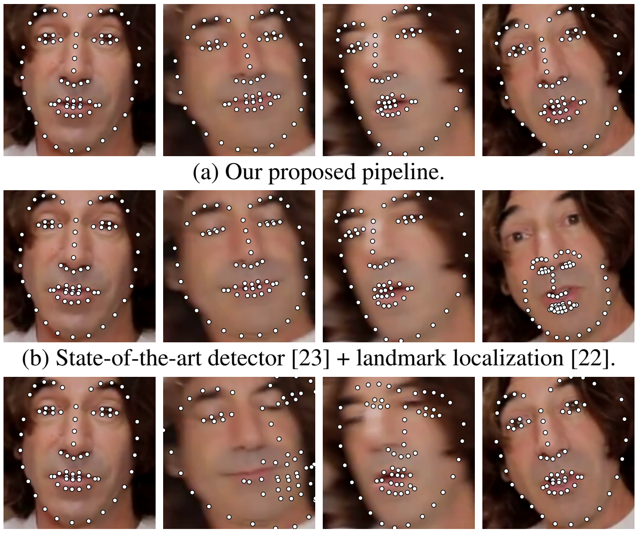
Offline Deformable Face Tracking in Arbitrary Videos

Grigorios G. Chrysos Epameinondas Antonakos* Stefanos Zafeiriou* Patrick Snape
Department of Computing, Imperial College London
180 Queens Gate, SW7 2AZ, London, U.K.
{g.chrysos, e.antonakos, s.zafeiriou, p.snape}@imperial.ac.uk

Abstract

Generic face detection and facial landmark localization in static imagery are among the most mature and well-studied problems in machine learning and computer vision. Currently, the top performing face detectors achieve a true positive rate of around 75-80% whilst maintaining low false positive rates. Furthermore, the top performing facial landmark localization algorithms obtain low point-to-point errors for more than 70% of commonly benchmarked images captured under unconstrained conditions. The task of facial landmark tracking in videos, however, has attracted much less attention. Generally, a tracking-by-detection framework is applied, where face detection and landmark localization are employed in every frame in order to avoid drifting. Thus, this solution is equivalent to landmark detection in static imagery. Empirically, a straightforward application of such a framework cannot achieve higher performance, on average, than the one reported for static imagery[1]*. In this paper, we show for the first time, to the best of our knowledge, that the results of generic face detection and landmark localization can be used to recursively train powerful and accurate person-specific face detectors and landmark localization methods for offline deformable tracking. The proposed pipeline can track landmarks in very challenging long-term sequences captured under arbitrary conditions. The pipeline was used as a semi-automatic tool to annotate the majority of the videos of the 300-VW Challenge*[2]*.*



(a) Our proposed pipeline.

(b) State-of-the-art detector [23] + landmark localization [22].

(c) State-of-the-art tracker [16] + landmark localization [22].

Figure 1: We propose a pipeline for robust and accurate offline deformable face tracking in long-term sequences. Our system significantly outperforms current tracking-by-detection techniques using either state-of-the-art face detectors (1b) or rigid object trackers (1c).

1. Introduction

Generic face detection is widely regarded as a mature field and has been successfully integrated into a number of consumer-grade electronics including digital cameras and modern smartphones [45, 48, 39, 28]. Modern face detectors are robust under a large variety of arbitrary, often termed "in-the-wild", conditions. Recently, due to (i) the efforts made by the community to collect and annotate facial data with regards to a consistent set of facial landmarks [19, 11, 25, 48, 32, 33, 30], and (ii) the development of robust deformable models [37, 41, 8, 4, 5, 2, 38, 22, 29, 6, 10, 3], significant progress has been made towards accurate and efficient facial landmark localization under both controlled and unconstrained conditions. This progress is, arguably, one of the most important steps towards high performance face recognition and verification [36], as well as facial expression recognition [34].

Despite the fact that face detection and facial landmark localization in static imagery has received considerable attention, facial landmark tracking in lengthy videos (also re-

*The authors had equal contribution.

[1] We found that, in practice, the performance is dramatically lower than the one reported for static imagery due to the extremely challenging recording conditions of arbitrary videos.

[2] http://ibug.doc.ic.ac.uk/resources/300-VW/

ferred to as deformable face tracking) has attracted much less research effort. Currently, there are no existing methods that are able to reliably track the landmarks of a human face over lengthy periods without the need of human intervention and re-initialization. This challenging problem is often referred to, in literature, as "tracking in long-term sequences" or simply "long-term tracking" [21], even though those terms are used for tracking by means of a bounding box. The main reasons behind this lack of attention for deformable face tracking are:

- The creation of a meticulously designed benchmark for deformable face tracking in lengthy videos requires the manual annotation of a set of landmarks in each frame of each video. This is a laborious and very expensive task that has yet to be tackled by any research group. Furthermore, the sequences that are currently used for demonstrating qualitative performance of deformable face tracking algorithms are very short (in the range of 3-5 seconds long [40]) and are chosen so that standard face detection algorithms, such as Viola Jones [39], perform well [41, 31, 9, 12].
- There is a trend towards approaching facial landmark tracking as a by-product of generic face detection and landmark localization (i.e., tracking is performed by applying face detection followed by landmark localization at each frame) [41, 38, 29]. This is motivated also by the recent literature in object tracking where tracking-by-detection is usually applied in order to circumvent the drifting issues [21, 47].

We believe that deformable face tracking cannot be effectively solved by using the standard tracking-by-detection procedure of applying face detection followed by landmark localization at each frame [41, 38, 29]. By inspecting the results of coupling state-of-the-art face detection and landmark localization algorithms [32, 30, 28, 2, 38, 13], we note a significant gap in performance. The current state-of-the-art generic face detection algorithms [28, 13] report a true positive rate of about 75-80% in the popular FDDB benchmark [20] whilst allowing for a very small number of false positives. Hence, approximately 20-25% of faces have not been successfully detected. Similarly, the results of the latest landmark localization competition (i.e., 300W [32, 30]), as well as the reported performance of recent methods [2, 38, 29], indicate that the best performing landmark localization methods manage to successfully align no more than 70% of the images of databases with challenging capture conditions. In practice, we empirically found that the performance of state-of-the-art generic face detection and landmark localization algorithms on arbitrary videos collected from YouTube and other sources can be much lower than reported on static images. Figure 1 shows a characteristic example in which neither a state-of-the-art tracker [16], or state-of-the-art face detector [23], followed by landmark localization, successfully track the face in an arbitrary video. In contrast, our pipeline is both accurate and robust in these sequences.

