cases of large pose variations, abrupt illumination changes, extreme facial expressions and heavy occlusion.

Regression-based facial points detection methods [3, 18, 5] have become very popular recently. Such models basically consist of two steps, feature extraction and regression. Features are extracted around the predicted shape at each cascade stage. The shape update for the current prediction can be easily generated by projecting the features to a learned regression matrix. After a few steps, the estimation error can converge to an arbitrarily small error. A major reason behind the popularity of the regression-based methods is their efficiency. Compared with other methods, e.g. Deformable Part Model [14, 22] and Active Appearance Models [17, 12], the regression-based models can be much faster [5, 18, 13].

Though regression-based models are fast and accurate in most cases, landmark detection is still challenging for faces with large poses and expressions. One major limitation of recent model-based regression methods [5, 18, 13] is that they may be easily trapped by a local optimum if the starting shape is far away from the ground-truth shape. Multiple random initializations [5] can improve the localization performance for simple cases. For faces with large pose, poor illumination or a large expression, multiple random initializations cannot guarantee good performance [21]. Burgos et al. proposed smart initialization in [3] where the set of initial shapes is stopped if its variance of regressed shapes is large and a new set of shapes is randomly selected for regression initialization. While this enables selection of good starting points, but too many uncertainties are introduced due to randomness. The performance enhancement due to smart restart is limited as reported in [3].

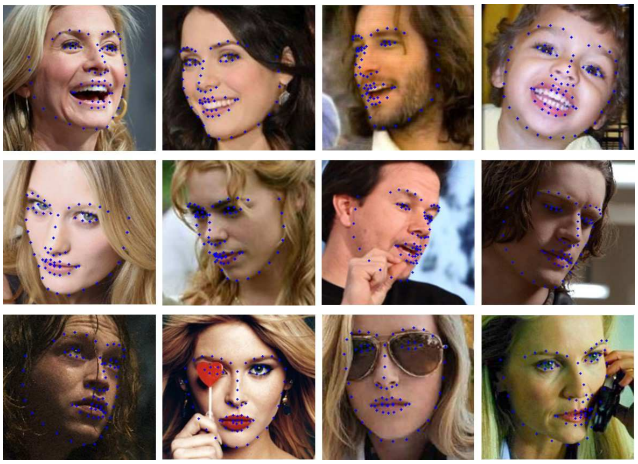

Figure 1. Some landmark localization results from challenging face images collected from the Web. Top: faces with expressions; middle: faces with large poses; bottom: faces under occlusion.

In this paper, we present progressive initialization for the facial landmark detection and tracking. The work is based on the observation that obvious points which have very strong discriminative features, e.g., eyes and mouth corners, are usually positioned first with strong confidence

when we manually annotate these landmarks. Landmarks on the face contour, nose bridge and eyebrows, which are more challenging to be positioned, are localized at last usually with reference to the early defined points. This is actually a simple but very efficient strategy which guides the annotating process and ensures the great efficiency even if the to-be-processed face image is under a challenging condition. These points can be roughly inferred with reference to the early-positioned points. Zhang et al. [19] proposed a deep Convolutional Neural Network-based framework to learn a model which solves multiple objectives, i.e., regression of five facial landmarks, smile detection and sunglasses detection. They have validated that the five facial points detected by their approach can be used for more robust initialization and improve the landmark detection results for more points. Motivated by their findings, we propose to progressively initialize the face shape from a few easy landmarks to challenging landmarks. The major limitation of [19], which is its heavy computational resource requirement, is solved with the popular cascaded regression tree model. Fig. 1 shows some landmark detection results for a few selected face images under challenging conditions, e.g. expressions, poses and occlusion.

The main contributions of this paper are summarized as follows:

- An efficient face tracking approach is proposed.
- A simple but efficient facial landmark tracking approach which guarantees real-time performance in an unconstrained environment is presented.
- Progressive initialization is proposed to ensure a robust initialization for landmark detection of 68 points.

The remainder of the paper is as follows. We provide a review of related work in Section 2 before introducing our progressive initialization algorithm in Section 3. The experimental results are presented and discussed in Section 4 before we conclude the paper in Section 5.

2. Related Work

Regression-based models have been very successful and popular in recent research. In this section, the cascaded regression tree and the local binary feature are briefly reviewed. More details can be found in [18, 5, 13, 3].

