Composite recurrent network with internal denoising for facial alignment in still and video images in the wild

Decky Aspandi *, Oriol Martinez, Federico Sukno, Xavier Binefa

Department of Information and Communication Technologies, Pompeu Fabra University, Barcelona, Spain

A R T I C L E   I N F O
Article history:
Received 21 October 2019
Received in revised form 20 November 2020
Accepted 21 April 2021
Available online 26 April 2021

Keywords:
Facial alignment
Facial tracking
Temporal modeling
Internal denoising

A B S T R A C T
Facial alignment is an essential task for many higher level facial analysis applications, such as animation, human activity recognition and human - computer interaction. Although the recent availability of big datasets and powerful deep-learning approaches have enabled major improvements on the state of the art accuracy, the performance of current approaches can severely deteriorate when dealing with images in highly unconstrained conditions, which limits the real-life applicability of such models. In this paper, we propose a composite recurrent tracker with internal denoising that jointly address both single image facial alignment and deformable facial tracking in the wild. Specifically, we incorporate multilayer LSTMs to model temporal dependencies with variable length and introduce an internal denoiser which selectively enhances the input images to improve the robustness of our overall model. We achieve this by combining 4 different sub-networks that specialize in each of the key tasks that are required, namely face detection, bounding-box tracking, facial region validation and facial alignment with internal denoising. These blocks are endowed with novel algorithms resulting in a facial tracker that is both accurate, robust to in-the-wild settings and resilient against drifting. We demonstrate this by testing our model on 300-W and Menpo datasets for single image facial alignment, and 300-VW dataset for deformable facial tracking. Comparison against 20 other state of the art methods demonstrates the excellent performance of the proposed approach.

© 2021 The Author(s). Published by Elsevier B.V. This is an open access article under the CC BY license (http://creativecommons.org/licenses/by/4.0/).

1. Introduction

The human face is arguably one of the most important deformable objects for analysis in real world applications, such as facial animation, human activity recognition and human - computer interaction [1]. Facial alignment, which aims to detect a set of facial landmark positions, is essential for the high level analysis required in those applications to function well [2]. Currently, the development of facial alignment models is growing rapidly with the availability of large facial landmarked datasets such as 300-W [3] and Menpo [4]. This has made it possible the development of powerful deep learning models that have pushed forward the alignment accuracy and are considered the current state of the art [5,6].

However the performance of current facial alignment models can severely deteriorate when dealing with images in highly unconstrained conditions, e.g. extreme pose or illumination changes, large occlusions [7] or, in general, whenever the test images can be considered to show less favorable conditions than those available for training. In other words, we may say that under challenging conditions, test images contain some form of distortion or noise that will impair the performance of facial alignment models. This limits the real-life applicability of such models to in-the-wild images which naturally contain such challenging conditions [8,9]. While there have been attempts to improve the robustness of facial alignment models to target in-the-wild data [7,10], most of them have done so without modeling the effects of noise in their formulation. Nevertheless, understanding and incorporating such effects within the model training has proven beneficial to improve performance in other facial analysis tasks [8,11,12].

A very important aspect of facial alignment algorithms is their ability to operate reliably in the presence of image sequences (i.e. video data). In such cases, the problem becomes more challenging as there is the need for persistent stability to keep track of the facial features throughout the whole sequence [13]. Progress on facial tracking has been relatively slower when compared to single image facial alignment, and it has been less influenced by deep learning models [14]. Furthermore, the majority of currently available trackers make little use of temporal information: most of them process each frame independently and rely on doing so with sufficient precision to achieve tracking-like performance [15,16], or convert single-image facial alignment to perform tracking by using the results of the previous frame as initialization [17–19]. However, the latter approach makes the models vulnerable to drifting since they are not designed specifically to track [14]. In contrast, other trackers incorporate some
temporal modeling, but they are mostly limited to the adjacent frames [20,21]. All of the above suggests that current facial trackers may not take full advantage of the temporal information contained in video sequences [22].

In this paper we present a composite end to end facial tracking model which jointly addresses both single image and video sequence alignment. We introduce a robust facial landmark estimator equipped with an internal denoiser autoencoder to selectively enhance the images on a case-by-case basis to boost the accuracy of single-image landmark estimation. Temporal information is handled by means of internal Long Short Term Memory (LSTM) layers that allows modeling of short and long temporal dependencies between frames when video is available. Combination of these two main modules results in an algorithm that produces very accurate facial alignment and is also resilient against drifting, leading to a robust facial tracker.

The contributions of this paper are as follow:

1. We present a unified approach for both single image alignment and facial tracking with our joint robust facial alignment and tracker.
2. Our model employs an internal image denoising network to obtain an enhanced intermediate facial image that helps to improve the accuracy of landmark localization.
3. We incorporate temporal modeling between frames using internal multilayer LSTMs, which unlike other approaches, allow to consider time dependencies at various time scales.
4. We achieve state of the art results for both targeted tasks: Single Image Alignment (on 300-W [3] and Menpo [4] datasets), and Deformable Facial Tracking on the wild (using 300-VW dataset [2]).
5. We investigate the impact of the considered temporal scale for tracking as well as the impact of the internal denoiser in the overall accuracy of our model.

Preliminary results of landmark detection by denoising autoencoder networks and composite recurrent tracking were presented in [19,23]. The rest of the paper is organized as follows: Section 2 describes the related work in context of single image facial alignment and facial tracking. In Section 3, we explain our Denoised Composite Recurrent Tracker, which consists of multiple sub-networks operating in tandem and merged using our novel tracking algorithm. Section 4 reports our experiments on both Single Image Facial Alignment and Deformable Facial Tracking. Finally in Section 5 we derive our conclusions.

2. Related work

In this section, we describe the prior works related to the two fundamental problems addressed by our approach, namely Single Image Facial Alignment and Facial Tracking.