In this paper we show that even though state-of-the-art generic methodologies for face detection and landmark localization cannot solve the facial landmark tracking problem in unconstrained videos, they provide an excellent base on which to build effective procedures. That is, we propose a pipeline for joint automatic construction of person-specific deformable face detection and landmark localization in an offline manner. Automatic construction of person-specific statistical facial deformable models from videos by exploiting a generic model [14, 31] or in an unsupervised manner [7, 44] has only very recently received attention. Nevertheless, all these methods assume that correct bounding boxes are provided by a robust face detector for all the images, which is rarely the case for arbitrary videos. The aforementioned methods all fail if false positive detections are provided. On the contrary, our method jointly solves the problem of automatic construction of deformable face detection and facial landmark localization. We show that the proposed pipeline is extremely effective for robust long-term offline deformable face tracking. We show quantitative experiments in 16 lengthy videos (1-2 mins, 25 fps), which have been annotated with regards to 68 facial landmarks (more than 30,000 annotated frames) and qualitative results in more than 100 long-term sequences.

In summary, the contributions of this paper are:

- We present an accurate and efficient pipeline for offline deformable facial landmark tracking in long-term sequences. The proposed system is the first, to the best of our knowledge, to return accurate results for all the frames of arbitrary length videos without false positives or drifting issues.
- The proposed technique iteratively updates a person-specific face detector and facial landmark localization model which gradually improves accuracy.
- Our experiments show that the proposed pipeline significantly outperforms existing tracking methods that employ state-of-the-art face detection [48, 39, 28] and landmark localization [41, 8, 38, 22, 29] techniques in a tracking-by-detection manner on challenging videos.
- The proposed pipeline was used as a semi-automatic tool to annotate the videos of the 300 Videos in-the-wild (300-VW) Challenge [35][2].

2. Deformable Face Tracking Pipeline

The proposed pipeline, presented in Fig. 2, aims to perform accurate facial landmark tracking on all the frames

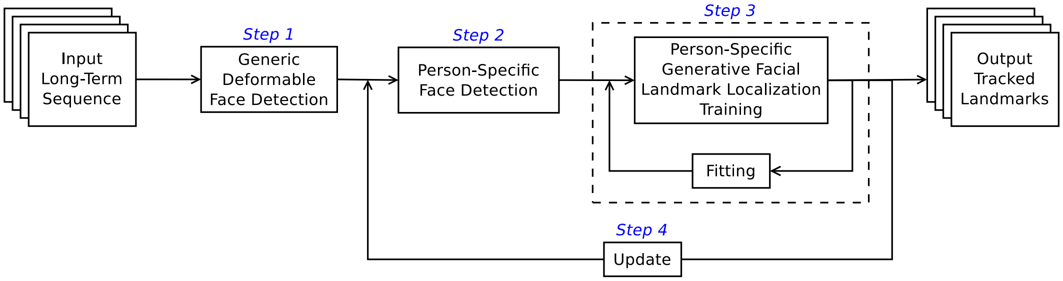


Figure 2: Overview of the pipeline. After applying a state-of-the-art generic deformable face detector (Step 1) to obtain an initial estimation of the shapes with minimal false positive detections, we iteratively (Step 4) train and fit a person-specific face detector (Step 2) and a person-specific generative deformable model (Step 3) that gradually improves the results.

of lengthy video sequences[3]. Let us denote the shape instance of a frame j as $\mathbf{s}^j = [\boldsymbol{\ell}_1^j, \ldots, \boldsymbol{\ell}_n^j]^T$, that consists of the Cartesian coordinates of n landmark points, denoted by $\boldsymbol{\ell}_i^j = [x_i^j, y_i^j]^T, \ \forall i = 1, \ldots, n$. Thus, given an input long-term sequence of N_0 frames, i.e. $\mathcal{I}^{N_0} = \{\mathbf{I}^1, \ldots, \mathbf{I}^{N_0}\}$, the pipeline aims to estimate the corresponding set of landmarks per frame, i.e. $\{\mathbf{s}^1, \ldots, \mathbf{s}^{N_0}\}$. This involves four discrete steps solved in an iterative manner which are described in detail in the following sections:

Step 1: Acquire an initial estimation of the landmarks per frame using state-of-the-art generic face detection and landmark localization techniques.

Step 2: Train and fit a person-specific face detector.

Step 3: Train, fit and iteratively update a person-specific generative deformable model for landmark localization.

Step 4: Update the person-specific detector and re-apply Steps (2) and (3).

Step 1: Generic Deformable Face Detection

The first step is to acquire an initial estimation of the landmarks per frame. We take advantage of recent state-of-the-art generic face detection and landmark localization methods in order to obtain acceptable initial estimations for as many frames as possible. There have also been proposals to perform face detection and landmark localization jointly [48, 13]. Nevertheless, the best results are achieved by using different methods for face detection and landmark localization (e.g. by combining [48] with [2, 38] or with [41]).

The results reported in the most well-known face detection benchmark (FDDB [20]) show that there are many face detection systems that achieve impressive true positive vs. false positive rates; some with publicly available implementations [28, 23, 43, 48, 26], others without [13, 42]. The biggest advantage of most of these methods is that they can be adjusted not to return any false positive detections, generally at the cost of decreasing the true positive rate. Since the detected faces will be used as initializations when training person-specific models, we require a face detector that returns a minimal number of false positives and is highly efficient. We have experimented with the majority of the top performing publicly available detectors and we empirically found that the implementation of [23] had the best ratio of performance and speed[4]. The decision threshold of the method in [23] was set so that virtually no false positives were returned whilst retaining a true positive rate of approximately 65%.

Existing state-of-the-art generic facial landmark localization methods can be separated in two categories: generative [37, 2, 38, 5, 6] and discriminative [41, 8, 22, 29]. Most methods are accompanied by publicly available implementations pre-trained on multiple databases which makes them easy to access and use. Based on the results reported on standard datasets of static images, we chose to use the open-source implementation[4] of the discriminative technique in [22], which utilizes an ensemble of regression trees and shows accurate real-time performance.

We wish to reiterate that any reliable face detection and landmark localization technique is suitable for this step, as long as the number of false positives is low. Moreover, as aforementioned, this step is equivalent to the current state-of-the-art for deformable face tracking in the literature. However, as we show in our experiments and in Fig. 1, it is inadequate in the case of arbitrary videos, mainly due to the small true positive rate of the detectors and the limited accuracy demonstrated by a generic deformable model as opposed to a person-specific one.