2.1. Cascaded Regression Trees

Cascaded regression trees formulate the shape regression for L landmarks into an additive cascade form as follows:

$$\hat{s}^k = \hat{s}^{k-1} + \mathcal{M}^k(\Phi^k(I, \hat{s}^{k-1})), \tag{1}$$

where $\hat{s}^k \in R^{2L\times 1}$ and $\Phi^k(\cdot)$ are the predicted shape and the feature extractor at the k-th cascade stage accordingly. $\mathcal{M}^k(\cdot)$ represents the mapping function which projects the feature extracted from the image I at the predicted shape $\hat{s}^{k-1}$ to the target shape. For cascaded regression trees, this mapping function can be formulated as

$$\mathcal{M}^k(\Phi^k(I, \hat{s}^{k-1})) = \sum_{t=1}^{T} f_t^k(\phi_t^k(I, \hat{s}^{k-1})), \tag{2}$$

where T is the number of trees in the current stage and $f_t^k(\cdot)$ generates shape increment with the given image I and the estimated shape $\hat{s}^{k-1}$. $\phi_t^k(\cdot)$ is the feature extracted from the t-th regression tree in the k-th cascade stage. For a given input image I and a current shape estimation $\hat{s}^{k-1}$, after passing a few node split tests from the t-th regression tree with L_f leaf nodes, a leaf node with index $l_f \in \{1, 2, ..., L_f\}$ will be reached and the corresponding regression output $r_{t,l_f}^k = f_t^k(\phi_t^k(I, \hat{s}^{k-1}))$ is generated. The feature extracted is simply an L_f bits binary value with the l_f-th bit being 1 and the rest bits being 0.

For clarification purposes, the cascade regression process of shape prediction can be further defined as

$$\hat{s}^K = \boldsymbol{\mathcal{M}}^K(I, s^0) \tag{3}$$

where $\boldsymbol{\mathcal{M}}^K$ represents a K-stage cascade regression which takes input image I and initial shape s^0 and generates output $\hat{s}^K$.

2.2. Local Binary Feature [13]

To learn the structure of the regression trees, Ren et al. [13] divided the process into two steps: 1) learn tree structures locally and 2) learn tree output globally.

Consider a global feature $\Phi^k = [\phi_1^k, \phi_2^k, ..., \phi_L^k]$ where ϕ_x^k is the local feature for the x-th landmark. $\phi_x^k = [\phi_{1,x}^k, \phi_{2,x}^k, ..., \phi_{T_x,x}^k]$ where T_x is the number of regression trees trained for the landmark x. For T_x trees, the structure of the tree is learned locally via optimizing

$$\min_{w_x^k, \phi_x^k} \sum_{i=1}^{M} \|\pi_x \circ \tilde{s}_i^k - w_x^k \phi_x^k\|_2^2, \tag{4}$$

where the global regression target $\tilde{s}_i^k$ is defined as $\tilde{s}_i^k = s_i^* - \hat{s}_i^k$, and π_x is an operator which extracts the regression target of the x-th landmark. The node split is trained by maximizing the reduced variance of local regression targets, $\pi_x \circ \tilde{s}^k$, from all training samples passed into the node.

When the structures of regression trees are fixed, the tree output is learned jointly by solving the regression problem similar as [18] but with a regularization term to prevent overfitting as stated below

$$\min_{W^k} \sum_{i=1}^{M} \|\tilde{s}_i^k - W^k \Phi^k\|_2^2 + \lambda_k \|W^k\|_2^2, \tag{5}$$

where M is the number of training samples, λ_k is the regularization term at the k-th stage and $W^k \in R^{2L \times T_D}$ is the regression matrix with T_D being the number of leaf nodes of all regression trees within k-th cascade stage. The leaf node output is simply given by the corresponding column vector from the regression matrix W^k.

3. Progressive Initialization

In this section, we introduce the components of our approach, including progressive initialization for both training and testing. Some techniques used for robust and efficient landmark tracking are also presented. The notations used in the rest of the paper are given in Table 1.

Table 1. Notations used in this paper

Category	Notation	Meaning
Scalars	K	cascade stages of a model
	M	number of initializations used
Vectors	I	image window
	S_n	n points face shape
	$\mathcal{N}_S$	transformation matrix of S to mean shape space
	$\mathbf{D}$	distances between prior shape and K-mean centers
	$\hat{\mathbf{S}}^0$	initial shapes $\hat{\mathbf{S}}^0 = \{\hat{S}^{0,1}, ..., \hat{S}^{0,N}\}$
Functions	$\Phi_n(I, S_n)$	feature extraction for S_n at I
	$\mathcal{M}_n(I, \Phi_n)$	mapping function for n points shape prediction with model $\boldsymbol{\mathcal{M}}_n$ and starting shape S_n^0
	$\boldsymbol{\mathcal{M}}_n^K(I, S_n^0)$	
Symbols	$\hat{x}$	value/vector estimated
	x^*	ground-truth value/vector

3.1. Normalized K-means Shape Centers for Initialization

All training shapes are first normalized by similarity transformation where a training shape is aligned to the mean shape to minimize their L_2 distance given by

$$\mathcal{N}_S = \arg_{\mathcal{N}} \min \|\bar{S} - \mathcal{N} \circ S\|_2, \tag{6}$$

where $\bar{S}$ is the mean shape and $\mathcal{N}$ consists of rotation and scaling operations. The normalized shapes are then defined by $S_{\mathcal{N}} = \mathcal{N}_S \circ S$. K-means shape centers are formed with the normalized shapes. Fig. 3 and Fig. 4 show randomly selected K-means centers of normalized training shapes for 19 points and 68 points respectively. From these two figures, we observe that all centers have a rotation angle of about zero. This is because all training samples undergo a similarity transformation, i.e., Eqn. (6). More details will be given on how to preserve the rotation information later.