2.1. Single image facial alignment

Face alignment has attracted considerable attention due to its importance for several applications, such as face recognition [24], head pose estimation [25], facial reenactment [26], emotion recognition [67, 68] and others [66]. This task is generally conducted by estimating a predefined set of landmark positions which provide a structure representation of the facial geometry. Traditionally, facial landmarks have been estimated using either shape and appearance models [27,28] or regression based models [29,30], with the latter offering some advantage due to their computational efficiency [31]. Recent examples of regression-based models include the work of Kazemi and Sullivan [32], who use a cascade of regression functions to efficiently regress the landmark locations, and Zhu et al. [33] who refined the regressors cascade with coarse to fine search.

The recent availability of large datasets such as 300-W [3] and Menpo [4] allows for large scale data modeling and development of deep learning-based models that benefit from these load of data. For instance, the work of Bulat and Tzimiropoulos [5] used multiple hour-glass shaped convolutional networks with heatmap-guided layers to predict the facial landmarks, while the Zadeh et al. [34] introduce multiple patch experts to adjust each of their final landmark estimations, which has been incorporated in the Openface framework [35]. Zhang et al. [36] in addition, enforced the facial attributes as auxiliary features to their cascaded convolutional network to improve landmark estimates. These deep learning models currently hold the state of the art accuracy both on the single image facial alignment [3] and on the facial landmark tracking tasks [2].

Despite the maturity of current facial alignment models, there are still challenges for them when targeting images with large appearance variations due to heavy occlusion, severe pose or illuminations conditions, etc. [7]. This is especially relevant in real world application deployed in highly unconstrained settings [8]. Thus, there has been growing interest to improve the performance of facial models under such settings, including efforts such as iterative initialization of regression cascades to minimize the impact of outliers [10] or the combination of data- and model-driven estimators to enhance the robustness of landmark detection [7]. However, most efforts have focused on building robust feature extractors with the assumption that the model will be able to discard the noise and select only the meaningful features.

In contrast, we address this problem from a different perspective: we focus on modeling image noise by means of our internal denoiser network, which is conceived as an auxiliary block that aims to automatically enhance the quality of the input image to improve the accuracy of landmark localization. This strategy can be justified by recent findings such as those from Dong et al. [11], who aggregated different styles of images using Generative Adversarial Networks to improve their landmark estimates, revealing that even a slight color style variations in a given facial images (which is otherwise kept fixed) can impact the accuracy of landmark estimations. Similar findings have been reported in other facial analysis tasks, such as those from Zhou et al. [9], who evaluated several facial detection models with synthetic noise, and Goswami et al. [12] who investigated the robustness of facial classification to synthetically distorted images. The findings in these reports have led us to hypothesize that modeling certain types of noise may help to characterize the behavior of facial analysis systems with unconstrained images and that incorporating such noise modeling into the training process could improve the robustness of their results.

2.2. Facial tracking

Currently, the most popular facial tracking technique is Tracking by Detection, which consists of performing facial detection and landmark localization at each frame. Some examples of this strategy include the work from Uricar et al. [37] and the OpenFace tracker. Uricar et al. use tree-based Deformable Part Models (DPM) for facial landmark detection and localization with Kalman Filter smoothing, while the OpenFace tracker [35] uses the multiple convolutional experts method described in the previous Section [34] initialized with the bounding box from a Multi-Task Cascaded Neural Network face detector [38].

Other tracking methods perform face detection only in the first frame and then apply facial landmark localization using the fitting result from the previous frame as initialization. One such example is the work from Xiao et al. [20] which adopts a multi-stage regression-based approach to initialize the shape of landmarks with high semantic meaning. Other examples include the work from Raja et al. [21] which combines a global shape model with sets of response maps for different head angles indexed on the shape model parameters and the work from Wu et al. [39] who apply shape augmented regression. There are also hybrid approaches that combine tracking by detection and initialization based on the latest fitting result. Among
these, combinations of Coarse-To-Fine Shape Search (CFSS) [33] landmark localizer with multiple general-object trackers have shown to perform particularly well [14].

However, all methods derived from tracking by detection share the limitation of not considering the temporal information contained in video sequences. Furthermore, it is difficult to obtain consistent initializations from most face detectors, which tends to reduce the final landmark localization accuracy [40]. Some approaches try to mitigate this problem by including the information from the adjacent frames to capture short temporal dependencies. For example, Yang et al. [17] used time series regression on pairs of adjacent frames, which led them to achieve the best result reported so far on the original challenge in the 300 Videos in the Wild dataset (300-VW) [2], which is the largest deformable facial tracking benchmark.

With the recent growth of facial landmark datasets, such as 300-W [3], Menpo [41], 300-VW and LS3D-W [5], current methodologies on facial analysis started to shift from systems based on handcrafted features towards incorporating deep learning architectures [4,42]. Rapid progress can be seen on the development of various convolutional architectures as the main spatial feature extractor used on both facial detection [38] and landmark localization models [5,43], achieving state of the art accuracy. In spite of this, localization is still mainly performed on every single frame, without taking into account the temporal information.

On the other hand, introduction of recurrent neural networks (RNN), especially Long Short Term Memory (LSTM) [44], has allowed incorporating temporal information with great success in several applications [45]. This is the case of the recently introduced general object tracker Re3 [46], which is robust against image occlusions and can be trained on long sequences thanks to its internal LSTM networks. Nonetheless, RNN have received little attention in the context of facial tracking yet, as RNN have received little attention in the context of facial tracking.

3. End to end denoised composite recurrent facial tracker

Our tracking model is a composite network that receives raw frames as input and returns the localization of facial landmarks as the final output. It is composed by four sub-networks, arranged in a way that permits end-to-end training for each of them without involving any hand-crafted features. Specifically, let \( \mathbf{X} \) and \( \mathbf{X}_{-1} \) denote the current and previous frame; our Denoised Composite Recurrent Tracker (DCRT) will estimate the position of \( n \) facial landmarks in the current frame \( \mathbf{X} \):

\[
\mathbf{I}_1 = \{ (\hat{x}_1, \hat{y}_1), \ldots, (\hat{x}_n, \hat{y}_n) \} = \text{DCRT}_\Phi(\mathbf{X}, \mathbf{X}_{-1}, \mathbf{b}_{t-1})
\]

\[
\mathbf{I}_t = \text{DCRT}_\Phi(\mathbf{X}, \mathbf{X}_{-1}, \mathbf{b}_{t-1})
\]

where \( \Phi \) are the parameters \( \{\theta^1, \theta^2, \ldots, \theta^5\} \) of our composite networks DCRT and \( \{\hat{x}_1, \ldots, \hat{x}_n, \hat{y}_1, \ldots, \hat{y}_n\} \) \( \in \mathbb{R} \times \mathbb{R} \).