[3] We make the assumption that the person may leave the frame or that the camera will move, hence in many frames the face may not be visible.

[4] The DLib C++ Library provides open-source implementations of [22, 23, 16] in `http://dlib.net/`. The Menpo Project [1] (`http://www.menpo.org/`) provides open-source implementations of various deformable models, such as [37, 38, 5, 2, 41, 6], and interfaces easily with the DLib library. Furthermore, the deformable models of [28] and [48] show similar performance but they are more computationally expensive.

Step 2: Person-Specific Face Detection

We assume that the first step of the pipeline (Step 1) returns the detected landmarks $\{\mathbf{s}^1, \ldots, \mathbf{s}^{N_1}\}$ for a subset of the original frames $\mathcal{I}^{N_1} = \{\mathbf{I}^1, \ldots, \mathbf{I}^{N_1}\} \subseteq \mathcal{I}^{N_0}$ for which we obtained a correct detection. In the second step of the pipeline, we train a person-specific face detector using the previously returned detections by building a Deformable Part Model (DPM) [17, 48], which is the discriminative counterpart to the generative Pictorial Structures (PS) [18].

Let us define a tree structure $G = (V, E)$, where $V = \{v_1, \ldots, v_n\}$ is a set of vertices that correspond to n landmarks (parts) and there exists an edge $(v_i, v_j) \in E$ for each pair of connected landmarks. DPMs aim to learn a mixture of M different tree models ($G^m = (E^m, V^m), m = 1, \ldots, M$), each one consisting of a set of parameters that correspond to the appearance of each part and a set of parameters that describe the deformation of the part connections. The purpose of this mixture is to cover a range of different facial poses. DPMs learn the appearance and deformation parameters using a discriminative training procedure, typically via Support Vector Machines (SVM). Then, given an image $\mathbf{I}$, the cost function for the m-th mixture component is expressed as

$$\begin{aligned} C(\mathbf{s}|\mathbf{I}, m) &= A(\mathbf{s}|\mathbf{I}, m) + S(\mathbf{s}|\mathbf{I}, m) + a^m \\ A(\mathbf{s}|\mathbf{I}, m) &= \sum_{i=1}^{n} \mathbf{w}_i^{mT} \mathcal{F}(\boldsymbol{\ell}_i|\mathbf{I}) \\ S(\mathbf{s}|\mathbf{I}, m) &= \sum_{(v_i,v_j)\in E^m} \big(a_{ij}^m dx_{ij}^2 + b_{ij}^m dx_{ij} + \\ &\qquad\qquad + c_{ij}^m dy_{ij}^2 + d_{ij}^m dy_{ij}\big) \end{aligned} \tag{1}$$

where

- $A(\mathbf{s}|\mathbf{I}, m)$ is the appearance cost of placing each part $v_i^m \in V^m$ at the image location $\boldsymbol{\ell}_i$, measured as the mismatch between the learnt template (filter) $\mathbf{w}_i^m$ and the extracted image appearance $\mathcal{F}(\boldsymbol{\ell}_i|\mathbf{I})$. $\mathcal{F}$ denotes a feature vector extraction function (e.g. HoG [15], SIFT [27]) from the neighbourhood around location $\boldsymbol{\ell}_i$.
- $S(\mathbf{s}|\mathbf{I}, m)$ denotes the deformation cost when all the adjacent parts $(v_i^m, v_j^m) : i, j \in E^m$ are placed in locations $\boldsymbol{\ell}_i$ and $\boldsymbol{\ell}_j$ respectively. $dx_{ij} = x_i - x_j$ and $dy_{ij} = y_i - y_j$ are the relative locations (displacements) of the i-th part with respect to the j-th one.
- α^m is a scalar bias (prior) per mixture component.

The cost function of Eq. 1 can be expressed in a more convenient form using a dot product as

$$C(\mathbf{s}|\mathbf{I}, m) = \mathbf{w}_m^T \mathbf{y} \tag{2}$$

where $\mathbf{w}_m$ is the vector of the concatenated appearance and deformation parameters

$$\mathbf{w}_m = [\mathbf{w}_m^1, \ldots, \mathbf{w}_m^n, \ldots, a_{ij}^m, b_{ij}^m, c_{ij}^m, d_{ij}^m, \ldots, \alpha^m] \tag{3}$$

The final landmark locations are obtained by maximizing Eq. 2 with respect to $\mathbf{s}$ and m, as

$$C^*(\mathbf{I}) = \max_{m,\mathbf{s}} C(\mathbf{s}|\mathbf{I}, m) \tag{4}$$

This problem is solved in linear time with respect to the number of parts n, number of components M and image size, by employing an efficient dynamic programming algorithm based on the Generalized Distance Transform [18].

There are two main annotation settings for estimating the parameters of DPMs: weakly and strongly supervised.

Weakly Supervised Setting: In this setting, only the bounding boxes of the positive examples $\{\mathbf{s}^1, \ldots, \mathbf{s}^{N_1}\}$ and a set of negative examples are available. The number of parts n and mixtures M are defined a priori, while the part locations and the mixtures in the training set are considered as hidden (latent) information revealed during training [17].

By defining $\mathbf{z} = [m, \boldsymbol{\ell}_1^m, \ldots, \boldsymbol{\ell}_n^m]$ to be a latent variable vector, the goal is to learn a vector of parameters $\mathbf{w}_m = [\mathbf{w}_m^1, \ldots, \mathbf{w}_m^n]$. Since only one of the mixture tree models G^m can be activated, we define a general sparse feature vector $\mathbf{y}(\mathbf{z}) = [\mathbf{0}, \ldots, \mathbf{y}(\tilde{\mathbf{z}}), \ldots, \mathbf{0}]$, which is the score for the hypothesis $\tilde{\mathbf{z}} = [\boldsymbol{\ell}_1^m, \ldots, \boldsymbol{\ell}_n^m]$. By denoting the space of possible latent values for an example $\mathbf{q}$ as $\mathcal{Z}(\mathbf{q})$, the classifier that scores this example has the form $f_{\mathbf{w}_m}(\mathbf{q}) = \max_{\mathbf{z}\in\mathcal{Z}(\mathbf{q})} \mathbf{w}_m^T \mathbf{y}(\mathbf{q}, \mathbf{z})$. In order to find the parameters $\mathbf{w}_m$, given the set of N_1 training examples $\{(\mathbf{q}_1, y_1), \ldots, (\mathbf{q}_{N_1}, y_{N_1})\}$ and $y_i \in \{-1, 1\}$, which are images with a bounding box annotation, we minimize the SVM objective function using the standard hinge loss

$$C(\mathbf{q}) = \frac{1}{2}||\mathbf{w}_m||^2 + C\sum_{j=1}^{N_1} \max(0, 1 - y_j f_{\mathbf{w}_m}(\mathbf{q})) \tag{5}$$

which can be reformulated as

$$\begin{aligned} &\arg\min_{\mathbf{w}_m} \frac{1}{2}||\mathbf{w}_m||^2 + C\sum_{j=1}^{N_1} \max(0, 1 - y_j \mathbf{w}_m^T \mathbf{y}(\mathbf{q}, \mathbf{z}^*)) \\ &\text{s.t.} \quad \mathbf{z}^* = \max_{\mathbf{z}\in\mathcal{Z}(\mathbf{q})} \mathbf{w}_m^T \mathbf{y}(\mathbf{q}, \mathbf{z}) \end{aligned} \tag{6}$$