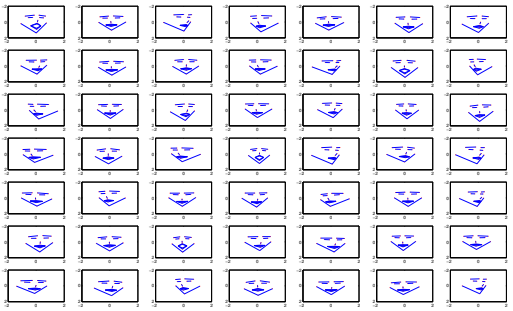

Figure 3. K-means centers, $\mathbf{S}_{\mathcal{N}19}$, for normalized 19-points shapes in the mean shape space. There are 191 K-means centers in total and 49 K-means centers are randomly selected and shown here.

Figure 4. K-means centers, $\mathbf{S}_{\mathcal{N}68}$, for normalized 68-points shapes in the mean shape space. There are 681 K-means centers in total and 49 K-means centers are randomly selected and shown here.

3.2. Guided Initialization with Prior Shape

A prior shape is a predicted face shape but with less points. For instance, $\hat{S}_5$ is the prior shape for 19P shape prediction and $\hat{S}_{19}$ is the prior shape for 68P shape prediction. Prior shapes provide essential information for initial shape selection. For instance, to select initial shapes for 19P shape prediction, we first calculate the L_2 distance from the normalized prior shape $\mathcal{N}_{S_5} \circ S_5$ to the 19P K-means centers, i.e.,

$$D_i = \|\mathcal{P}_{19\to5} \circ S^i_{\mathcal{N}19} - \mathcal{N}_{S_5} \circ S_5\|_2 \tag{7}$$

where $\mathbf{S}_{\mathcal{N}19} = \{S^1_{\mathcal{N}19}, S^2_{\mathcal{N}19}, ..., S^{N_{19}}_{\mathcal{N}19}\}$ denotes the 19P K-means centers with N_{19} being the number of centers. $\mathcal{P}_{19\to5}$ extracts the 5P landmarks from 19P in a way that $\mathcal{P}_{19\to5} \circ S^i_{\mathcal{N}19}$ and the prior shape S_5 are within the same landmark space.

The corresponding initial shape for the i-th K-means center is defined as

$$S^{0,i} = \mathcal{N}^{-1}_{S_5} \circ S^i_{\mathcal{N}19}. \tag{8}$$

Since the distance from the prior shape to K-mean centers is known, i.e., $D_i,\ i = \{1, 2, ..., N_{19}\}$, initial shapes