Our DCRT consists of four individual sub-networks: Multi-Task Cascaded Neural Network face detector (MTCNN), facial bounding Box Tracker (BT), Facial Validator (FV) and Denoised Facial Alignment (DFA). Note that for face detection we relied on the state of the art MTCNN [38]. This task-specific arrangement allows us to optimize each network to each of their task characteristics, and simultaneously to inspect their behavior and contributions to our final landmark estimates when chained together using our tracking algorithm (cf. Section 3.4).

A schematic diagram of our tracker can be seen in Fig. 1. We start by assuming a tracking scenario, where we have an existing estimate for the bounding box of the preceding frame.\(^1\) This bounding box, together with the current and previous frames \( (\mathbf{X}, \mathbf{X}_{-1}) \) are fed to our BT network to produce a first estimate of the targeted landmarks \( \{\mathbf{b}^f_1\} \) and bounding box \( \{\mathbf{b}^b_1\} \), while at the same time its internal state is updated.

Once we have our first landmarks estimate \( \{\mathbf{b}^f_1\} \), we use the FV network to validate the result obtained by the tracker. To do so, we train the FV network to estimate the probability that the object tracked within \( \{\mathbf{b}^f_1\} \) is a face \( p(f) \). In case of obtaining a low probability, which would suggest that the BT network has lost track, we use the MTCNN to perform face detection on the current frame and re-initialize the whole network for the next time step.

In contrast, if \( \{\mathbf{b}^f_1\} \) is successfully validated by the FV network, the current frame and its bounding box \( \{\mathbf{b}^b_1\} \) are fed to the DFA network, which produces the final estimates for the target landmarks \( \{\mathbf{b}^f_t\} \) and the corresponding bounding box \( \{\mathbf{b}^b_t\} \). DFA works by first evaluating the image condition and, if necessary, applying denoising to the image; subsequently, landmark coordinates are estimated from a cleaned version of the input image. Note that, DFA will operate from an already detected and validated bounding box, which allows it to achieve a more accurate result. Further, the DFA network would be able to operate by itself, independently of the rest of the network (e.g. in the single image facial alignment task), but in such case it would lose the temporal modeling given by the previous sub-networks.

3.1. Bounding box tracker

We base our BT network on the structure of the Re3 tracker [46], which is a full end to end object tracker with internal LSTM networks to capture the temporal dependencies from video. Given input frames \( \{\mathbf{X}, \mathbf{X}_{-1}\} \) cropped as \( \{\mathbf{X}^{ib}, \mathbf{X}^{ob}\} \) with the previous Bounding Box \( \{\mathbf{p}_b = b_{t-1}\} \), the BT network estimates the landmark positions for the current frame \( \{\mathbf{b}^f_t\} \) and updates the internal state of the LSTM \( \mathbf{h}_t \) as follows:

\[
\mathbf{h}_t, \mathbf{b}^f_t = \text{BT}_\Phi(\mathbf{X}, \mathbf{X}_{-1}, \mathbf{p}_b, h_{t-1})
\]

\[
= \text{BT}_\Phi(\mathbf{X}^{ib}, \mathbf{X}^{ob}, h_{t-1})
\]

\[
= \text{LSTM}_\Phi \left( \text{EL}_\Phi \left( \mathbf{X}^{ib}, \mathbf{X}^{ob}, h_{t-1} \right) \right) \odot W^{b\phi}_{\Phi}
\]

where LSTM refers to the set of internal LSTM [44] networks, EL stands for the Embedding Layer, \( W^{b\phi}_{\Phi} \) is the set of weights of each fully connected layers of BT and EL respectively and res is the Inception-Residual Network [48] (Inception-Resnet). The Embedding Layer is a weighted concatenation of the residual network coefficients:

\[
\text{EL} = \begin{bmatrix}
\text{res}_{\Phi} \left( \mathbf{X}^{ib} \right), \text{res}_{\Phi} \left( \mathbf{X}^{ob} \right)
\end{bmatrix} \odot W^{EL}_{\Phi}
\]

We use \( \Phi^4 \) to denote the parameters of all sub-networks contained in BT. Finally, we also generate an estimate of the bounding box for the current frame \( \{\mathbf{b}^b_t\} \) directly from the estimated landmarks:

\(^1\) For initialization, this estimate can be obtained from the MTCNN detector or from an external input.
Note that, even though the architecture of $BT$ is based on $Re^3$, we introduce several key modifications to adapt this recurrent tracker model into this new problem domain:

1. First we preconditioned the convolutional network of our $BT$ to contain common facial features by replacing the internal Skip Convolution Networks (SkipNet) with the more sophisticated Inception-Resnet that has been pre-trained on the MS-Celeb [49] and CasiaWebFace [50] datasets with triplet loss [51]. Fig. 2 visualizes the differences between the original SkipNet on $Re^3$ versus the more complex structure of $BT$, which is inherited from the Inception-Resnet (Version 1). Each block of Inception-Resnet architecture can be expressed as below:

$$r_{i+1} = H(r_i) + F(r_i, W_i)$$

where $r_i$ and $r_{i+1}$ are the input and output of the i-th block, $H(b_i)$ is the identity matrix and $F$ represents the combined effect of the various convolutional and ReLU layers. Notice that SkipNet does not have the advantage of residual connection as in the Inception-Resnet which eases the gradient flow in optimization [48].