The minimization of the above cost function is highly non-convex, but becomes convex once the latent information is specified for the positive training examples. In [17], the authors propose an alternating optimization procedure, called latent-SVM. Specifically, by fixing $\mathbf{w}_m$, the highest scoring latent value for each positive example is determined. Then, by fixing the latent values for the positive set of examples, $\mathbf{w}_m$ is updated by minimizing the SVM cost of Eq. 5, using stochastic gradient descent.

Strongly Supervised Setting: Under a strongly supervised setting, we assume that (i) the mixture components of the training set are labelled, and (ii) the training set consists of images $\mathcal{I}^{N_1}$ with annotated landmarks, i.e.

$\{\mathbf{s}^1, \dots, \mathbf{s}^{N_1}\}$. Consequently, there are no hidden latent variables $\mathbf{z}_j$ to be estimated for each training sample $\mathbf{q}_j$ and only the model parameters $\mathbf{w}_m$ need to be learnt. That is

$$\begin{aligned} \underset{\mathbf{w}_m, \{\xi_j\}}{\arg\min} \quad & \frac{1}{2}\mathbf{w}_m^T\mathbf{w}_m + C\sum_{j=1}^{N_1}\xi_j \\ \text{s.t.} \quad & \forall\, \mathbf{q}_j \in \mathcal{C}^+,\ \mathbf{w}_m^T\mathbf{y}(\mathbf{q}_j, \mathbf{z}_j) \geq 1 - \xi_j \\ & \forall\, \mathbf{q}_j \in \mathcal{C}^-,\ \mathbf{w}_m^T\mathbf{y}(\mathbf{q}_j, \mathbf{z}_j) \leq -1 + \xi_j \\ & \forall\, k \in \mathcal{K}, w_k \leq 0 \end{aligned} \tag{7}$$

where $\mathcal{K}$ is the set of indices of $\mathbf{w}_m$ that correspond to the quadratic terms of the shape cost a_{ij}^m and c_{ij}^m. The above constraints state that the score of the positive examples should be larger than 1 (minus the small slack variable value ξ_j), while the score of the negative examples should be less than -1 (adding the small slack variable value ξ_j), for all part configurations and components. The last set of constraints guarantees that the deformation cost is a proper metric. The family of optimization problems that have the form of Eq. 7 are referred to as structural SVMs. Similar to [48], we employ a modified coordinate descent method, which allows the incorporation of the last set of negativity constraints. This method iterates between finding $\mathbf{w}_m$ in the dual space and mining for violated constraints according to the current estimate of $\mathbf{w}_m$ until convergence is met.

Output Detections: In our experiments, we employ a strongly supervised DPM trained on the shapes that are returned from the generic deformable face detector of the previous step, i.e. $\{\mathbf{s}^1, \dots, \mathbf{s}^{N_1}\}$. Of course, a weakly supervised DPM can also be applied, however, in this case, the final loop of the pipeline (Step 4) that updates the person-specific DPM would have no effect. This is because the bounding boxes cannot be improved from the generative landmark localization procedure of Step 3. Finally, the person-specific nature of the DPM ensures that the returned true positive rate will reamin extremely high, while keeping the false positive rate close to zero. This means that the set of N_2 output shapes for which the DPM returns a correct detection will be close to the number of the initial frames, i.e. $N_1 < N_2 \approx N_0$.

Step 3: Person-Specific Generative Facial Landmark Localization

Due to the use of a highly flexible tree-structure, the strongly supervised DPMs do not achieve state-of-the-art facial landmark localization performance. Furthermore, since the method is discriminative, it is very sensitive to inaccurate landmark detections. For this step, we employ a state-of-the-art generative deformable model that is able to correct inaccurately localized landmarks. Motivated by recent developments in automatic construction of generative deformable models [14, 31, 44, 7], we use the state-of-the-art part-based Active Appearance Model (AAM) of [38], referred to as the Gauss-Newton DPM (GN-DPM), and iteratively improve the appearance model.

GN-DPM [38] is a generative statistical model of shape and appearance that is able to recover a parametric description of a face via Gauss-Newton optimization. By applying Generalized Procrustes Analysis to align the shapes $\{\mathbf{s}^1, \dots, \mathbf{s}^{N_2}\}$ obtained from Step 2 and using Principal Component Analysis (PCA), we build an orthonormal basis of n_S eigenvectors $\mathbf{U}_S \in \mathbb{R}^{2n \times n_S}$ plus a mean shape $\bar{\mathbf{s}}$. This linear shape model can be used to generate shape instances as $\mathbf{s}(\mathbf{p}) = \bar{\mathbf{s}} + \mathbf{U}_S\mathbf{p}$, where $\mathbf{p} = [p_1, \dots, p_{n_S}]^T$ is the vector of shape parameters. Similarly, in order to build the appearance model, we first sample all the training images $\mathcal{I}^{N_2} = \{\mathbf{I}^1, \dots, \mathbf{I}^{N_2}\}$ in patches centred around each landmark location using the function $\mathcal{F}(\boldsymbol{\ell}_i^j|\mathbf{I}^j)$, $\forall j = 1, \dots, N_2$, $\forall i = 1, \dots, n$ defined for Eq. 1 and concatenate these vectors in order to acquire an $na \times 1$ vectorized part-based appearance representation $\mathbf{a}(\mathbf{s}|\mathbf{I}) = [\mathcal{F}(\boldsymbol{\ell}_1|\mathbf{I})^T, \mathcal{F}(\boldsymbol{\ell}_2|\mathbf{I})^T, \dots, \mathcal{F}(\boldsymbol{\ell}_n|\mathbf{I})]^T$ for all images. Then, the linear appearance model of GN-DPM is trained by performing PCA on the set of part-based appearance vectors of all training images that results in a subspace of n_A eigenvectors $\mathbf{U}_A \in \mathbb{R}^{na \times n_A}$ and the mean appearance $\bar{\mathbf{a}}$. This model can be used to synthesize shape-free appearance instances, as $\mathbf{a}(\boldsymbol{\lambda}) = \bar{\mathbf{a}} + \mathbf{U}_A\boldsymbol{\lambda}$, where $\boldsymbol{\lambda} = [\lambda_1, \dots, \lambda_{N_A}]^T$ is the vector of appearance parameters. Given a test image $\mathbf{I}$, the optimization problem of GN-DPM employs an ℓ_2^2 norm and is expressed as

