Facial Landmark Tracking by Tree-based Deformable Part Model Based Detector

Michal Uřičář, Vojtěch Franc, and Václav Hlaváč
Center for Machine Perception, Department of Cybernetics
Faculty of Electrical Engineering, Czech Technical University in Prague
166 27 Prague 6, Technická 2, Czech Republic
{uricamic, xfrancv, hlavac}@cmp.felk.cvut.cz

Abstract

In this paper we describe a tracker of facial landmarks submitted to the 300 Videos in the Wild (300-VW) challenge. Our tracker is a straightforward extension of a well tuned tree-based DPM landmark detector originally developed for static images. The tracker is obtained by applying the static detector independently in each frame and using the Kalman filter to smooth estimates of the face positions as well as to compensate possible failures of the face detector. The resulting tracker provides a robust estimate of 68 landmarks running at 5 fps on an ordinary PC. We provide an open-source implementation of the proposed tracker at (`http://cmp.felk.cvut.cz/~uricamic/clandmark/`).

1. Introduction

The tracking of facial landmarks in image sequences is an important and challenging task in computer vision. The precise landmark localization can improve the face recognition [5], expression analysis [19] or head-pose estimation [30]. All of these tasks are important e.g. for a human-computer interaction, where the ability to recognize the state of human face gives the possibility of much more natural communication.

The 300-VW challenge is constrained to the scenario in which i) a single person appears in the video, ii) the face is typically visible from the first to the last frame and iii) all face appearances are near frontal so that there are almost no self-occluded landmarks. The challenge videos require a tracker working reliably in unconstrained environments, under various lighting conditions, with faces in arbitrary expressions and possibly occluded by glasses, moving hands etc.

In this paper we describe a landmark tracker obtained by straightforward extension of a well tuned landmark detector for static images [28]. Our static landmark detector uses tree-based deformable part models trained by the Structured Output SVMs [25]. The tree based DPM allow for a global inference procedure solved by the dynamic programming hence not suffering with locally optimal estimates typical for methods with complex shape model.

The main problem of the tree-based DPMs, i.e the long processing time resulting from the combinatorial inference problem, is alleviated by using a coarse-to-fine search strategy. In particular, the pipeline of the static landmark detector has the following three stages. In the first stage, it finds a rough position by a face detector. In the second stage, it refines the found face box by applying a tree-based DPM landmark detector operating on a low-resolution image. In the last third stage, it uses the refined face position to define a narrow search space for each landmark in a higher resolution image and it finds the resulting landmark configuration by another tree-based DPM detector. The tracker is obtained by applying the static landmark detector independently in each frame and using the Kalman filter [12] to smooth the estimate of the face position (more precisely the face box parametrized by its center and size) which is computed in the first and the second stage of the pipeline. The resulting tracker provides a robust estimate of 68 landmarks and it runs 5 fps on an ordinary PC.

An example frame from one of the 300-VW public sequences is shown in Figure 1.

The Kalman filter has become a standard tool for image based object tracking. In the context of the landmark tracking it has been used many times e.g. in conjunction with the Active Shape Models (ASM), either for tracking the face boxes [16] or the directly the landmarks [15]. A robust tracking of multiple faces by a Kalman filter (similarly as we do in our pipeline) was proposed e.g. in [20]. On the other hand, we are not aware of other work combining the Kalman filter and the tree-based DPMs [3].

The paper is organized as follows. Section 2 describes

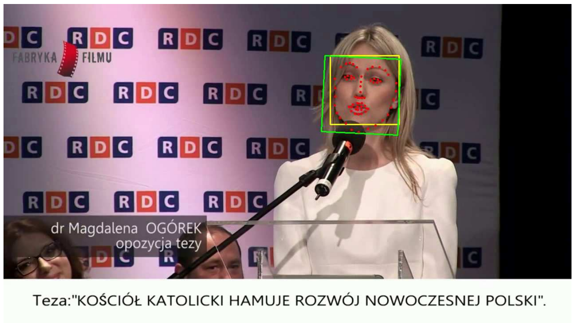


Figure 1: Exemplary frame from one of the 300-VW public sequences. The yellow box represents the output of the face detector. The green box is the refined face box constructed from the landmarks detected by the coarse detector. The red points are detected landmarks as detected by the fine detector.

the pipeline and its components. Experimental evaluation of the proposed method is given in Section 3 and, finally, Section 4 concludes the paper.

2. Proposed Method

The processing pipeline of the proposed tracker is depicted in Figure 2. The core of the tracker is the static landmark detector (blue boxes in the figure) using two-stage coarse-to-fine search strategy to speed up the tree-based DPM structured classifier of landmark positions. We describe the static tree-based DPM detector in Section 2.1, its learning in Section 2.2 and the mentioned coarse-to-fine strategy in Section 2.3. The orange boxes in Figure 2 correspond to the Kalman filters used to stabilize estimates of the face position (the face boxes) provided by the detector and the one computed from the output of the coarse DPM landmark detector. The particular setting of the used Kalman filters is described in Section 2.4.

2.1. Landmark Detector on Static Image

We build our work on [27], which formulated a DPM facial landmark detector on static images. We recapitulate the important parts for the sake of completeness. The DPM approach [9, 6, 7] pose the detection task as an energy minimization problem. [27] translates this scheme into a structured output classification framework by introducing a scoring function which is to be maximized w.r.t. the landmark positions. The shape model is represented by an undirected graph $G = (V, E)$, where V is a finite set of vertices representing the landmarks and $E \subset \binom{V}{2}$ is a set of edges between pairs of landmarks, whose positions are related[1].