2. Second we use the $BT$ network to produce a first estimate of landmark locations ($l_{BT}^t$) following the work of [6], but we split the fully-connected layer that receives the output from the LSTMs into five independent fully-connected networks so that each of them is focused on a specific facial region. To this end, we divide the facial landmarks in the following regions: facial silhouette (our outer contour), eyebrows, eyes, nose and lips. Thus $W_{BT} = \{W_{B1}, W_{B2}, W_{B3}, W_{B4}, W_{B5}\}$.

3. Finally, we reduce the dimensionality of the input image size to $128 \times 128$ to accelerate the training process. This in turn also reduces the number of the neurons of the original $Re^3$ by half and helps to avoid over-fitting [52] while still achieving state of the art accuracy.

2 The trained inception resnet is publicly available on: https://github.com/davidsandberg/facenet.
3.3. Denoised facial alignment

We argue that when given a noisy input image, final landmark estimation can be improved by first minimizing the existing noise and only then passing the image to any facial landmark estimator. To achieve this, we adopt a joint approach involving two major sub-networks to build our Denoised Facial Alignment network (DFA): Internal Image Denoiser (IID) and Facial Landmark Estimator (FLE) networks. We use the current cropped facial area $X^{ib}$ as input image, which we assume to be potentially contaminated with unknown noise. In other words, $X^{ib}$ is assumed to be an estimate of a noiseless image $Y$ to be estimated by the IID network. We do so in a two-step approach in which we first detect if the noise in $X^{ib}$ complies with any of our noise models and, if so, clean the input image to generate an enhanced estimate of $Y$, namely $Y = IID(X^{ib})$. Otherwise, if $X^{ib}$ does not match any of the internal noise models, then $Y = X^{ib}$. The resulting image $Y$ will be fed to the FLE network to estimate $n$ landmark coordinates $I^t$:

$$I^t = DAE_{\phi^t, \psi^t}(X^{ib}) = FLE_{\phi^t, \psi^t}(IID_{\phi^t, \psi^t}(X^{ib}))$$

### 3.3.1. Internal Image Denoiser

We adopt a selective denoising approach \[19, 53\] by combining Multi-Task noise Classifier (MTC) network and multiple Specialized Denoiser Auto Encoders (SDAE) sub-networks to work in tandem. This joint model selectively performs denoising based on the detected condition of the input image to trigger separate specialized denoiser sub-models. This has been shown to perform better than directly denoising all input images, as the latter may actually distort an already clean input image \[19\]. The formulation of the IID block is as follows:

$$\tilde{Y} = IID_{\phi^t, \psi^t}(X^{ib}) = SDAE_{\phi^t}(MTC_{\phi^t}(X^{ib}), X^{ib})$$

Given $C$ known image degradation types, the MTC will estimate the probability that degradation type $c$ is present in the input image $X^{ib}$:

$$s_c = \frac{\exp(W_{\phi^t} \circ CNN_{\phi^t}(X^{ib}))}{\sum_{c=1}^{C} \exp(W_{\phi^t} \circ CNN_{\phi^t}(X^{ib}))}$$

$$\tilde{c} = MTC_{\phi^t}(X^{ib}) = \text{argmax}_c \{s_c\} \quad \{s_c = \{s_0, s_1, \ldots, s_C\}\}$$

where $W_{\phi^t}$ are the multinomial bottleneck regression layers parameterized by $\phi^t$, $CNN$ is the set of convolutional layers for MTC, and $s_c$ is the estimated probability that the current input $X^{ib}$ is contaminated with specific noise-class $c$, with $s_0$ denoting the noiseless case. If the classifier detects noise in the input image, then $\tilde{c} > 0$ and the image is denoised by one of the specialized denoisers \{DAE1, DAE2, \ldots DAEc\}. We build our Internal Image Denoiser based on the Hourglass shaped Auto-Encoder Architecture with skip connection \[54\]. The structure of our DAE is similar to the work of \[55\] with a few additional mirror layers on both encoder and decoder parts. Each of these blocks can be trained to capture a different type of noise:

$$SDAE_{\phi^t}(\tilde{c}, X^{ib}) = DAE_{\phi^t, \psi^t}(X^{ib}) = \text{enc}_{\epsilon_{\phi^t}}(\text{dec}_{\phi^t, \psi^t}(X^{ib}))$$

where $\phi^t$ contains the parameters learned for all DAE blocks. In case $\tilde{c} = 0$, the denoising process is skipped to avoid unnecessarily distorting the input image, i.e. $SDAE_{\phi}(0, X^{ib}) = X^{ib}$.

### 3.3.2. Facial landmark estimator

We build our final pipeline of FLE using the state-of-the-art Squeeze and Excitation Network (SENET) \[56\] which has been pre-trained on the recently published VGGFace2 \[57\] facial dataset.\footnote{https://github.com/ox-vgg/vgg_face2.} This landmark localization procedure can be expressed mathematically as below:

$$FLE_{\phi^t}(\tilde{Y}) = W_{\phi^t}^{FLE} \circ SEN_{\phi^t}(\tilde{Y})$$

### 3.4. Recurrent denoised facial tracking algorithm

The operation of our Denoised Composite Recurrent Tracker, DCRT, which combines all the blocks explained previously in this section, is shown in Algorithm 1. When a suitable detection of the facial region is available, e.g. from initialization or the previous frame (lines 8 and 10), the BT network produces a first estimate of facial landmarks (line 13) and bounding box (line 14). Then, the FV network is used to estimate the probability $p(f)$ that the output from BT corresponds to a face. If $p(f)$ is sufficiently high (above threshold $T_f$), the initial estimate is refined by the DFA network to produce the final tracker estimate (lines 18 and 19) with its selective internal denoisers. Otherwise, it is assumed that the BT has lost track and there is a need to re-initialize the tracker (line 16).