$$\underset{\mathbf{p}, \boldsymbol{\lambda}}{\arg\min} \|\mathbf{a}(\mathbf{s}(\mathbf{p})|\mathbf{I}) - \bar{\mathbf{a}} - \mathbf{U}_A\boldsymbol{\lambda}\|^2 \tag{8}$$

This can be efficiently solved using the Gauss-Newton algorithm. Due to limited space, the optimization process is not included. For more details, please refer to [38].

The goal of this step of the pipeline is to construct an optimal generative appearance model that best describes the appearance variation that is present in the video sequence. For that purpose, we formulate an iterative optimization problem that aims to minimize the mean GN-DPM fitting ℓ_2^2 norm of Eq. 8 over all the frames of the video. In order to facilitate this procedure, we assume that the initial shape and appearance models that are utilized are trained using a database with static images of generic faces. Let us denote the generic shape basis as $\mathbf{U}_S$, the generic appearance basis as $\mathbf{U}_A$ and a person-specific appearance basis as $\mathbf{B}_A$. Then, the minimization problem is expressed as

$$\begin{aligned} \underset{\bar{\mathbf{a}}, [\mathbf{U}_A\ \mathbf{B}_A], \mathbf{p}^i, \boldsymbol{\lambda}^i}{\arg\min} \quad & \frac{1}{N_2}\sum_{i=1}^{N_2} \|\mathbf{a}(\mathbf{s}^i(\mathbf{p}^i)|\mathbf{I}^i) - \bar{\mathbf{a}} - [\mathbf{U}_A\ \mathbf{B}_A]\boldsymbol{\lambda}^i\|^2 \\ \text{s.t.} \quad & [\mathbf{U}_A\ \mathbf{B}_A]^T[\mathbf{U}_A\ \mathbf{B}_A] = \mathbf{I}_{eye} \end{aligned} \tag{9}$$

where $\mathbf{I}_{eye}$ denotes the identity matrix. This procedure iteratively trains a new person-specific appearance basis $\mathbf{B}_A$

based on the current estimate of the N_2 shapes, combines the generic appearance model $\mathbf{U}_A$ with $\mathbf{B}_A$ so that they are orthogonal ($[\mathbf{U}_A \ \mathbf{B}_A]^T [\mathbf{U}_A \ \mathbf{B}_A] = \mathbf{I}_{eye}$) and then re-estimates the parameters $\{\mathbf{p}^i, \boldsymbol{\lambda}^i\}, \ i = 1, \ldots, N_2$ by minimizing the ℓ_2^2 norm for each frame. Thus, the optimization is solved in an alternating manner in two steps:

(1) Fix $\{\mathbf{p}^i, \boldsymbol{\lambda}^i\}$ *and minimize for* $\{\bar{\mathbf{a}}, [\mathbf{U}_A \ \mathbf{B}_A]\}$*:* Having estimated the current shapes for each frame $i = 1, \ldots, N_2$, we build a person-specific appearance subspace $\mathbf{B}_A$ from $\{\mathbf{a}(\mathbf{s}^i(\mathbf{p}^i)|\mathbf{I}^i)\}$ and combine it with the generic appearance model $\mathbf{U}_A$ in order to create an updated orthogonal subspace, thus satisfy the constraint $[\mathbf{U}_A \ \mathbf{B}_A]^T [\mathbf{U}_A \ \mathbf{B}_A] = \mathbf{I}_{eye}$.

(2) Fix $\{\bar{\mathbf{a}}, [\mathbf{U}_A \ \mathbf{B}_A]\}$ *and minimize for* $\{\mathbf{p}^i, \boldsymbol{\lambda}^i\}$*:* Having created the orthogonal basis that combines the generic and person-specific appearance variation, we now aim to estimate the shape and appearance parameters per frame. Based on Eq. 8, this is done by solving $\arg\min_{\mathbf{p}^i, \boldsymbol{\lambda}^i} \|\mathbf{a}^i(\mathbf{s}^i(\mathbf{p}^i)|\mathbf{I}^i) - \bar{\mathbf{a}} - [\mathbf{U}_A \ \mathbf{B}_A]\boldsymbol{\lambda}^i\|^2, \ \forall i = 1, \ldots, N_2$ using the Gauss-Newton optimization [38].

The above iterative procedure gradually improves the appearance model and therefore also improves the detected landmarks. Note that other state-of-the-art generative methods [37, 2, 5] could also be used. Our experiments showed that, given the person-specific nature of the generative model, only a few iterations are adequate in order to obtain accurate results.

Step 4: Person-Specific Face Detection Update

The output of Steps 2 and 3 is a set of tracked shapes $\{\mathbf{s}^1, \ldots, \mathbf{s}^{N_2}\}$ that correspond to each of the frames in $\mathcal{I}^{N_2} = \{\mathbf{I}^1, \ldots, \mathbf{I}^{N_2}\}$. These frames are the ones for which a correct detection was returned from Step 2. As explained in Step 2, due to the person-specific setting of the DPM, N_2 must be very close if not equal to N_0. The next step is to further improve the fitting accuracy and increase the number of true positive detections, if required. This is done by updating the person-specific DPM with the acquired set of tracked shapes, which makes the generative deformable model more expressive. Given the fitted shapes $\{\mathbf{s}^1, \ldots, \mathbf{s}^{N_2}\}$, the DPM can be updated in two ways: (i) retrain it under a strongly supervised setting (Step 2), or (ii) update the parameters of the existing model using the passive-aggressive algorithm of [47]. We have experimentally verified that both have similar performance with the latter being considerably faster.

This update step dramatically improves the true positive rate of the detector, as well as the accuracy of the deformable model. Our experiments showed that one such iteration is enough for the vast majority of videos. The average recall (i.e., true positive rate) we achieve with the proposed procedure is more than 98% with almost an 0% of false positive rate. However, for a small number of frames the person-specific model may not return a face. We treat those cases as a tracking problem and assume a first order Markov dependency. That is, we deal with those frames by initializing their shapes with the shapes of the previous frames and applying the person-specific GN-DPM.