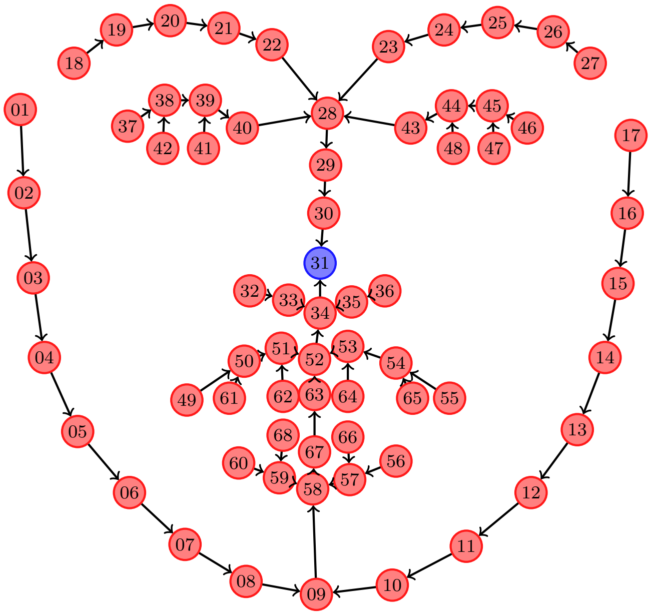


Figure 3: The graph structure of the 68 landmarks configuration used for both coarse and fine detectors. Note, that the graph forms a tree rooted at the landmark emphasized by a blue circle.

For the sake of 300-VW competition, we use $|V| = 68$ and restrict the graph G to form a tree (see Figure 3), which enables the globally optimal solution in a feasible time.

Let $I \in \mathcal{I}^{H\times W}$ be a fixed-size image, let $\boldsymbol{s} = (\boldsymbol{s}_1, \ldots, \boldsymbol{s}_{|V|}) \in \mathcal{S} = (\mathcal{S}_1 \times \cdots \times \mathcal{S}_{|V|}) \in (H \times W)^{|V|}$ be a configuration of landmark locations and, finally, let $\boldsymbol{w}$ denote the vector of weights composed of weights $\boldsymbol{w}_i \in \mathbb{R}^{n_i}$ and $\boldsymbol{w}_{ij} \in \mathbb{R}^4$ (n_i denotes the number of parameters) associated with the unary and pair-wise potentials, respectively. Then, the scoring function and landmark detector $h\colon \mathcal{I} \to \mathcal{S}$, are defined as follows:

$$\begin{aligned} f(I, \boldsymbol{s}; \boldsymbol{w}) &= \sum_{i\in V} q_i(\boldsymbol{s}_i, I; \boldsymbol{w}_i^q) + \sum_{(i,j)\in E} g_{ij}(\boldsymbol{s}_i, \boldsymbol{s}_j; \boldsymbol{w}_{ij}^g) \\ h(I; \boldsymbol{w}) &= \arg\max_{\boldsymbol{s}\in\mathcal{S}} f(I, \boldsymbol{s}; \boldsymbol{w}). \end{aligned} \quad (1)$$

The unary potentials $q_i(\boldsymbol{s}_i, I; \boldsymbol{w}_i^q)$ measure the quality of a fit of individual landmarks $\boldsymbol{s}_i$, $i \in V$, to the image I. We use a linear parametrization of unary potentials

$$q_i(\boldsymbol{s}_i, I; \boldsymbol{w}_i^q) = \langle \boldsymbol{w}_i^q, \boldsymbol{\Psi}_i^q(I, \boldsymbol{s}_i) \rangle \ , \quad (2)$$

where feature descriptor $\boldsymbol{\Psi}_i^q(I, \boldsymbol{s}_i)\colon \mathcal{I} \times \mathcal{S}_i \to \mathbb{R}^{n_i^q}$ cropped from the image patch around the position $\boldsymbol{s}_i$ is represented by a multi-scale pyramid of Sparse Local Binary Patterns (S-LBP) [23, 27], precomputed on a whole input image I as suggested in [28]. The dimensionality of a multi-scale

[1] A set of edges of a fully connected graph with nodes V is denoted by $\binom{V}{2}$.

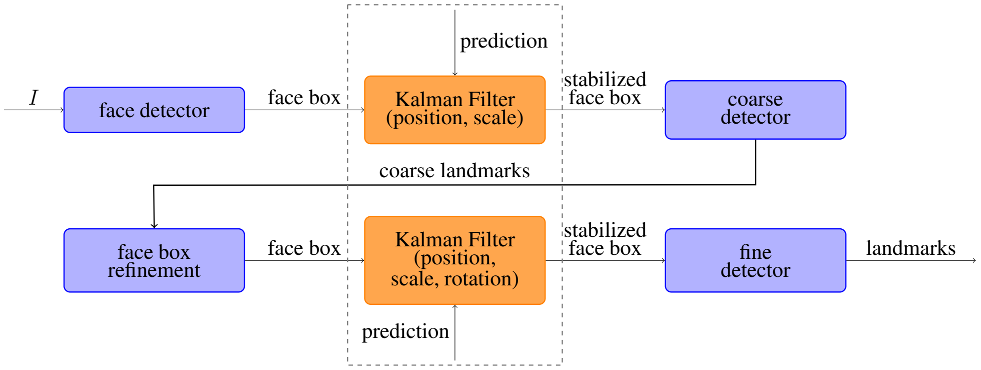


Figure 2: Complete scheme of the proposed method. The blue boxes represent the coarse-to-fine strategy alone. By pluging in the Kalman filters for both face boxes used by the coarse and fine detectors, we get the proposed tracker working on image sequences.