We perform re-initialization between lines 3 and 6. We start by detecting the face in the current frame by means of the MTCNN network. This detector is likely to produce multiple detections, hence its outputs are validated with respect to the bounding box of the previous frame $b_{t-1}$. Specifically, we compare the Euclidean distance between each new detection and the center of the previous bounding box $d(b_{t-1}, b^{MT})$ with respect to the magnitude of the previous bounding box, and keep the one that produces the minimum ratio:

$$p_b = \begin{cases} b, & \min_{v \in b_{t-1}} \frac{d(b_{t-1}, b)}{\|b_{t-1}\|} < T_b \\ b_{t-1}, & \text{otherwise} \end{cases}$$

as long as there is at least one detection whose ratio is below threshold $T_b$. Otherwise, all new detections are too far from the previous tracking result and no re-initialization is performed. The latter is necessary to tackle the cases in which the face being tracked moves out of the visual field. In such cases, without threshold $T_b$ the system might be incorrectly re-initialized to track another face. In contrast, by using $T_b$ the tracker remains in its latest valid coordinates awaiting for the tracked object to come back to the field of view.

Finally, (SeqBT) controls the length of the temporal window that is considered by the tracker (in frame units), which is fixed at training time (see next section). If the tracker is re-initialized or if the sequence length (SeqT) exceeds the temporal window (SeqBT), then the internal state of the BT network is reset to 0 (line 12). This allows the network to refresh the facial appearances encoding to adapt to the surrounding characteristic of the current view.
3.5. Overall loss and model training

To train the $BT$ sub-network, we performed curriculum learning with sequence lengths between $SeqBT = 2$ to $SeqBT = 32$ frames [46]. Multiple stages of transfer learning [58] were used to condition the pre-trained Inception-Resnet. To do so, we fine-tuned this network to perform single image landmark estimation by adding auxiliary Fully Connected layer and training the resulting network with $\ell_2$ loss using the 300-W [3] and Menpo [41] datasets. After convergence, we integrated the internal convolutional layers into our $BT$ network, which was then trained end-to-end for landmark tracking using the 300-VW training dataset with $\ell_1$ loss. Furthermore, we also used the same facial landmark alignment datasets to train $FV$ with standar cross entropy loss.

We consider two types of image degradations to train the $IID$ sub-network: Image Blur and Illumination Differences. This is motivated by our previous findings about the common weaknesses of current facial alignment models against such degradation models [19]. Specifically, each degradation model was obtained as follows:

1. **Blur Model.** We added Gaussian Blur by convolving the input image with two dimensional Gaussian filters with $\sigma \in \{1,3,5\}$ [9,19]. In addition, we also included motion blur [59] by averaging $L$ frames, with $L \in \{10,15,25\}$ to further simulate real world blur distortion.

2. **Illumination Model.** On this degradation models, we incorporated both synthetic and real life illumination degradation. First we synthetically created images with low pixel intensities by scaling the original pixel values linearly by a factor $s \in \{0.8,0.5,0.2\}$ from the original intensities. Second, we used the Yale [60,61] and SOF datasets [62] which contain facial images captured under large illumination variations. Specifically, Yale dataset provides some facial examples taken under different poses, while SOF dataset offers the colored facial images with different emotional expressions, both are under different illumination settings. We used two versions of Yale dataset: original Yale dataset [61] that consists of 15 subjects taken under different lighting (left, right and center lighting); the extended Yale Dataset B [60] consists of 28 subjects with different light source directions and poses. For the original version, we obtain the ground-truth (considered well-lit images) by using the centered-lit faces. Whereas for the extended Yale, we use the face taken with neutral position lighting, i.e. indicated by azimuth and elevation degree with the value of 0. As for the SOF dataset, we use its second version that includes videos of 12 subjects with four different emotions (normal, happy, sad, angry, disgusted, and surprises), and filmed under extreme lighting conditions (using arbitrarily located...
wheel-whirled lamp). We used the faces taken on the ideal lighting, indicated through the provided metadata of ‘illuminationQuality’ label of valued 0 for each associated emotion expression as the ground truth for this dataset. Finally, we crop-centered the faces from SOF dataset using the provided 17 facial landmarks, while our previous tracker [23] and facial alignment [19] were used to obtain the landmarks for the Yale data given that they are not available for this dataset.

Examples of some training images together with their degraded versions are depicted in Fig. 3.

These datasets, alongside with the previous facial alignment dataset were used to perform joint training of sub-modules of DFA, i.e. FLE and IID, allowing them to be optimized in parallel. We argue that synchronous optimization of IID and FLE allows IID to produce cleaner intermediate images Y since that favors their mutual aim to achieve more accurate landmark estimates. Hence the complete loss for DFA training is as follows:

\[ L_{DFA} = \min \lambda_1 l^2(FLE(Y), I) + \lambda_2 L_{IID} \]  

\[ L_{IID} = \min \lambda_3 (H(\hat{c}, MTC(X)) + \lambda_4 l^2(X, \hat{X}) + L_{SDA} \]  

\[ L_{SDA} = \min \lambda_5 l^2(Y, D(X)) \]

where \( X \) is the input image, \( Y \) is the input image after denoising (i.e. an estimate of the unknown clean image \( Y \)), 1 are the ground-truth landmarks, \( \hat{c} \) is the estimated noise class, \( H() \) is the cross-entropy and \( \hat{X} \) is the autoencoder reconstruction of \( X \). The \( \lambda \) coefficients act as regularization parameters for each term [5,63]. In the initial training phase, we set a higher value for \( \lambda_2 \) compared to \( \lambda_1 \), for instance \( \lambda_2 \) of 0.75 and \( \lambda_1 \) of 0.25, to accelerate the training of the denoising part (\( \lambda_3 \) and \( \lambda_4 \) are set with equal value of 0.5), since we found that this part took more time to converge than the former. As training progresses, we gradually increase the value of \( \lambda_1 \) by a step value of 0.1, and reduce the value of \( \lambda_2 \) with a similar value to better balance the training priority, reaching a final value of 0.5 for both coefficients.

Fig. 4 illustrates the impact of our proposed joint training when the input images present some sort of noise or distortion. We can see that using this joint training scheme, DFA is able to improve the accuracy of the estimated landmarks (5th column), compared to the results using separate loss (6th column). We can also see that the internally denoised images are also less distorted (4th and 5th column), suggesting that these cleaner image characteristics are beneficial for FLE to produce more accurate landmark estimates [59]. Finally, as we expected, the use of FLE (7th column) by itself, i.e. without IID, yields the worst performance given its lack of any internal denoising modeling (more details can also be seen on the next Section 4).