3. Experiments

In this section, we show that our proposed pipeline outperforms all state-of-the-art tracking methods by a substantial margin. We also explain how the proposed framework was employed as a semi-automatic tool to annotate the majority of the videos of the 300-VW Challenge [35][2].

3.1. Dataset and Implementation

We found that the majority of the videos currently used for demonstrating face tracking results are both easy and extremely short. Therefore, we carried out experiments on two video categories: (1) videos for which accurate face detection can be achieved by most existing detectors, and (2) very challenging videos where even state-of-the-art detectors fail to detect the face in the majority of frames. The motivation behind this classification is that although existing methods can perform quite well for the sequences of category 1, in category 2 we find a noticeable deterioration in the number of true positives and thus the point-to-point error. The input videos were manually annotated with 68 facial landmark points, using the standard mark-up of Multi-PIE [19].

Category 1: Includes 14 videos (about 30,000 frames in total) that are selected from the testset of the 300-VW Challenge[2] (scenarios 1 and 2). These videos exhibit large variations in pose, resolution and illumination.

Category 2: Includes 2 very challenging videos with 3058 frames in total. The videos exhibit very challenging conditions, even for the state-of-the-art detectors and trackers. Roughly 10% of the frames were annotated, which was deemed a sufficient number given the nature of the videos. We note that these videos were so difficult that even manual annotation was challenging.

Implementation details for our pipeline: For Step 1 (i.e., [23, 22]) and Step 3 (i.e., GN-DPM [38]), we used the publicly available implementations in Menpo[4]. The generic bases of GN-DPM are trained on the iBUG and HELEN datasets [32, 30], approximately 500 images in total. The person-specific update of GN-DPM bases is conducted using 100 randomly selected frames of the video. The GN-DPM is optimized using a multi-scale Gaussian pyramid of 2 levels. The number of employed shape components (n_S) is 3 and 12 for the low and high pyramidal levels respectively. The number of appearance components (n_A) is 50 and 100 respectively and dense SIFT features were used. Finally, the DPM of Step 2 is trained in a strongly supervised manner.

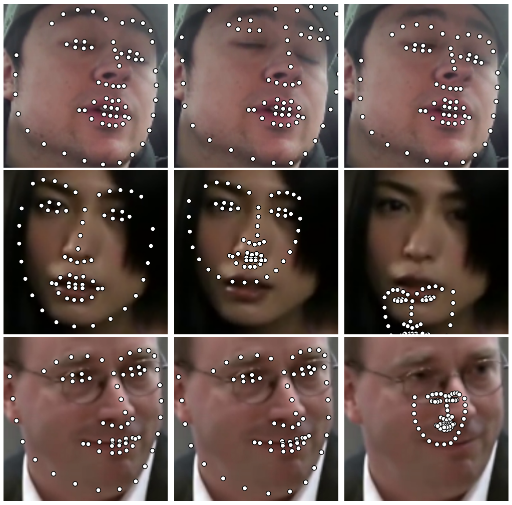


Figure 3: Indicative qualitative results. *Left*: Ours. *Middle*: Deformable detector ([23] + [22]). *Right*: SPOT [47]

3.2. Pipeline Experimental Analysis

Herein, we measure the impact of each step of our pipeline. We report the mean point-to-point error normalized with the face size (i.e., the standard metric that was proposed in [48]) for each category of videos. Figure 5 shows the output of each stage of the pipeline by applying a single loop of Step 4. For the first category of videos, the generic face detector and landmark localization method produced favourable initial results. The person-specific DPM increased the true positive rate and the person-specific GN-DPM of Step 3 improves the landmark localization accuracy. That is, for the very low error value of 0.03, the percentage increased from 63% to 72%. Note that a single run of the proposed system is adequate since the update of Step 4 does not change the results. By inspecting the results of the second category videos, we see that the effect of the pipeline is much more evident. That is, the generic face detector failed to detect faces in more than 25% of frames, in contrast to the final iteration of our proposed pipeline which managed to reach 100% true positive rate and also improved the facial fitting accuracy. Nevertheless, the update of Step 4 does not contribute much, due to the difficulty of the videos.

3.3. Comparison with state-of-the-art

Figure 4 compares our pipeline with other state-of-the-art frameworks. Specifically, we compare with methods that use the standard approach of generic face detection plus generic facial landmark localization. We selected two such representative methods: (1) the method used to initialize our pipeline in Step 1 which consists of [23] followed by [22], and (2) the publicly available implementation of Chehra [9]. For both these methods, when the face detector fails to return a result we assume a 1st order Markov dependency and initialize from the most recent returned detection. Moreover, we also compare with methods that employ state-of-the-art tracking plus facial landmark localization. For this category of techniques, we selected some of the best performing tracking-by-detection trackers that have appeared recently and that provide open-source implementations, i.e., SPOT [47], FCT [46] and Correlation tracker (Correl) [16]. For these trackers, the ground truth bounding box of the first frame was provided as the initial input. The generic landmark localization method used in combination with these trackers is the state-of-the-art method of [22].

Figure 4 shows cumulative error distribution curves based on the 49 internal points of the face (excluding the points of the boundary), as this is the mark-up returned by Chehra. Our pipeline significantly outperforms the rest of the trackers in both categories. For the very challenging videos of category 2, the state-of-the-art trackers fail to detect several frames. However, our proposed pipeline manages to maintain a high true positive rate.