S-LBP features for a patch of size 15×15 px is $(13 \times 13) + (5 \times 5) + (1 \times 1) \cdot 256 = 195 \cdot 256 = 49,920$, with only 195 non-zero entries. We use S-LBP features mainly because of the neat trade-off between the accuracy and speed.

The pair-wise potentials $g_{ij}(\boldsymbol{s}_i, \boldsymbol{s}_j; \boldsymbol{w}^g_{ij})$ measure the likelihood of the mutual position of the connected pairs of landmarks. We use the linear parametrization as proposed in [27]

$$g_{ij}(\boldsymbol{s}_i, \boldsymbol{s}_j; \boldsymbol{w}^g_{ij}) = \langle \boldsymbol{w}^g_{ij}, \boldsymbol{\Psi}^g_{ij}(\boldsymbol{s}_i, \boldsymbol{s}_j) \rangle \ , \tag{3}$$

where the feature vector $\boldsymbol{\Psi}^g_{ij}(\boldsymbol{s}_i, \boldsymbol{s}_j) \colon \mathcal{S}_i \times \mathcal{S}_j \to \mathbb{R}^4$ has exactly the same form as suggested in [28]

$$\boldsymbol{\Psi}^g_{ij}(\boldsymbol{s}_i, \boldsymbol{s}_j) = \begin{bmatrix} \delta x \\ \delta y \\ \delta x^2 \\ \delta y^2 \end{bmatrix} \text{,where} \begin{bmatrix} \delta x \\ \delta y \end{bmatrix} = \boldsymbol{s}_i - \boldsymbol{s}_j = \begin{bmatrix} x_i - y_i \\ x_j - y_j \end{bmatrix} . \tag{4}$$

which allows to use distance transform [8] in the inference computation, as long as the g_{ij}, $\forall (ij) \in E$ are concave. This requires the weights $\boldsymbol{w}^g_{ij}$ corresponding to δx^2 and δy^2 in the $\boldsymbol{\Psi}^g_{ij}$ to be negative. Note that this can be easily enforced during learning. We denote the set of all indices whose weight should be negative by symbol W^-.

2.2. Learning of static detector

Because of the linear parametrization of both the unary and the pair-wise potentials, the scoring function is indeed also linear. This leads to a linear classifier, which can be learned by the fully supervised structured output SVM (SO-SVM) framework [25].

We denote the joint parameter vector $\boldsymbol{w}$ and joint feature vector $\boldsymbol{\Psi}$, both of which were constructed as a concatenation of the unary and pair-wise weights $\boldsymbol{w}^q_i, \boldsymbol{w}^g_{ij}$, and feature vectors $\boldsymbol{\Psi}^q_i, \boldsymbol{\Psi}^g_{ij}$, respectively.

The SO-SVM algorithm translates the learning of the parameter vector of a linear structured classifier into the following convex program

$$\begin{aligned} \boldsymbol{w}^* &= \arg\min_{\boldsymbol{w} \in \mathbb{R}^n} \left[\frac{\lambda}{2} \|\boldsymbol{w}\|^2 + \frac{1}{m} \sum_{i=1}^{m} r_i(\boldsymbol{w}) \right] \\ &\text{s.t.} \quad w_i \le c^-, \quad i \in W^- \ . \end{aligned} \tag{5}$$

where $r_i(\boldsymbol{w})$ is a loss incurred by the classifier on the i-th training example $(I^i, \boldsymbol{s}^i)$ and $\frac{\lambda}{2}\|\boldsymbol{w}\|^2$ is a quadratic regularizer introduced to prevent over-fitting. The optimal setting of the regularization constant $\lambda > 0$ is tuned on a validation set. The inequality constraints are used to ensure the concavity of functions g_{ij}. We set c^- to a small negative constant. The loss $r_i(\boldsymbol{w})$ is the margin-rescaling convex proxy (c.f. [25]) of the true loss $\Delta(\boldsymbol{s}, \boldsymbol{s}')$ and it reads

$$r_i(\boldsymbol{w}) = \max_{\boldsymbol{s} \in \mathcal{S}} \left[\Delta(\boldsymbol{s}, \boldsymbol{s}') + \left\langle \boldsymbol{w}, \boldsymbol{\Psi}(I^i, \boldsymbol{s}) - \boldsymbol{\Psi}(I^i, \boldsymbol{s}^i) \right\rangle \right] . \tag{6}$$

The true loss is defined as the normalized average displacement of the ground truth and estimated landmark locations

$$\Delta(\boldsymbol{s}, \boldsymbol{s}') = \frac{1}{\kappa(\boldsymbol{s})|V|} \sum_{j=1}^{|V|} \|\boldsymbol{s}_j - \boldsymbol{s}'_j\|, \tag{7}$$

where the normalization constant $\kappa(\boldsymbol{s})$ is the inter-ocular distance (IOD) as commonly used in the landmark localization precision evaluation [17].