Finally, in all the training process, we performed data augmentations by means of horizontal flipping, -45° to 45° degree rotations and artificial strip boxes across the images to simulate occlusions. We trained our model using both Stochastic Gradient Descent (SGD) and ADAM optimizer [64] with scheduled weight learning decay every 10,000 iterations. We first initialized training with SGD to speed up convergence, and then used ADAM for fine-tuning at final training stages. Five NVIDIA Titan GPUs were used for training which took approximately one to two days to train a single BT for a defined sequence length, and around two days for both DFA and FV.

![Fig. 4. Visual examples of different results of DFA training with respect to the use of our proposed joint training of IID and FLE modules. The first two columns shows the degraded input images and their respective cleaned (ground truth) versions. Columns 3 and 4 consist of the denoised output of IID with and without joint training of FLE respectively. Columns 5 and 6 provide the predicted landmark estimates of FLE + IID, with and without joint training. Column 7 shows the FLE results without the use of IID.](image-url)
4. Experiments

We present our results on two major experiments: Single Image Face Alignment and Deformable Facial Tracking. For fair comparisons, we use the widespread 68 facial landmark points from [2–4] and compare our model against other state of the art methods by either gathering their reported results or reproducing them from their publicly available code.

4.1. Facial alignment experiments

In this section, we compare our internal Denoised Facial Alignment (DFA) presented in Section 3.3 for single image facial alignment. In addition, we also report results from our FLE component operating with-internal denoising in all the considered metrics. Secondly, DFA produces state of the art results for both datasets, consistently achieving the highest AUC and the lowest FR and NMSE among all compared methods. Finally, we also observe a noticeable improvement against our previous FaDeNN [19] model, which highlights the effectiveness of the newly proposed approach.

We also observe that generally, models based on deep learning perform particularly well, with SAN [11] and ECT [7] achieving the second and third best performance in terms of AUC, followed by FAN [5], which seems to produce more stable estimates judging from its low FR values. Overall, these models perform better than those using handcrafted features, such as ERT [32] and CFSS [33], which reflects the maturity of deep learning approaches and their advantage compared to the traditional approaches.

4.1.2. Results on 300-W and Menpo datasets

Table 1 shows the results on both 300-W and Menpo test datasets, while the corresponding AUC graph is shown on Fig. 5. Based on these results we can see the improvement obtained by incorporating the internal denoiser (IID) in our facial alignment model. Firstly, DFA obtains higher performance than FLE (namely, the same model without denoising) in all the considered metrics. Secondly, DFA produces state of the art results for both datasets, consistently achieving the highest AUC and the lowest FR and NMSE among all compared methods. Finally, we also observe a noticeable improvement against our previous FaDeNN [19] model, which highlights the effectiveness of the newly proposed approach.

We also observe that generally, models based on deep learning perform particularly well, with SAN [11] and ECT [7] achieving the second and third best performance in terms of AUC, followed by FAN [5], which seems to produce more stable estimates judging from its low FR values. Overall, these models perform better than those using handcrafted features, such as ERT [32] and CFSS [33], which reflects the maturity of deep learning approaches and their advantage compared to the traditional approaches.

4.1.3. Visual results on 300-W and Menpo

Fig. 6 visualizes examples of estimated facial landmarks for our model without (FLE) and with (DFA) denoising, against the other three best performing approaches in Table 1: SAN [11], ECT [7], and FAN [5]. For each dataset, the different columns show examples of input images under different imaging conditions. The first column of each dataset shows what could be considered an optimal (clean) image (columns 1 and 6). The second and third columns show blurring artifacts, and the fourth and fifth columns show sub-optimal illumination.

By visually investigating these examples, we first notice that in case of relatively cleaned image input, our models produce more accurate predictions than the compared alternatives. This might be due to the inclusion of the denoising block during training, which removes part of the artifacts present in the training data and thus enhances the quality of the training set. On the other hand, in case of degraded images, i.e. with...
noisy input is introduced, we can see that the internal denoiser enhances the image, thus improving the quality of the final landmark estimates as shown on both of blurry and low illumination examples. This finding of the correlation between cleaned input image and accurate landmark estimates, i.e. cleaner the images input will yield more accurate landmark predictions, also conforms with a recent related study [59] suggesting the benefit of our denoising approach. Finally, we also notice that the internally enhanced images are visually better than the original ones, which further shows the effectiveness of each of our internal denoiser modules for removing their specific noise class.

4.2. Facial tracking experiments

In this section, we compare our full DCRT model described in Section 3 for deformable facial tracking in the wild. We empirically set the threshold values of $T_D = 1.0$ and $T_C = 0.5$ for our model to perform tracking. As will be shown later, our model produces the best results when the sequence length is fixed to 2 frames and the internal denoiser module is activated (see Section 4.2.4).

4.2.1. Experiment setup

We compared our model against other 13 available state of the art facial tracking models, including our previous results using the CRCT model [23] and FADeNN [19]. We also provide results form the SAN model [11], which obtained the second-best performance in the facial