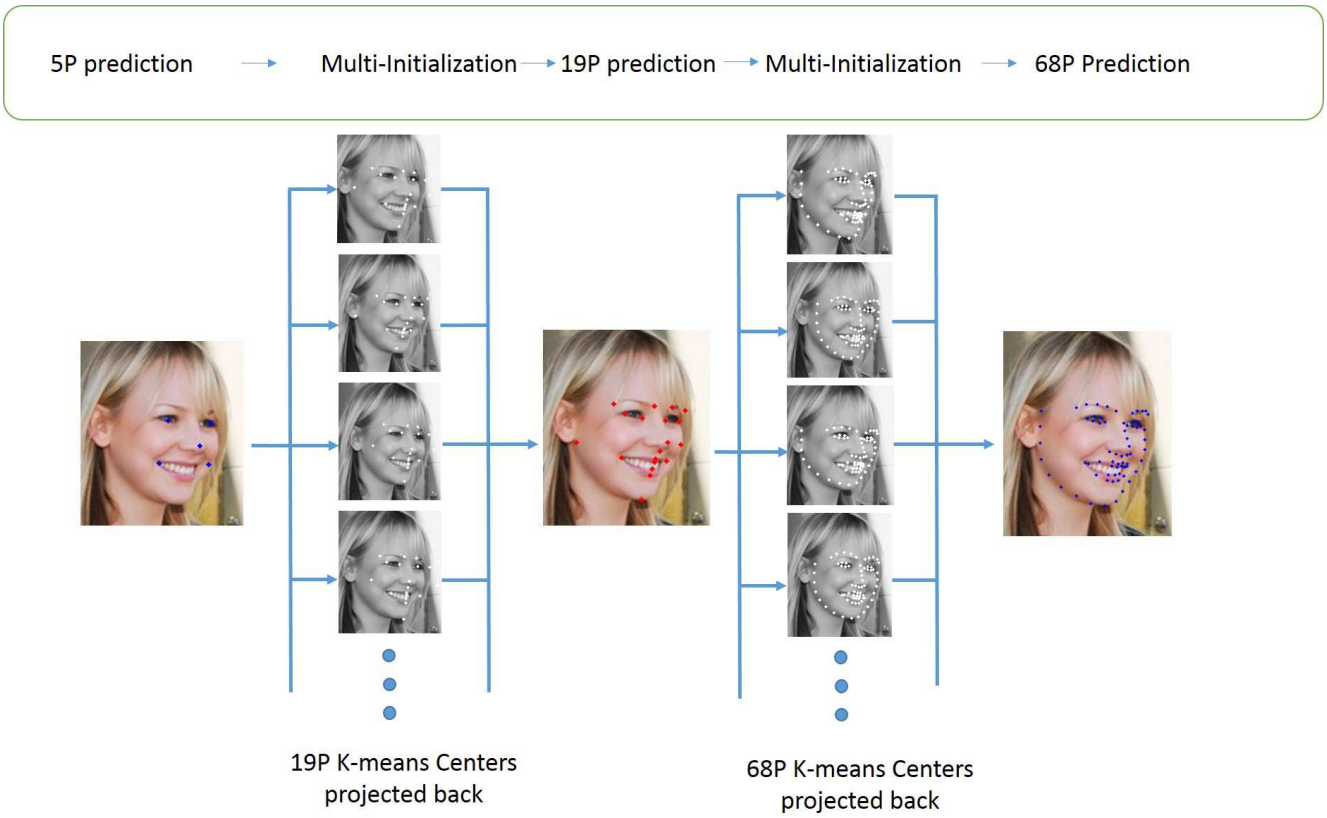

Figure 2. The framework of our progressive initialization consists of three stages. Each stage consists of a shape regressor. Three regressors are trained progressively which predict shapes with 5, 19 and 68 points. The 5 points with strong features, i.e., eyes, mouth corners and nose tip, are predicted first. This predicted 5P shape is then used to guide the initialization process of the 19P face shape predictor. Similarly, the 68P shape predictor uses the predicted 19P shape as reference to select initial shapes.

which are close to the prior shapes can be easily selected by sorting $\mathbf{D} = \{D_1, D_2, ..., D_{N_{19}}\}$. Compared to [21], the computational resources required are much lower for searching good initial shapes, as only the distances between the prior shape and a limited number of K-means centers are required. Rotation information can still be preserved by directly applying inverse similarity transformation, i.e., $\mathcal{N}_{S_5}^{-1}$, to the selected K-means centers.

The general initial shape selection process can be summarized in Algorithm 1.

3.3. Perturbed Training for Robustness

Robustness of landmark tracking can be achieved in the training process. The key factors such as bounding box variation and bad prior shape initialization are handled explicitly in the training process.

3.3.1 Bounding Box Robustness

To ensure bounding box robustness, bounding box augmentation is performed when training the 5P shape predictor. The center of the detected bounding box is perturbed with

Algorithm 1 Initial Shape Selection from K-means Centers

Input: Prior shape detected: $\hat{S}_p$. K-means face centers: $\mathbf{S}_{\mathcal{N}c} = \{S^1_{\mathcal{N}c}, S^2_{\mathcal{N}c}, ..., S^{N_c}_{\mathcal{N}c}\}$. Number of K-means centers: N_c. Number of shapes to be selected: x; Number of landmarks used in $\mathbf{S}_{\mathcal{N}c}$: c. Number of landmarks used in $\hat{S}_p$: p.
 # center shape for next stage has more points than prior shape, i.e, centers $\mathbf{S}_{\mathcal{N}19}$ *with prior* $\hat{S}_5$
Output: Initial shapes for next stage shape prediction: $\hat{\mathbf{S}}^0_c = \{\hat{S}^{0,1}_c, \hat{S}^{0,2}_c, ..., \hat{S}^{0,x}_c\}$
Get $\mathcal{N}_{\hat{S}_p}$ with Eqn. (6)
for i=1:$\mathbf{S}_{\mathcal{N}c}$.length()
  $\mathbf{D}[i] = \|\mathcal{P}_{c\to p} \circ \mathbf{S}^i_{\mathcal{N}c} - \mathcal{N}_{\hat{S}_p} \circ \hat{S}_p\|_2$
end
Sort $\mathbf{D}$ to get a vector of distance order in ascending form: I_{dx}
for i=1:x
  $\hat{S}^{0,i}_c = \mathcal{N}^{-1}_{S_p} \circ S^{I_{dx}[i]}_{\mathcal{N}c}$ # reverse transformation
end

a uniformly distributed random translational offset within 10% of the interocular distance calculated from the ground-truth shape. A scaling disturbance is also applied to the bounding box.

3.3.2 Initialization Robustness

Each prior shape provides initialization guidance for the face shape predictor of the next stage. In the training process, initial shapes are selected in a way that distant K-mean centers can also be selected. This mimics the situation when prior face landmark detection fails and a bad K-mean center is selected as the starting shape. Algorithm 2 is used to select centers to train a 19P predictor from the 5P shape prior.