We solve (5) by the Bundle Methods for Regularized Risk Minimization (BMRM) algorithm [24] modified to accept the inequality constraints on $\boldsymbol{w}$. We skip the details of the BMRM algorithm and refer the reader to [24]. The only requirements of BMRM algorithm are procedures evaluating the risk $r(\boldsymbol{w}) = \frac{1}{m}\sum_{i=1}^{m} r_i(\boldsymbol{w})$ and the sub-gradient $\boldsymbol{r}'(\boldsymbol{w}) = \frac{1}{m}\sum_{i=1}^{m} \boldsymbol{r}'_i(\boldsymbol{w})$. The sub-gradient for the i-th training example is computed as follows

$$\begin{aligned} \boldsymbol{r}'_i(\boldsymbol{w}) &= \boldsymbol{\Psi}(I^i, \hat{\boldsymbol{s}}) - \boldsymbol{\Psi}(I^i, \boldsymbol{s}^i), \text{ where} \\ \hat{\boldsymbol{s}} &= \arg\max_{\boldsymbol{s}\in\mathcal{S}} \left[\Delta(\boldsymbol{s}, \boldsymbol{s}') + \left\langle \boldsymbol{w}, \boldsymbol{\Psi}(I^i, \boldsymbol{s}) \right\rangle \right] . \end{aligned} \quad (8)$$

For learning both static DPM detectors, we use the 300-W dataset [18], which consists of the following datasets with unified landmark annotation: AFW [30], HELEN [13], IBUG [18], LPFW [2] and XM2VTS [14]. The 300-W dataset contains $6,193$ examples in total. We use the original split of the images into the training and testing part. For the validation of the regularization constant λ (5), we further reserve 551 examples from the training part, which leaves us with $5,124$ examples for training. The dimensionality of the coarse and fine DPM detector is $2,478,348$ and $3,456,012$, respectively. This implies that having more training example should be beneficial.

2.3. Coarse-to-Fine Strategy to Speed Up DPM detector

In order to keep the detection speed and the landmark localization accuracy high, we propose the following strategy. We learn two different classifiers, which operate on a different scale input image. The first, coarse, detector operates on 80×80 px input image I, with landmark patch size 13×13 px, and serves as the refinement of the imprecise face detector as well as the estimator of the possible in-plane rotation of the face. The second, fine, detector operates on 160×160 px image I, with landmark patch size 15×15 px, and its processing time is significantly reduced by tightening the search spaces of individual landmarks $\mathcal{S}_i$. The root landmark ($\boldsymbol{s}_{31}$, see Figure 3) has patch size 21×21 px for both detectors.

The scheme of the coarse-to-fine strategy is depicted in Figure 2. Both coarse and fine detectors were trained on the identical training data, obtained from the 300-W training subset by running the face detector which returned face boxes with possible in-plane rotation for the coarse detector and face boxes constructed from the ground truth landmarks annotation for the fine detector, respectively. The benefit of this strategy is best seen in Figure 4 presenting the comparison of the accuracy of individual detectors and the coarse-to-fine strategy computed on the 300-W testing set.

2.4. Stabilization of the Face Detection by Kalman Filter

In the previous paragraphs, we have described the detector operating on static images. To extend this approach to image sequences, we mainly need to cope with possible failures (i.e. overlooked faces) of the face detector. In particular, we use a commercial implementation of the Waldboost face detector [22][2]. To this end, we apply the Kalman filter [12] to the estimates of the face boxes (the landmark positions themselves are not filtered). Besides solving the problem with missing detections, the Kalman filter also stabilizes the face box positions which has further positive effect on the resulting estimate of the landmark positions.

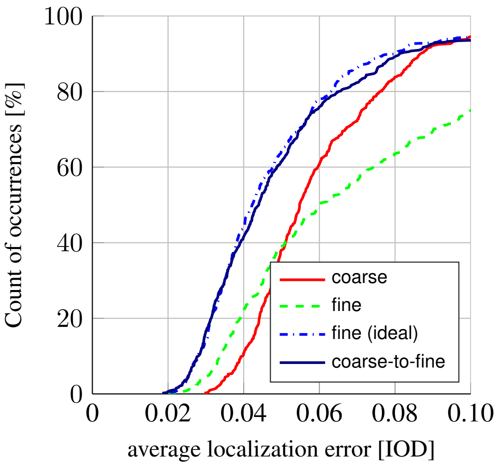


Figure 4: Normalized localization error for the 300-W test set. Note that the fine detector with ideal input (i.e. we assume a perfect face box input that is computed from the ground truth annotation) copies the coarse-to-fine strategy.

Namely, we apply a Kalman filter to stabilize the position and size of the face box returned by the face detector. We also use the Kalman filter to stabilize the position, size and rotation of the refined face box computed from the output of the coarse DMP detector.

For the stabilization of both face boxes position, we use a Kalman filter with a constant velocity model. We set the state vector $\boldsymbol{x}_t$ in frame t as a center of the face box and the quantity of change between the previous and current frames

$$\boldsymbol{x}_t = [x, y, \delta x, \delta y]^\top \, . \quad (9)$$

The prediction is driven by the linear equation

$$\boldsymbol{x}_{t+1} = \boldsymbol{A}\boldsymbol{x}_t + \xi(t) \, , \quad (10)$$

where $\xi(t) \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{Q}(t))$ and the state transition matrix $\boldsymbol{A}$ is defined as follows

$$\boldsymbol{A} = \begin{pmatrix} 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} . \quad (11)$$

The measurement vector is given by

$$\boldsymbol{z}_t = \boldsymbol{H}_t \boldsymbol{x}_t + \zeta(t) \, , \quad (12)$$

where $\zeta(t) \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{R}(t))$ and $\boldsymbol{H}_t$ is a measurement matrix defined as

$$\boldsymbol{H}_t = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{pmatrix} . \quad (13)$$

[2] Courtesy of Eyedea Recognition, Ltd.