<table>
<thead>
<tr>
<th>Methods</th>
<th>Category 1 AUC</th>
<th>Category 1 FR</th>
<th>Category 2 AUC</th>
<th>Category 2 FR</th>
<th>Category 3 AUC</th>
<th>Category 3 FR</th>
</tr>
</thead>
<tbody>
<tr>
<td>Yang [17]</td>
<td>0.791</td>
<td>2.4</td>
<td>0.788</td>
<td>0.32</td>
<td>0.710</td>
<td>4.46</td>
</tr>
<tr>
<td>MD_CSS [14]</td>
<td>0.784</td>
<td>1.8</td>
<td>0.783</td>
<td>0.34</td>
<td>0.713</td>
<td>7.47</td>
</tr>
<tr>
<td>ME_CSS [14]</td>
<td>0.758</td>
<td>3.56</td>
<td>0.772</td>
<td>0.38</td>
<td>0.659</td>
<td>11.3</td>
</tr>
<tr>
<td>Gu [6]</td>
<td>0.718</td>
<td>1.2</td>
<td>0.703</td>
<td>0.2</td>
<td>0.617</td>
<td>4.83</td>
</tr>
<tr>
<td>Uricar [37]</td>
<td>0.657</td>
<td>7.62</td>
<td>0.677</td>
<td>4.13</td>
<td>0.574</td>
<td>7.96</td>
</tr>
<tr>
<td>Xiao [20]</td>
<td>0.750</td>
<td>5.9</td>
<td>0.762</td>
<td>3.84</td>
<td>0.695</td>
<td>7.38</td>
</tr>
<tr>
<td>Raja [21]</td>
<td>0.735</td>
<td>6.56</td>
<td>0.717</td>
<td>3.91</td>
<td>0.659</td>
<td>8.29</td>
</tr>
<tr>
<td>Wu [39]</td>
<td>0.674</td>
<td>13.9</td>
<td>0.732</td>
<td>5.6</td>
<td>0.602</td>
<td>13.1</td>
</tr>
<tr>
<td>OpenFace [35]</td>
<td>0.668</td>
<td>12.58</td>
<td>0.715</td>
<td>7.78</td>
<td>0.566</td>
<td>14.62</td>
</tr>
<tr>
<td>SAN_DT [11]</td>
<td>0.690</td>
<td>3.0</td>
<td>0.705</td>
<td>0.76</td>
<td>0.611</td>
<td>6.57</td>
</tr>
<tr>
<td>SAN_PV [11]</td>
<td>0.713</td>
<td>9.61</td>
<td>0.756</td>
<td>0.83</td>
<td>0.571</td>
<td>19.06</td>
</tr>
<tr>
<td>CRCT [23]</td>
<td>0.784</td>
<td>0.5</td>
<td>0.790</td>
<td>0.05</td>
<td>0.729</td>
<td>1.75</td>
</tr>
<tr>
<td>FADeNN [19]</td>
<td>0.768</td>
<td>2.73</td>
<td>0.767</td>
<td>0.26</td>
<td>0.729</td>
<td>1.7</td>
</tr>
<tr>
<td>DCRT (Ours)</td>
<td><strong>0.822</strong></td>
<td><strong>0.4</strong></td>
<td><strong>0.813</strong></td>
<td><strong>0.03</strong></td>
<td><strong>0.752</strong></td>
<td><strong>2.32</strong></td>
</tr>
</tbody>
</table>
We use the original 2D facial landmarks directly as ground-truth and convert them to minimum-effort trackers (i.e. using none or very little temporal information). The list of compared trackers is as follows:

1. Yang: The original winner of 300-VW deformable facial tracking challenge from Yang et al. [17].
2. MEEM_CFFS and MD_CFFS: Two hybrid trackers that showed the best performance in the recent facial tracking review from Chrysos et al. [14].
4. Uricar, Xiao, Raja and Wu: Four other trackers from the original 300VW competition [2] consisting of models from Uricar et al. [37], Xiao et al. [20], Raja et al. [21], and Wu et al. [39].
5. OpenFace: Our previous tracking model [23], which in contrast to the current proposed model, does not include internal denoising.
6. SAN_DT and SAN_PREV: The SAN facial image alignment model [11] converted into a tracker by using the MTCNN face detector (SAN_DT) or the previous frame result (SAN_PREV) as initialization.
7. CRCT: Our previous tracking model [23], which in contrast to the currently proposed model, does not include internal denoising.
8. FADeNN: Our previous facial alignment model [19] which uses internal denoising, converted into a tracker by using the result from the previous frame as initialization.

We performed our experiments using the 300-VW dataset [2], which is the largest available deformable facial landmark tracking dataset. It comprises 55 videos divided into three categories according to the difficulty levels:

1. The first category contains people recorded in well-lit conditions, which is intended to evaluate facial tracking with images acquired in nearly ideal conditions.
2. The second category consists of people recorded in unconstrained conditions (variable illumination, dark rooms, etc.) with arbitrary poses. This setting evaluates facial tracking models in real-world human-computer applications.
3. The third category contains videos of people recorded in fully unconstrained conditions which include cases of ambient illumination differences, large occlusions, expressions, etc. This scenario aims to assess the models under arbitrary recording conditions.

We use the original 2D facial landmarks directly as ground-truth and first-frame initialization for all models. We follow [2,14] to evaluate each tracked landmark with Area Under the Curve (AUC) and Failure Rate (FR) for Normalized Mean Error (NME), as previously explained in Section 4.1.1.

4.2.2. Results on 300-VW test dataset

Table 2 shows the results of all compared models for each category, while the respective AUC curves for several trackers are shown in Fig. 7. We can observe that our model achieves top state of the art performance, consistently outperforming all compared alternatives. We also notice a large improvement with respect to our previous models, namely CRCT [23] and FADeNN [19], which were partial implementations of the short/long-range tracking and denoising, respectively; this highlights the need for a unified approach as the one presented here.

Regarding the other compared methods, we found the results of the OpenFace library (one of the most popular tools for nowadays) among the less accurate ones, which contrasts with the relatively good results obtained by this library in the single image alignment experiments. This occurs because, as stated previously, OpenFace is not actually designed for tracking and is indicative of the accuracy loss that can be experienced by not taking into account temporal dependencies. This is also supported by the results obtained by SAN_PV, which showed comparable or better performance than several other trackers and was clearly superior than its detection counterpart (SAN_DT).

4.2.3. Visual results on 300-VW

Fig. 8 showcases results of our DCRT model against the state of the art Yang et al. [65] and the widely used OpenFace [35]. We can point out several observations based on these visual examples. Firstly, our model performs better on the majority of clean inputs, as shown in the first two rows, consistently producing accurate estimations even in cases of extreme and profile poses.