Algorithm 2 K-means Centers Selection for Shape Initialization in Training Process

Input: Prior shape detected: $\hat{S}_5$. 19P K-means centers: $\mathbf{S}_{\mathcal{N}19} = \{S^1_{\mathcal{N}19}, S^2_{\mathcal{N}19}, ..., S^{N_{19}}_{\mathcal{N}19}\}$. Number of K-means centers in $\mathbf{S}_{\mathcal{N}19}$: N_{19}. Number of initializations for training: x. K-means center selection sampling rate: r.
Output: K-means centers selected: $\mathbf{S}_K = \{S^1_K, S^2_K, ..., S^x_K\}$
Get $\mathcal{N}_{\hat{S}_5}$ with Eqn. (6)
for i=1:N_{19}
  $\mathbf{D}[i] = \|\mathcal{P}_{19\to5} \circ \mathbf{S}^i_{\mathcal{N}19} - \mathcal{N}_{\hat{S}_5} \circ \hat{S}_5\|_2$
end
Sort $\mathbf{D}$ to get a vector of distance order in ascending form: I_{dx}
for i=1:x
  p=$rand > 0.1$ # random number
  if p
    $S^i_K = S^{I_{dx}[(i-1)r+1]}_{\mathcal{N}19}$
  else
    $S^i_K = S^{I_{dx}[N_{19}-(i-1)r]}_{\mathcal{N}19}$
end
i-th initial shape: $S^{0,i}_{19} = \mathcal{N}^{-1}_{S_5} \circ S^i_K$

Similar steps are taken to generate the initial shapes for the 68P predictor in the training process. We use 68P K-means centers $\mathbf{S}_{\mathcal{N}_{68}}$ and sort the calculated distances between the prior shape $\hat{S}_{19}$ and $\mathbf{S}_{\mathcal{N}_{68}}$ when selecting the initial shapes in the training process.

After initializations are fixed, LBF [13] is used to train the models for 5P, 19P and 68P shape regressors progressively. In the testing process, the top M closest K-means centers are selected for initialization if multiple initializations are used. The steps are shown in Algorithm 1.

3.4. Shape Prediction via Progressive Initialization

Our approach consists of three stages which predict shapes with 5, 19 and 68 points (the points selected are manually defined at each stage) in a cascaded way. Fig. 2 shows the framework of the proposed approach. The 5P predictor first locates the 5 key points, including eyes, mouth corners and nose tip. These 5 key points are chosen as they have obvious features and used by most face detectors to identify a face. They are relatively more robust to face detectors as compared to other landmarks, e.g. eyebrows and chin. The predicted 5P face shape acts as the prior shape for the 19P landmark detector to help choose initial shapes from the K-means centers. The shape estimated from the 19P predictor then guides the initial shapes selection for the 68P shape regressor. These steps can be summarized as Algorithm 3 in the testing process.

Algorithm 3 Shape Prediction via Progressive Initialization

Input: Image: I; Bounding box: B; models: $\boldsymbol{\mathcal{M}}^{K_5}_5$, $\boldsymbol{\mathcal{M}}^{K_{19}}_{19}$, $\boldsymbol{\mathcal{M}}^{K_{68}}_{68}$; K-means shape centers: $\mathbf{S}_{\mathcal{N}19}$, $\mathbf{S}_{\mathcal{N}68}$; Previous Estimation: $\hat{S}_{68,p}$; Number of initial shapes for 5P, 19P and 68P: $x_5,\ x_{19},\ x_{68}$;
if B is from face detector
  $\hat{S}^0_5 = \bar{S}_5$
  $\hat{S}^{K_5}_5 = \boldsymbol{\mathcal{M}}^{K_5}_5(I, \hat{S}^0_5)$
else # *estimated bounding box from last frame*
  Select initial shapes $\hat{\mathbf{S}}^0_5$ from $\mathbf{S}_{\mathcal{N}19}$ based on $\hat{S}_{68,p}$
  $\hat{S}^{K_5}_5 = \frac{1}{x_5}\sum_i^{x_5} \boldsymbol{\mathcal{M}}^{K_5}_5(I, \hat{S}^{0,i}_5)$
Given $\hat{S}^{K_5}_5$, select initial shapes $\hat{\mathbf{S}}^0_{19}$ with algorithm 1
19P prediction: $\hat{S}^{K_{19}}_{19} = \frac{1}{x_{19}}\sum_i^{x_{19}} \boldsymbol{\mathcal{M}}^{K_{19}}_{19}(I, \hat{S}^{0,i}_{19})$
Given $\hat{S}^{K_{19}}_{19}$, select initial shapes $\hat{\mathbf{S}}^0_{68}$ with algorithm 1
68P prediction: $\hat{S}^{K_{68}}_{68} = \frac{1}{x_{68}}\sum_i^{x_{68}} \boldsymbol{\mathcal{M}}^{K_{68}}_{68}(I, \hat{S}^{0,i}_{68})$
end

3.5. Face Detection and Tracking

Although the processing speed of the recent face landmark detection algorithms can be less than 1ms [13], the speed for processing face alignment in videos can hardly reach 1ms in practice. There are a few key factors which limit the face landmark tracking in video sequences, including the video decoding speed and the face detection speed. Loading all frames into memory and processing individual frames are hardly practical due to the huge amount of memory consumption for decoding and storing the decoded video frames. Popular existing OpenCV face detectors, e.g., Viola-Jones-based, can achieve almost-real-time face detection on a normal PC. However, the detection performance deteriorates significantly for faces with large poses, slight occlusion and bad light illumination. Yu et al. [1] developed a face detection library which guarantees a satisfactory detection speed and accuracy for challenging cases. How-

ever, the speed decreases dramatically for faces under challenging conditions. To ensure efficiency, face tracking is essential as the computational resources required for face detection make real-time facial landmark tracking almost impossible with most existing algorithms.