The Kalman filter keeps track of the state estimate $\hat{\boldsymbol{x}}_t$ and its error covariance $\hat{\boldsymbol{P}}_t$. During the prediction stage $\hat{\boldsymbol{x}}'_t$ and $\hat{\boldsymbol{P}}'_t$ are obtained

$$\begin{aligned} \hat{\boldsymbol{x}}'_{t+1} &= \boldsymbol{A}\hat{\boldsymbol{x}}_t && (14) \\ \hat{\boldsymbol{P}}'_{t+1} &= \boldsymbol{A}\hat{\boldsymbol{P}}_t\boldsymbol{A}^\top + \boldsymbol{Q}(t)\ . && (15) \end{aligned}$$

During the correction stage, a Kalman gain $\boldsymbol{K}_t$ is computed

$$\boldsymbol{K}_t = \hat{\boldsymbol{P}}'_t\boldsymbol{H}^\top(\boldsymbol{H}\hat{\boldsymbol{P}}'_t\boldsymbol{H}^\top + \boldsymbol{R}(t))^{-1}\ , \qquad (16)$$

which minimizes the a posteriori error covariance and is used to refine the predicted state and error covariance using the noisy measurements

$$\begin{aligned} \hat{\boldsymbol{x}}_t &= \hat{\boldsymbol{x}}'_t + \boldsymbol{K}_t(\boldsymbol{z}_t - \boldsymbol{H}\hat{\boldsymbol{x}}'_t) && (17) \\ \hat{\boldsymbol{P}}_t &= (\boldsymbol{I} - \boldsymbol{K}_t\boldsymbol{H})\hat{\boldsymbol{P}}'_t\ . && (18) \end{aligned}$$

The scale of both face boxes and rotation of the corrected box are also stabilized by applying Kalman filter on the corresponding 1D signal. The corrected state is computed analogically as described in the previous paragraph.

3. Experiments

In the experimental evaluation we show the results on the non-public 300-VW [21, 4, 26] test data, as well as the results on the whole public part (i.e. all 50 sequences), which we do not use for training purposes. On the public part, we show the benefits of the proposed tracker with the stabilization of face detections by a Kalman filter.

3.1. Results on the Public 300-VW Data

In the first experiment, we evaluate i) the coarse DPM detector when used alone, ii) the coarse-to-fine strategy, and iii) the coarse-to-fine strategy with the stabilization by Kalman filters, i.e. the proposed tracker.

Figure 5a shows the average localization error normalized by the IOD distance for all of the public 300-VW sequences. To get the better insight of the proposed tracker capabilities, we further split the public 300-VW sequences into 2 parts— easy, which contains the sequences where the face detector triggered on almost all frames and where the head pose was mostly near-frontal, i.e. within the yaw angle in the interval $(-15°, 15°)$, and hard where the face detector failure was fairly higher and where the yaw angle of the head was from much broader interval. The results obtained on the easy and hard sequences are depicted in Figures 5b and 5c, respectively.

Note, that the coarse-to-fine detector for static images performs very well. However, the stabilization of the face detection brings some improvement, even on the easy sequences. The significant improvement of the proposed tracker is apparent on the hard sequences.

The noticeable drop of performance on the hard sequences is partially explained by the too extreme poses for which we do not have trained our coarse-to-fine facial landmark detector.

The second experiment is the comparison of the proposed tracker to the state-of-the-art IntraFace [29] tracker. The results for all of the public 300-VW sequences are shown in Figure 6a, and for the easy and hard parts in Figures 6b and 6c, respectively. In the comparison, we use only the 49 landmarks common to both trackers. Note, that while IntraFace achieves overall better results, the proposed tracker has higher percentage of frames with error lower than 3% of the IOD. We observed that the IntraFace tracker works more reliably for much higher yaw intervals, since it was trained on the Multi-PIE [10] and LFW [11] databases. The results on the easy part shows that when the yaw range is close to $(-15°, 15°)$, the proposed tracker is on the par with IntraFace.

3.2. Results on the Non-public 300-VW Data

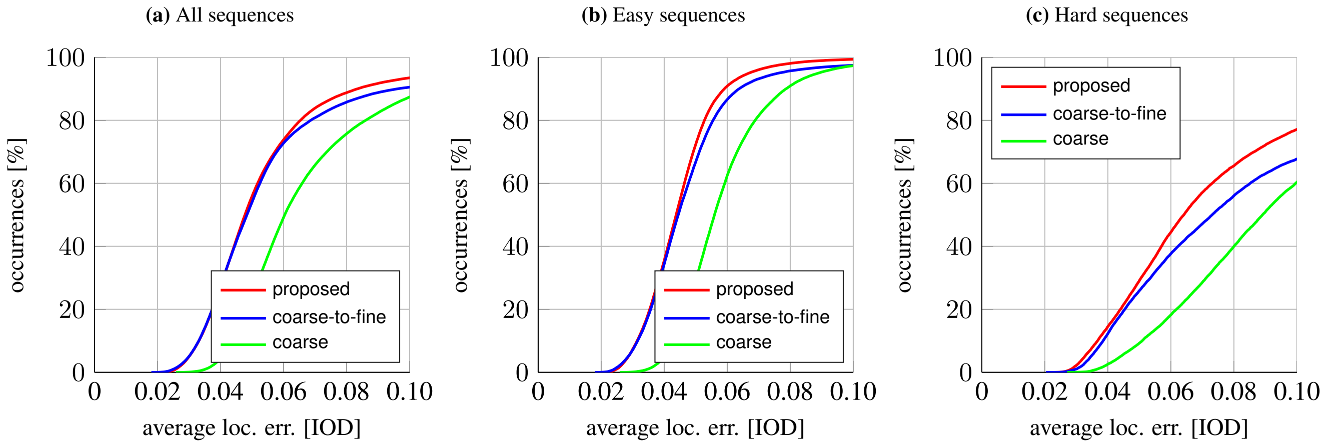
The third experiment was conducted by the 300-VW organizers and we present here the collected results in Figures 7a, 7b and 7c. The localization error is evaluated on a subset of 49 mutual to the chehra [1] facial landmark tracker. Note, that the proposed tracker outperforms the chehra baseline by a big margin in all three scenarios. The results for all 68 landmarks are depicted in Figures 8a, 8b and 8c.