Secondly, we see that our model is also robust against severe image degradation, e.g. blur, occlusion and illumination. This is especially clear when processing the extremely dark frames from the last two rows, where the faces sometimes are hardly visible to a human observer. In these examples, our model still manages to produce consistent landmark estimates thanks to the denoising block, while other models struggle. When visually inspect such difficult examples, we find that the intermediate images generated by the internal denoiser are considerably changed with respect to the original ones. This is especially noticeable in cases of poor illumination (e.g. last 2 rows of Fig. 8), where the denoised images not always look visually “cleaner” from a perception point of view, but we can still see that the illumination has been considerably enhanced, especially in the facial area.

Thirdly in some cases, we also notice that our tracker produced facial landmarks that are visually more convincing than the actual ground-truth locations, as can be seen on frames 346, 356, 358 and 706 of the
Fig. 8. Visual examples of tracked facial landmarks on the 300-VW dataset. Odd rows show the results from Yang [17] and OpenFace [35] overlaid on the original image against the ground truth. Even rows show our results, displayed on the internally denoised images. From top to bottom, we show examples of clean input images (rows 1 and 2), blurred inputs (rows 3 to 6) and low ambient illumination (rows 7 to 10).
first row. These occurrences, though small, occur due to the need to use semi-automatic annotation methods to generate ground truths for very large datasets [5].

4.2.4. Ablations study

In this section, we perform a detailed analysis of the behavior of our tracker under different operation conditions. Firstly, we analyze the impact of the temporal context in our full tracker, DCRT, by varying length of the training sequences i.e. SeqBT = 2, 4, 8, 16 which we refer to as DCRT-2, DCRT-4, ..., DCRT-32. Next, we also report the results obtained by a partial implementation of our tracker that does not include denoising, namely CRT-2, CRT-2, ..., CRT-32. Finally, we also report results under minimum-effort tracking:

1. Tracking by detection without denoising (FLE-Det).
2. Tracking by detection with denoising (DFA-Det).
3. Initialization from the previous frame, without denoising (FLE-Prev).
4. Initialization from the previous frame, with denoising (DFA-Prev).

Table 3 shows the results of all the above variants of our tracker separately for each category of the test dataset, and jointly considering the images in all categories to summarize the overall performance of each model. The respective AUC curve are displayed in Fig. 9. We also include the result of Yang et al. [17] as the baseline, as well as our previous results for both CRCT [23] and FaDeNN [19].

We can summarize our findings based on these results as follows:

• The minimum-effort tracking, when using initialization from the previous frame, yields quite comparable accuracy to the one provided by the baseline [17]. However, these results are still inferior against our models with internal temporal modeling (DCRT), in all cases with quite large margin. Further, all of our full models (DCRT-x, with internal temporal modeling), achieve better results than all tracking by detection approaches.

• The difference between results of models trained on different sequence lengths is minimal. In the majority of cases, shorter lengths were slightly better than longer ones. Nevertheless, these results must be read in relation to the test sequences, which show quite irregular (not necessarily natural) facial movements [23].

• We observe a general improvement when introducing the Internal

<table>
<thead>
<tr>
<th>Models</th>
<th>Category 1</th>
<th>Category 2</th>
<th>Category 3</th>
<th>All Category</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>AUC</td>
<td>FR</td>
<td>AUC</td>
<td>FR</td>
</tr>
<tr>
<td>YANG [17]</td>
<td>0.7910</td>
<td>2.4</td>
<td>0.7880</td>
<td>0.32</td>
</tr>
<tr>
<td>CRCT [23]</td>
<td>0.7835</td>
<td>0.5</td>
<td>0.7901</td>
<td>0.05</td>
</tr>
<tr>
<td>FaDeNN [19]</td>
<td>0.7682</td>
<td>2.73</td>
<td>0.7869</td>
<td>0.26</td>
</tr>
<tr>
<td>FLE-Det</td>
<td>0.7332</td>
<td>4.92</td>
<td>0.7367</td>
<td>1.9</td>
</tr>
<tr>
<td>FLE-Prev</td>
<td>0.7872</td>
<td>3.45</td>
<td>0.7811</td>
<td>0.33</td>
</tr>
<tr>
<td>CRT-2</td>
<td>0.8103</td>
<td>0.56</td>
<td>0.7991</td>
<td>0.03</td>
</tr>
<tr>
<td>CRT-4</td>
<td>0.8091</td>
<td>0.79</td>
<td>0.8023</td>
<td>0.03</td>
</tr>
<tr>
<td>CRT-8</td>
<td>0.7983</td>
<td>2.39</td>
<td>0.8031</td>
<td>0.03</td>
</tr>
<tr>
<td>CRT-16</td>
<td>0.7956</td>
<td>2.05</td>
<td>0.8022</td>
<td>0.04</td>
</tr>
<tr>
<td>CRT-32</td>
<td>0.8052</td>
<td>0.80</td>
<td>0.8022</td>
<td>0.05</td>
</tr>
<tr>
<td>DFA-Det</td>
<td>0.7483</td>
<td>3.01</td>
<td>0.7459</td>
<td>1.66</td>
</tr>
<tr>
<td>DFA-Prev</td>
<td>0.7965</td>
<td>3.04</td>
<td>0.7971</td>
<td>0.09</td>
</tr>
<tr>
<td>DCRT-2</td>
<td>0.8219</td>
<td>0.40</td>
<td>0.8130</td>
<td>0.03</td>
</tr>
<tr>
<td>DCRT-4</td>
<td>0.8199</td>
<td>0.65</td>
<td>0.8145</td>
<td>0.04</td>
</tr>
<tr>
<td>DCRT-8</td>
<td>0.8204</td>
<td>0.54</td>
<td>0.8145</td>
<td>0.03</td>
</tr>
<tr>
<td>DCRT-16</td>
<td>0.8087</td>
<td>1.55</td>
<td>0.8129</td>
<td>0.05</td>
</tr>
<tr>
<td>DCRT-32</td>
<td>0.8173</td>
<td>0.60</td>
<td>0.8140</td>
<td>0.04</td>
</tr>
</tbody>
</table>

Fig. 9. AUC graph of our models on 300-VW using different settings: a) results on images from the first category, b) second category, c) third category, and d) overall results including the images in all categories.

Declaration of Competing Interest
The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References