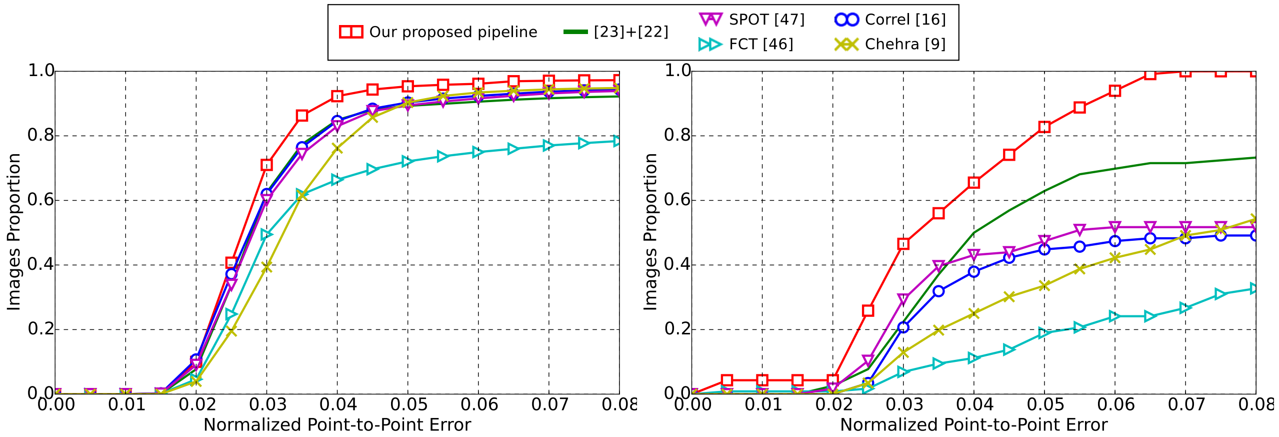
3.4. Semi-Automatic Annotation Tool

The proposed pipeline was used as a semi-automatic tool in order to annotate 86 videos for the 300-VW Challenge[2]. The annotation procedure involved the following five steps:

1. Download videos from the Internet (e.g. Youtube).
2. Apply the proposed pipeline as described in Sec. 3.1.
3. Manually correct the annotation for every eighth frame of each video, if required.
4. Use the corrected frames from the previous step to train and fit a GN-DPM per video (Step 3 of pipeline).
5. Visually inspect the results and finally manually correct the annotations, if required.

The above procedure manages to return accurate annotations in the majority of cases. Therefore, the required amount of manual correction was limited given the large number of frames and the difficulty of the videos. In total, the manual intervention of steps 3 and 5 required 837 hours of human labour. The annotations were performed by trained annotators using the web-based landmarking tool that is provided by the Menpo Project[5]. In order to practically measure the benefit of the proposed semi-annotation tool, we manually annotated 6 out of the 86 videos, which required approximately 260 working hours. Thus, we estimate that the manual annotations of the frames for all 86 videos would have taken around 3727 human working hours, i.e., approximately $4.5\times$ more than the proposed procedure. These numbers highlight the effectiveness of our pipeline for the semi-assisted annotation of large-scale databases of videos. The pipeline and the annotation tool

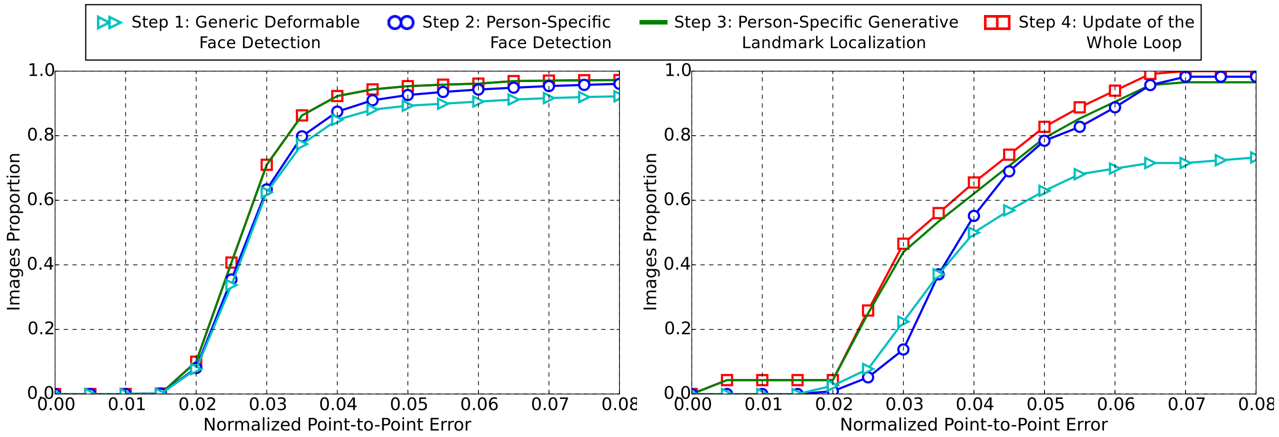
[5] https://www.landmarker.io/



(a) Video category 1 (average difficulty)

(b) Video category 2 (very challenging)

Figure 4: Comparison of our pipeline with state-of-the-art techniques, evaluated on 49 facial landmark points.



(a) Video category 1 (average difficulty)

(b) Video category 2 (very challenging)

Figure 5: Illustration of the contribution of each step of the proposed pipeline, evaluated on 49 facial landmark points.

were built within the Menpo Project [1] and will be publicly available soon. The annotation procedure was facilitated using the cloud service of [24].

4. Conclusions

In this paper we presented the first, to the best of our knowledge, pipeline that can perform long-term deformable face tracking in challenging videos. The pipeline takes advantage of generic face detection and landmark localization algorithms to iteratively train powerful and accurate person-specific face detectors and landmark localization techniques. The pipeline was used as a semi-automatic tool to annotate most of the videos of the 300-VW Challenge[2].

Acknowledgements

Grigorios G. Chrysos is supported by the EPSRC project EP/L026813/1 Adaptive Facial Deformable Models for Tracking (ADAManT). Epameinondas Antonakos is funded by the EPSRC project EP/J017787/1 (4D-FAB). Stefanos Zafeiriou is partially supported by the 4D-FAB and ADAManT projects. Patrick Snape is funded by a DTA from Imperial College London and by a Qualcomm Innovation Fellowship.

References

[1] J. Alabort-i-Medina, E. Antonakos, J. Booth, P. Snape, and S. Zafeiriou. Menpo: A comprehensive platform for parametric image alignment and visual deformable models. In *ACMMM*. ACM, 2014.

[2] J. Alabort-i-Medina and S. Zafeiriou. Bayesian active appearance models. In *CVPR*, 2014.

[3] J. Alabort-i-Medina and S. Zafeiriou. Unifying holistic and parts-based deformable model fitting. In *CVPR*, pages 3679–3688, 2015.

[4] E. Antonakos, J. Alabort-i-Medina, G. Tzimiropoulos, and S. Zafeiriou. Hog active appearance models. In *ICIP*, October 2014.

[5] E. Antonakos, J. Alabort-i-Medina, G. Tzimiropoulos, and S. Zafeiriou. Feature-based lucas-kanade and active appearance models. *IEEE TIP*, 24(9):2617–2632, 2015.

[6] E. Antonakos, J. Alabort-i-Medina, and S. Zafeiriou. Active pictorial structures. In *CVPR*, 2015.