3.3. Speed

In all experiments, we let the face detector detect the face in each frame. In case of a missing detection, we use the prediction provided by the Kalman filter, otherwise we correct the face detection by Kalman Filter. With these settings, we can achieve around 5 fps. The straightforward speedup can be achieved by not triggering the face detector in each frame, but only once per several frames. Since we have shown the generalization capabilities of the coarse detector, this modification is justified and brings a speedup to around 10 fps. Yet another speedup can be achieved by training the coarse classifier for only few landmarks, needed to calculate the possible in-plane rotation. This could drop the processing time of the coarse-to-fine landmark detector to around 50–60 ms per frame, which is comparable to the IntraFace tracker (which detects just 49 landmarks compared to the 68 landmarks detected by the proposed tracker). However, the mentioned speed ups have not been implemented.

4. Conclusions

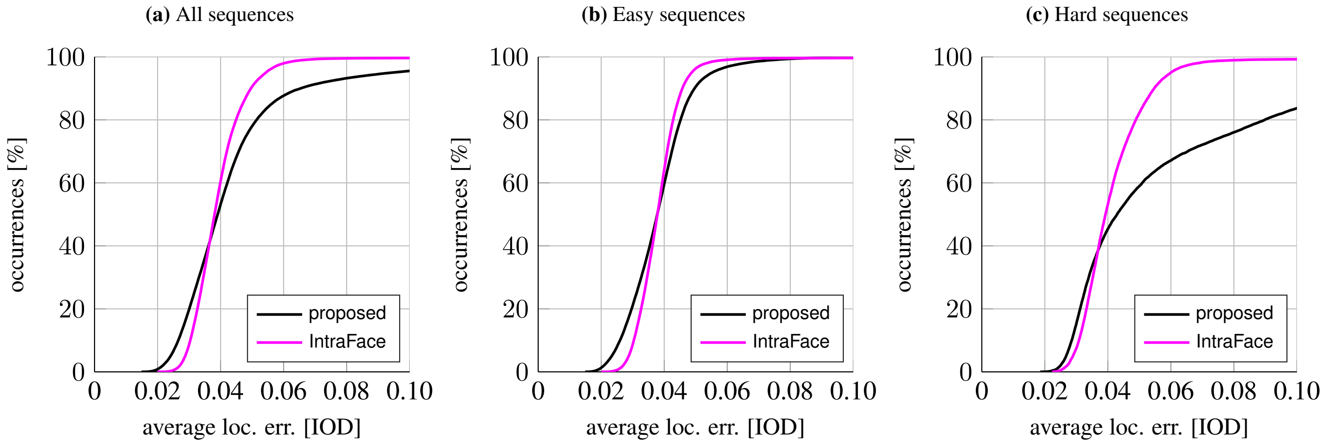
In this paper we showed that a robust landmark tracker can be obtained by running a static landmark detector independently in each frame and smoothing the face positions


(a) All sequences

(b) Easy sequences

(c) Hard sequences

Figure 5: The average normalized localization precision for all (a), easy (b) and hard (c) public 300-VW sequences evaluated on all 68 landmarks.


(a) All sequences

(b) Easy sequences

(c) Hard sequences

Figure 6: The precision curves for all (a), easy (b) and hard (c) public 300-VW sequences evaluated on 49 landmarks (a subset common to the proposed tracker and IntraFace [29]).

by the Kalman filter. The empirical results support the intuition that the Kalman filter improves the results mainly in complex scenes in which the face detector often fails. The resulting tracker runs at 5 fps on an ordinary PC but there is a large room for improvements in terms of the detection speed (e.g. not running the face detector in each frame).

We provide an open-source implementation of the proposed tracker for reproducing our results (`http://cmp.felk.cvut.cz/~uricamic/clandmark/`).