Since our framework can easily regress to a very stable face shape, we can use this face shape to estimate a bounding box. The estimated bounding box can then be used as the face bounding box for the next frame under the assumption that the face moves within an arbitrary range. However, there are still cases where landmark tracking fails due to heavy occlusion and scenery boundary. Inspired by smart restart [3], a similar scheme is applied to trigger the face detector when the variance of regression results exceeds a pre-defined threshold. This successfully prevents error from accumulating due to bad bounding box estimation. Meanwhile, the computational resources required for face detection can be significantly reduced.

4. Experiment and Result

Our models are trained with the 300W dataset (LFPW, HELEN, and AFW) and the randomly selected 10% frames from the training videos provided by the event organizer. Standard evaluation measures are adopted in this event. The error of each frame can be calculated as follows:

$$e_i = \frac{\|S^* - \hat{S}\|}{D_i}, \tag{9}$$

where D_i is the inter-ocular distance determined by outer corners of left and right eyes, i.e. the 37-th and the 46-th point.

4.1. Results on 300-VW Testing Set

Our approach has been evaluated independently by the 300-VW Challenge [15] organizers using their own testing videos which are not disclosed to the participants. The details of annotation process for the training and testing set are presented in [7, 16]. There are 150 testing videos which are divided into three scenarios. Scenario 1 consists of videos taken under well-lit conditions; Scenario 2 contains videos taken in unconstrained conditions without heavy occlusion; Scenario 3 consists of videos recorded under completely unconstrained conditions. The 49 points (excluding points from the face contour) errors are returned together with the baseline performance [2]. For 68 points errors, only the performance of our approach is provided.

Fig. 5 shows that our approach yields much better performance in facial landmark tracking for all categories when compared with the baseline [2]. More than 90% of the testing frames are within 8% point-to-point error for all categories. Our approach has outperformed the baseline for more than 20% in each category.

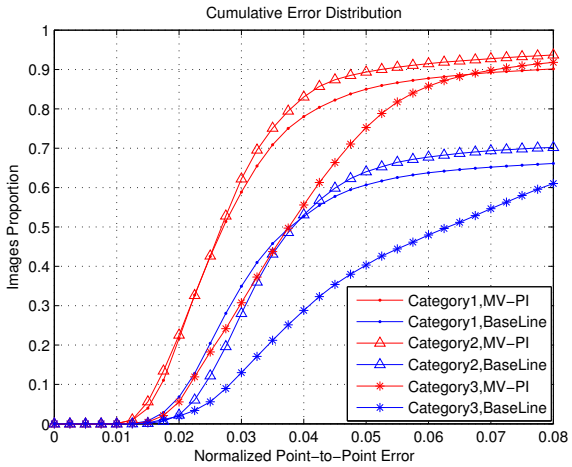

Figure 5. Comparison of facial landmark tracking performance on 49 points between our approach and the baseline [2] in all three scenarios.

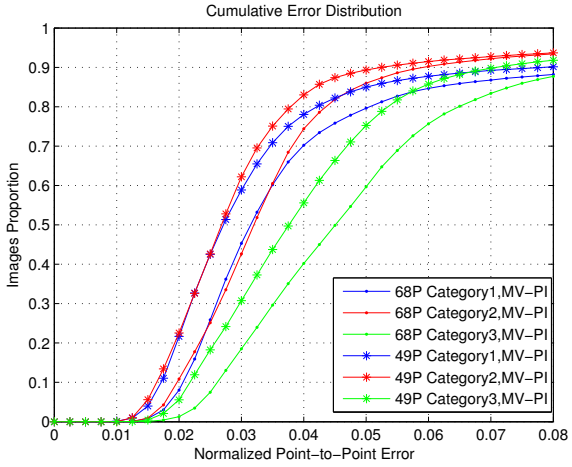

Figure 6. Results of 68 points and 49 points tracking in all three scenarios.

Fig. 6 shows that the performance of 68 points tracking degrades about $5\% \sim 10\%$ as compared to that of 48 points tracking. This is reasonable as landmarks on the face contour are challenging points with less discriminative features as compared to facial feature landmarks (49 Points). With the developed framework, which gradually determines these challenging landmarks with reference to the per-estimated inner facial landmarks, our 68 point landmark tracking performance is very promising as well.