[7] E. Antonakos and S. Zafeiriou. Automatic construction of deformable models in-the-wild. In *CVPR*, 2014.

[8] A. Asthana, S. Zafeiriou, S. Cheng, and M. Pantic. Robust discriminative response map fitting with constrained local models. In *CVPR*, 2013.

[9] A. Asthana, S. Zafeiriou, S. Cheng, and M. Pantic. Incremental face alignment in the wild. In *CVPR*, 2014.

[10] A. Asthana, S. Zafeiriou, G. Tzimiropoulos, S. Cheng, M. Pantic, et al. From pixels to response maps: discriminative image filtering for face alignment in the wild. *IEEE T-PAMI*, 37(6):1312–1320, 2015.

[11] P. N. Belhumeur, D. W. Jacobs, D. Kriegman, and N. Kumar. Localizing parts of faces using a consensus of exemplars. In *CVPR*, 2011.

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

[13] D. Chen, S. Ren, Y. Wei, X. Cao, and J. Sun. Joint cascade face detection and alignment. In *ECCV*. Springer, 2014.

[14] X. Cheng, S. Sridharan, J. Saragih, and S. Lucey. Rank minimization across appearance and shape for aam ensemble fitting. In *ICCV*, 2013.

[15] N. Dalal and B. Triggs. Histograms of oriented gradients for human detection. In *CVPR*, 2005.

[16] M. Danelljan, G. Häger, F. S. Khan, and M. Felsberg. Accurate scale estimation for robust visual tracking. In *BMVC*, 2014.

[17] P. F. Felzenszwalb, R. B. Girshick, D. McAllester, and D. Ramanan. Object detection with discriminatively trained part-based models. *IEEE T-PAMI*, 32(9):1627–1645, 2010.

[18] P. F. Felzenszwalb and D. P. Huttenlocher. Pictorial structures for object recognition. *IJCV*, 61(1):55–79, 2005.

[19] R. Gross, I. Matthews, J. Cohn, T. Kanade, and S. Baker. Multi-pie. *IMAVIS*, 28(5):807–813, 2010.

[20] V. Jain and E. G. Learned-Miller. Fddb: A benchmark for face detection in unconstrained settings. *UMass Amherst Technical Report*, 2010.

[21] Z. Kalal, K. Mikolajczyk, and J. Matas. Tracking-learning-detection. *IEEE T-PAMI*, 34(7):1409–1422, 2012.

[22] V. Kazemi and J. Sullivan. One millisecond face alignment with an ensemble of regression trees. In *CVPR*, 2014.

[23] D. E. King. Max-margin object detection. *arXiv preprint arXiv:1502.00046*, 2015.

[24] V. Koukis, C. Venetsanopoulos, and N. Koziris. ˜ okeanos: Building a cloud, cluster by cluster. *Internet Computing, IEEE*, 17(3):67–71, 2013.

[25] V. Le, J. Brandt, Z. Lin, L. Bourdev, and T. S. Huang. Interactive facial feature localization. In *ECCV*. Springer, 2012.

[26] J. Li and Y. Zhang. Learning surf cascade for fast and accurate object detection. In *CVPR*, 2013.

[27] D. G. Lowe. Object recognition from local scale-invariant features. In *ICCV*, 1999.

[28] M. Mathias, R. Benenson, M. Pedersoli, and L. Van Gool. Face detection without bells and whistles. In *ECCV*. Springer, 2014.

[29] S. Ren, X. Cao, Y. Wei, and J. Sun. Face alignment at 3000 fps via regressing local binary features. In *CVPR*, 2014.

[30] C. Sagonas, E. Antonakos, G. Tzimiropoulos, S. Zafeiriou, and M. Pantic. 300 faces in-the-wild challenge: Database and results. *IMAVIS*, 2015. in press.

[31] C. Sagonas, Y. Panagakis, S. Zafeiriou, and M. Pantic. Raps: Robust and efficient automatic construction of person-specific deformable models. In *CVPR*, 2014.

[32] C. Sagonas, G. Tzimiropoulos, S. Zafeiriou, and M. Pantic. 300 faces in-the-wild challenge: The first facial landmark localization challenge. In *ICCV'W*, 2013.

[33] C. Sagonas, G. Tzimiropoulos, S. Zafeiriou, and M. Pantic. A semi-automatic methodology for facial landmark annotation. In *CVPR'W*, June 2013.

[34] E. Sariyanidi, H. Gunes, and A. Cavallaro. Automatic analysis of facial affect: A survey of registration, representation and recognition. *IEEE T-PAMI*, 37(6):1113–1133, 2015.

[35] 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 *ICCV'W*, 2015.

[36] Y. Taigman, M. Yang, M. Ranzato, and L. Wolf. Deepface: Closing the gap to human-level performance in face verification. In *CVPR*, 2014.

[37] G. Tzimiropoulos, J. Alabort-i-Medina, S. Zafeiriou, and M. Pantic. Generic active appearance models revisited. In *ACCV*. Springer, 2013.

[38] G. Tzimiropoulos and M. Pantic. Gauss-newton deformable part models for face alignment in-the-wild. In *CVPR*, 2014.

[39] P. Viola and M. J. Jones. Robust real-time face detection. *IJCV*, 57(2):137–154, 2004.

[40] L. Wolf, T. Hassner, and I. Maoz. Face recognition in unconstrained videos with matched background similarity. In *CVPR*, 2011.

[41] X. Xiong and F. De la Torre. Supervised descent method and its applications to face alignment. In *CVPR*, 2013.

[42] J. Yan, Z. Lei, L. Wen, and S. Z. Li. The fastest deformable part model for object detection. In *CVPR*, 2014.

[43] B. Yang, J. Yan, Z. Lei, and S. Z. Li. Aggregate channel features for multi-view face detection. In *IJCB*, 2014.

[44] L. Zafeiriou, E. Antonakos, S. Zafeiriou, and M. Pantic. Joint unsupervised face alignment and behaviour analysis. In *ECCV*. Springer, 2014.

[45] S. Zafeiriou, C. Zhang, and Z. Zhang. A survey on face detection in the wild: past, present and future. *CVIU*, 138:1–24, September 2015.

[46] K. Zhang, L. Zhang, and M.-H. Yang. Fast compressive tracking. *IEEE T-PAMI*, 36(10):2002–2015, 2014.

[47] L. Zhang and L. van der Maaten. Preserving structure in model-free tracking. *IEEE T-PAMI*, 36(4):756–769, 2014.

[48] X. Zhu and D. Ramanan. Face detection, pose estimation, and landmark localization in the wild. In *CVPR*, 2012.