Acknowledgement

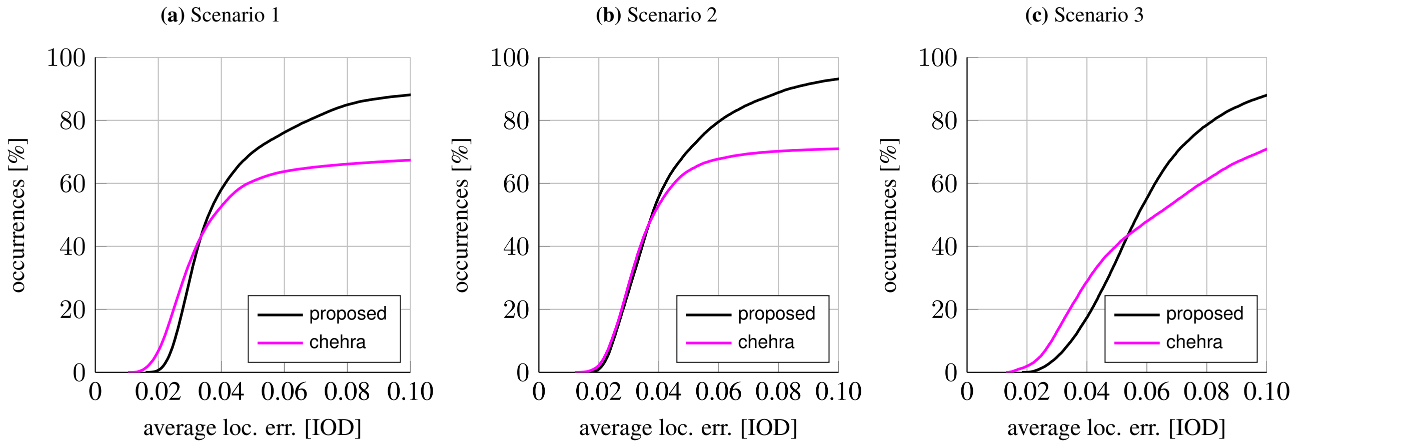
The first author was supported by The Grant Agency of the CTU in Prague project SGS15/201/OHK3/3T/13. The second author was supported by the Ministry of Education, Youth and Sports under project ERC-CZ LL1303. The third author was supported by the Ministry of Education under Project RVO13000.

References

[1] A. Asthana, S. Zafeiriou, S. Cheng, and M. Pantic. Incremental Face Alignment in the Wild. In *The 27th IEEE Conference on Computer Vision and Pattern Recognition, CVPR'14*, pages 1859–1866, Columbus, OH, USA, June 2014.

[2] P. N. Belhumeur, D. W. Jacobs, D. J. Kriegman, and N. Kumar. Localizing parts of faces using a consensus of exemplars. In *The 24th IEEE Conference on Computer Vision and Pattern Recognition, CVPR'11*, pages 545–552, Colorado Springs, CO, USA, June 2011.

[3] O. Çeliktutan, S. Ulukaya, and B. Sankur. A comparative study of face landmarking techniques. *EURASIP Journal on*

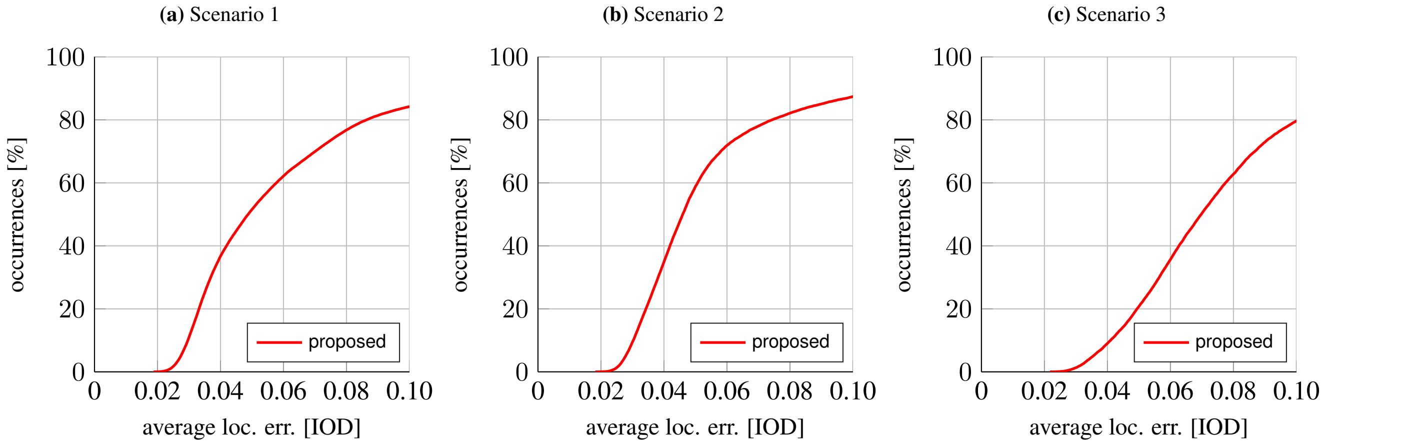


(a) Scenario 1

(b) Scenario 2

(c) Scenario 3

Figure 7: The precision curves for the Scenario 1 (a), 2 (b) and 3 (c) of the non-public 300-VW sequences evaluated on 49 landmarks (a subset common to the proposed tracker and chehra [1]).



(a) Scenario 1

(b) Scenario 2

(c) Scenario 3

Figure 8: The precision curves for the Scenario 1 (a), 2 (b) and 3 (c) of the non-public 300-VW sequences evaluated on all 68 landmarks.

Image and Video Processing, 2013(1):13, 2013.

[4] 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.

[5] G. J. Edwards, T. F. Cootes, and C. J. Taylor. Face recognition using active appearance models. In *Computer Vision - ECCV'98, 5th European Conference on Computer Vision, Freiburg, Germany, June 2-6, 1998, Proceedings, Volume II*, pages 581–595, 1998.

[6] P. F. Felzenszwalb, R. B. Girshick, D. A. McAllester, and D. Ramanan. Object Detection with Discriminatively Trained Part-Based Models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 32(9):1627–1645, 2010.