4.2. Results on Real-time Video

Further evaluation is done to verify our framework on the challenging Youtube Celebrities Database [10] which contains videos of celebrities captured in the wild. Some frames of challenging poses and expressions are shown in Fig. 7. We notice that even in the cases of extreme poses,

our approach can still robustly track the landmarks without using the face detector which is normally computationally expensive. The face detection process will be triggered (e.g. Fig. 7 Row-3 to Row-4) only when the variance (Sec. 3.5) is too large. This mechanism can efficiently help us to prevent error accumulation.

Our approach is computationally efficient. Without much code optimization, our implementation can reach 30+ FPS for landmark tracking on a single core E5-1603 CPU. The speed may be further improved via code optimization and parallel computation. In fact, the current version can be used in most real-time face-related applications.

5. Conclusion

In this paper, we introduced a facial landmark tracking framework which progressively initializes and predicts the face shapes for the 300-VW competition. The proposed method locates simple and easy-to-detect facial landmarks first which are then used to guide the initial shapes selection process for the regressor in later stages. To ensure overall landmark tracking efficiency, our efficient face tracking approach uses the bounding box estimated by landmark prediction from the previous frame. Our method showed significant improvement over the baseline in all three testing scenarios and its real-time performance also makes it possible to be used in many face-related applications.

Currently, the shape at each stage is fixed and manually determined. In the future, a framework which automatically and gradually infers the locations of landmarks from the most obvious landmarks to the most difficult landmarks is to be developed.

References

[1] libfacedetection. `https://github.com/ShiqiYu/libfacedetection`.

[2] A. Asthana, S. Zafeiriou, S. Cheng, and M. Pantic. Incremental face alignment in the wild. In *Computer Vision and Pattern Recognition (CVPR), 2014 IEEE Conference on*, pages 1859–1866. IEEE, 2014.

[3] X. P. Burgos-Artizzu, P. Perona, and P. Dollár. Robust face landmark estimation under occlusion. In *Computer Vision (ICCV), 2013 IEEE International Conference on*, pages 1513–1520. IEEE, 2013.

[4] C. Cao, Q. Hou, and K. Zhou. Displaced dynamic expression regression for real-time facial tracking and animation. *ACM Transactions on Graphics (TOG)*, 33(4):43, 2014.

[5] X. Cao, Y. Wei, F. Wen, and J. Sun. Face alignment by explicit shape regression. *International Journal of Computer Vision*, 107(2):177–190, 2014.

[6] D. Chen, S. Ren, Y. Wei, X. Cao, and J. Sun. Joint cascade face detection and alignment. In *Computer Vision–ECCV 2014*, pages 109–122. Springer, 2014.

[7] G. Chrysos, S. Zafeiriou, E. Antonakos, and P. Snape. Offline deformable face tracking in arbitrary videos. In *IEEE International Conference on Computer Vision Workshops (ICCVW), 2015*. IEEE.

[8] P. Dollár, P. Welinder, and P. Perona. Cascaded pose regression. In *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*, pages 1078–1085. IEEE, 2010.

[9] I. Kemelmacher-Shlizerman, S. Suwajanakorn, and S. M. Seitz. Illumination-aware age progression. In *Computer Vision and Pattern Recognition (CVPR), 2014 IEEE Conference on*, pages 3334–3341. IEEE, 2014.

[10] M. Kim, S. Kumar, V. Pavlovic, and H. Rowley. Face tracking and recognition with visual constraints in real-world videos. In *Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on*, pages 1–8. IEEE, 2008.

[11] L. Liu, J. Xing, S. Liu, H. Xu, X. Zhou, and S. Yan. Wow! you are so beautiful today! *ACM Transactions on Multimedia Computing, Communications, and Applications (TOMM)*, 11(1s):20, 2014.

[12] P. Martins, R. Caseiro, and J. Batista. Generative face alignment through 2.5 d active appearance models. *Computer Vision and Image Understanding*, 117(3):250–268, 2013.

[13] S. Ren, X. Cao, Y. Wei, and J. Sun. Face alignment at 3000 fps via regressing local binary features. In *Computer Vision and Pattern Recognition (CVPR), 2014 IEEE Conference on*, pages 1685–1692. IEEE, 2014.

[14] J. M. Saragih, S. Lucey, and J. F. Cohn. Deformable model fitting by regularized landmark mean-shift. *International Journal of Computer Vision*, 91(2):200–215, 2011.

[15] J. Shen, S. Zafeiriou, G. Chrysos, J. Kossaifi, G. Tzimiropoulos, and M. Pantic. The first facial landmark tracking in-the-wild challenge: Benchmark and results. In *IEEE International Conference on Computer Vision Workshops (ICCVW), 2015*. IEEE.

[16] G. Tzimiropoulos. Project-out cascaded regression with an application to face alignment. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3659–3667, 2015.

[17] G. Tzimiropoulos and M. Pantic. Optimization problems for fast aam fitting in-the-wild. In *Computer Vision (ICCV), 2013 IEEE International Conference on*, pages 593–600. IEEE, 2013.

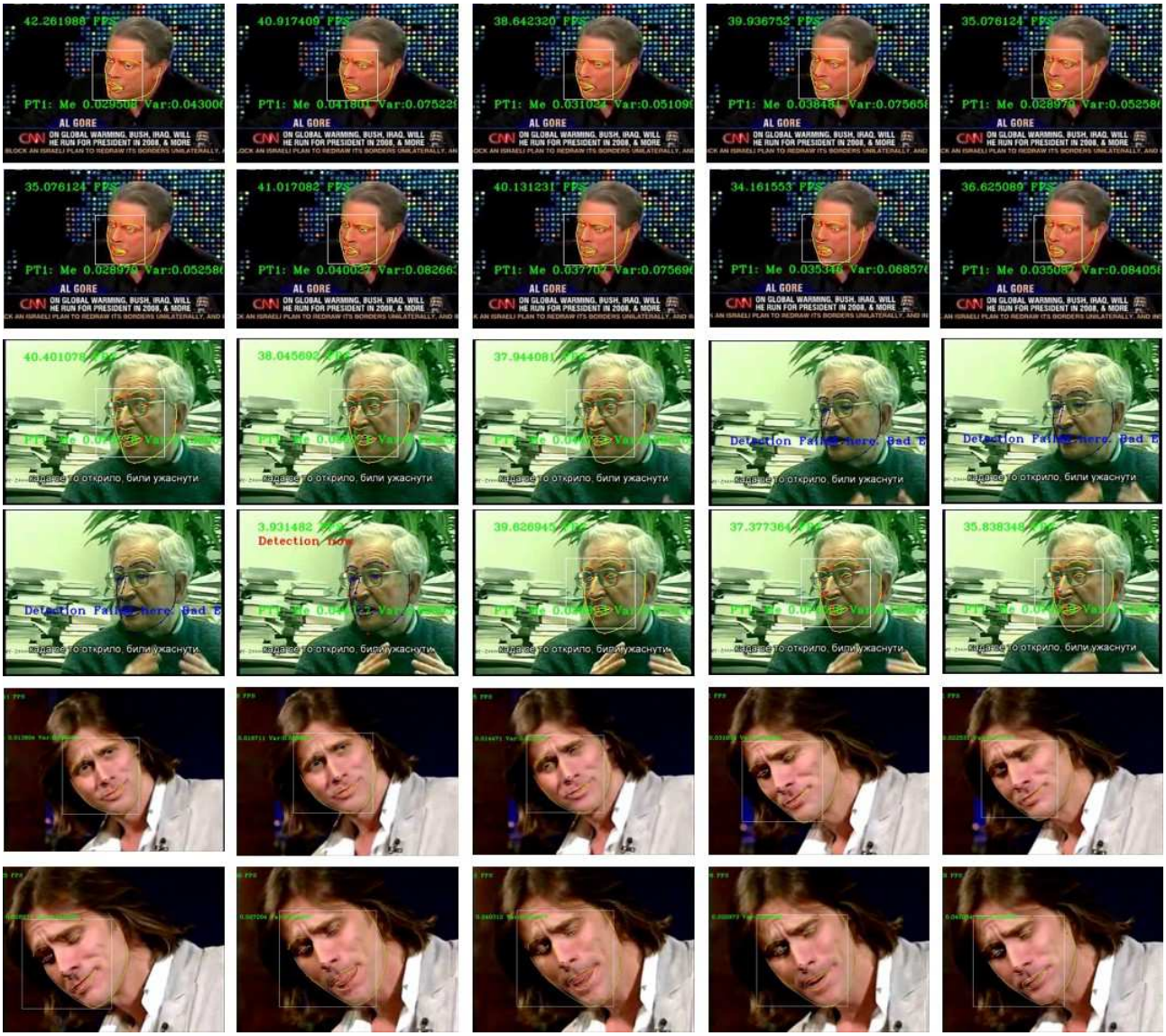

Figure 7. Qualitative facial landmark tracking results of selected frames from Youtube Celebrities Database.

[18] X. Xiong and F. De la Torre. Supervised descent method and its applications to face alignment. In *Computer Vision and Pattern Recognition (CVPR), 2013 IEEE Conference on*, pages 532–539. IEEE, 2013.

[19] Z. Zhang, P. Luo, C. C. Loy, and X. Tang. Facial landmark detection by deep multi-task learning. In *Computer Vision–ECCV 2014*, pages 94–108. Springer, 2014.

[20] W. Zhao, R. Chellappa, P. J. Phillips, and A. Rosenfeld. Face recognition: A literature survey. *ACM computing surveys (CSUR)*, 35(4):399–458, 2003.

[21] S. Zhu, C. Li, C. C. Loy, and X. Tang. Face alignment by coarse-to-fine shape searching. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4998–5006, 2015.

[22] X. Zhu and D. Ramanan. Face detection, pose estimation, and landmark localization in the wild. In *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*, pages 2879–2886. IEEE, 2012.