[7] P. F. Felzenszwalb and D. P. Huttenlocher. Pictorial Structures for Object Recognition. *International Journal of Computer Vision*, 61(1):55–79, 2005.

[8] P. F. Felzenszwalb and D. P. Huttenlocher. Distance Transforms of Sampled Functions. *Theory of Computing*, 8(1):415–428, 2012.

[9] M. A. Fischler and R. A. Elschlager. The Representation and Matching of Pictorial Structures. *IEEE Transactions on Computers*, C-22(1):67–92, 1973.

[10] R. Gross, I. Matthews, J. F. Cohn, T. Kanade, and S. Baker. Multi-PIE. *Image and Vision Computing*, 28(5):807–813, 2010.

[11] G. B. Huang, M. Ramesh, T. Berg, and E. Learned-Miller. Labeled Faces in the Wild: A Database for Studying Face Recognition in Unconstrained Environments. Technical Report 07-49, University of Massachusetts, Amherst, October 2007.

[12] R. E. Kalman. A new approach to linear filtering and prediction problems. *ASME Journal of Basic Engineering*, 1960.

[13] V. Le, J. Brandt, Z. Lin, L. Bourdev, and T. S. Huang. Interactive facial feature localization. In *Proceedings of the 12th European Conference on Computer Vision - Volume Part III*, ECCV'12, pages 679–692, Berlin, Heidelberg, 2012. Springer-Verlag.

[14] K. Messer, J. Matas, J. Kittler, and K. Jonsson. Xm2vtsdb: The extended m2vts database. In *In Second International Conference on Audio and Video-based Biometric Person Authentication*, pages 72–77, 1999.

[15] U. Prabhu, K. Seshadri, and M. Savvides. Automatic facial landmark tracking in video sequences using kalman filter assisted active shape models. In *Trends and Topics in Computer Vision - ECCV 2010 Workshops, Heraklion, Crete, Greece, September 10-11, 2010, Revised Selected Papers, Part I*, pages 86–99, 2010.

[16] B. Pu, S. Liang, Y. Xie, Z. Yi, and P.-A. Heng. Video facial feature tracking with enhanced asm and predicted meanshift. In *Computer Modeling and Simulation, 2010. ICCMS '10. Second International Conference on*, volume 2, pages 151–155, Jan 2010.

[17] C. Sagonas, G. Tzimiropoulos, S. Zafeiriou, and M. Pantic. 300 Faces in-the-Wild Challenge: The first facial landmark localization Challenge. In *Proceedings of IEEE International Conference on Computer Vision Workshops, ICCV'13 Workshops*, Sydney, Australia, December 2013.

[18] C. Sagonas, G. Tzimiropoulos, S. Zafeiriou, and M. Pantic. A Semi-automatic Methodology for Facial Landmark Annotation. In *The 26th IEEE Conference on Computer Vision and Pattern Recognition Workshops, CVPR'13 Workshops*, pages 896–903, Portland, OR, USA, June 2013.

[19] A. A. Salah, H. Ç. Akakin, L. Akarun, and B. Sankur. Robust facial landmarking for registration. *Annales des Télécommunications*, 62(1-2):83–108, 2007.

[20] Z. Shaik and V. Asari. A robust method for multiple face tracking using kalman filter. In *36th Applied Imagery Pattern Recognition Workshop, AIPR 2007, Washington, DC, USA, October 10-12, 2007, Proceedings*, pages 125–130, 2007.

[21] 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.

[22] J. Šochman and J. Matas. WaldBoost - Learning for Time Constrained Sequential Detection. In *The 18th IEEE Conference on Computer Vision and Pattern Recognition, CVPR'05*, pages 150–156, San Diego, CA, USA, June 2005.

[23] S. Sonnenburg and V. Franc. COFFIN: A Computational Framework for Linear SVMs. In *Proceedings of the 27th International Conference on Machine Learning, ICML'10*, pages 999–1006, Haifa, Israel, June 2010.

[24] C. H. Teo, S. V. N. Vishwanathan, A. J. Smola, and Q. V. Le. Bundle Methods for Regularized Risk Minimization. *Journal of Machine Learning Research*, 11:311–365, 2010.

[25] I. Tsochantaridis, T. Joachims, T. Hofmann, and Y. Altun. Large Margin Methods for Structured and Interdependent Output Variables. *Journal of Machine Learning Research*, 6:1453–1484, 2005.

[26] 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.

[27] M. Uřičář, V. Franc, and V. Hlaváč. Detector of Facial Landmarks Learned by the Structured Output SVM. In *Proceedings of the International Conference on Computer Vision Theory and Applications, VISAPP'12*, volume 1, pages 547–556, Rome, Italy, February 2012.

[28] M. Uřičář, V. Franc, D. Thomas, A. Sugimoto, and V. Hlaváč. Real-time Multi-view Facial Landmark Detector Learned by the Structured Output SVM. In *Proceedings of the 11th IEEE International Conference on Automatic Face and Gesture Recognition Conference and Workshops, BWILD'15*, Ljubljana, Slovenia, May 2015. IEEE.

[29] X. Xiong and F. D. la Torre. Supervised Descent Method and Its Applications to Face Alignment. In *The 26th IEEE Conference on Computer Vision and Pattern Recognition, CVPR'13*, pages 532–539, Portland, OR, USA, June 2013.

[30] X. Zhu and D. Ramanan. Face detection, pose estimation, and landmark localization in the wild. In *The 25th IEEE Conference on Computer Vision and Pattern Recognition, CVPR'12*, pages 2879–2886, Providence, RI, USA, June 2012.