Dual-Agent Deep Reinforcement Learning for Deformable Face Tracking

Minghao Guo, Jiwen Lu(✉), and Jie Zhou

Tsinghua University, Beijing, China
guomh2014@gmail.com,{lujiwen,jzhou}@tsinghua.edu.cn

Abstract. In this paper, we propose a dual-agent deep reinforcement learning (DADRL) method for deformable face tracking, which generates bounding boxes and detects facial landmarks *interactively* from face videos. Most existing deformable face tracking methods learn models for these two tasks individually, and perform these two procedures subsequently during the testing phase, which ignore the intrinsic connections of these two tasks. Motivated by the fact that the performance of facial landmark detection depends heavily on the accuracy of the generated bounding boxes, we exploit the interactions of these two tasks in probabilistic manner by following a Bayesian model and propose a unified framework for simultaneous bounding box tracking and landmark detection. By formulating it as a Markov decision process, we define two agents to exploit the relationships and pass messages via an adaptive sequence of actions under a deep reinforcement learning framework to iteratively adjust the positions of the bounding boxes and facial landmarks. Our proposed DADRL achieves performance improvements over the state-of-the-art deformable face tracking methods on the most challenging category of the 300-VW dataset.

Keywords: Deformable face tracking · Reinforcement learning
Deep learning

1 Introduction

Deformable face tracking has received considerable attention in computer vision recently with numerous applications such as human computer interaction, facial expression analysis, and person identification. The aim of deformable face tracking is to detect the key points around facial components and facial contours across all frames of a given face video. It is a challenging problem in practice because face samples are usually captured in unconstrained conditions, where large poses, heavy occlusions, illumination variations and motion artifacts usually occur.

Over the past decade, many efforts [1–3] have been devoted to this problem, which usually employ a "tracking-by-detection" strategy to perform deformable face tracking in a *serial* manner. Specifically, these methods first generate a

V. Ferrari et al. (Eds.): ECCV 2018, LNCS 11214, pp. 783–799, 2018.
https://doi.org/10.1007/978-3-030-01249-6_47

high-scored bounding box covering a face region, and then apply face alignment to localize facial landmarks based on the bounding box. Hence, face alignment depends heavily on the generated bounding box. Figure 1(a) shows an example to illustrate the effect of face box generation for facial landmark detection. We see that the bias from the ground-truth bounding box affects the alignment accuracy heavily because the bounding box is generated without considering the face conditions of pose and expression. Especially when face undergoes extreme conditions, the facial region selected by the bounding box usually misses facial landmarks, resulting in limited performance of face alignment. A desirable deformable face tracking approach is to exploit the rich interaction between face bounding box generation and face alignment. Since facial landmarks can effectively represent face pose across frames, they can provide auxiliary information for accurate bounding box generation. However, most existing deformable face tracking methods ignore such interaction, which results in low accuracy fitting for extreme conditions.

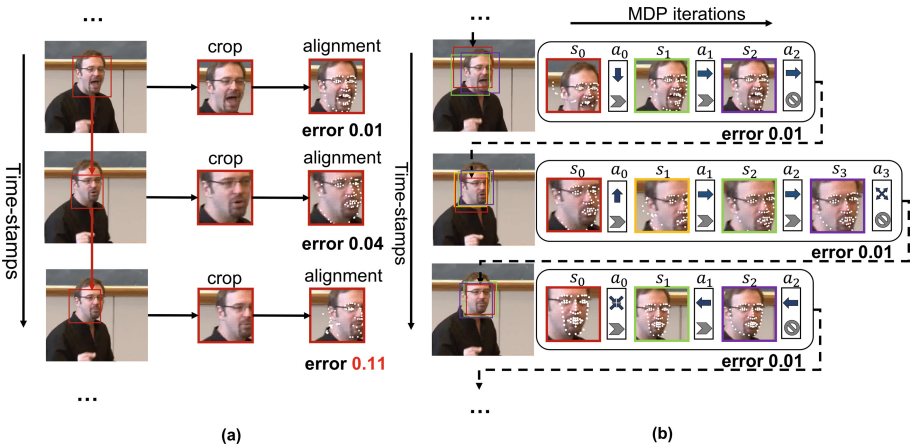


Fig. 1. (a) Existing "tracking-by-detection" methods [1–3] produce deformable face tracking in a *serial* manner. (b) Our DADRL method formulates deformable face tracking as a Markov decision process (MDP) problem, and produces bounding box tracking and landmark detection in an *interactive* manner. Here s_i denotes the MDP state, a_i denotes the MDP action. The dash line represents that initial bounding box of the current frame is the tracked box of previous frame. The blue color and the gray color denote the tracking agent action and the alignment agent action respectively (best viewed in color). (Color figure online)

In this work, we propose a dual-agent deep reinforcement learning (DADRL) method for deformable face tracking, which performs bounding box generation and facial landmark detection in *interactive* manner. Specifically, we exploit the interaction of these two procedures in probabilistic manner by following a Bayesian model. Unlike existing deformable face tracking methods which directly infer the decomposed form of joint probability for bounding boxes and facial landmarks, we train these two models to learn two conditional distributions

simultaneously. Then, the connections between these two tasks are formulated as two marginal distributions, and their correlation is explicitly modeled with learnable parameters. Motivated by the observation that the face tracking complexity varies across frames, our method utilizes reinforcement learning as a principled way to learn how to make adaptive decisions during deformable face tracking. We formulate this sequential procedure as a Markov decision process, which models bounding box generation and face alignment as two agents. These dual agents predict a variable-length sequence of actions to position updates of bounding boxes and landmarks. Experiment results show that our proposed DADRL achieves large performance improvements over the state-of-the-art deformable face tracking methods on the 300-VW dataset [4].

2 Related Work

Deformable Face Tracking: Deformable face tracking focuses on tracking a set of facial landmarks across all frames of a given face video. Existing deformable face tracking methods can be mainly classified into two categories: pure shape tracking methods and tracking-by-detection methods. Methods of first category [5–8] perform face detection in the first frame of each face video and then conduct facial landmark localization at each consecutive frame by using the alignment result of the previous frame as the initialization. Based on this fundamental process, recent works focus on exploiting the temporal dependency relationship of landmarks across different frames. For example, the recurrent encoder-decoder network [7] consists of a sequence of spatial and temporal recurrences. The two-stream transformer networks [8] captures both spatial and temporal information by using a couple of networks. These methods partially handle the large variations of pose and expression across the whole video, because the motion between two adjacent frames is usually small. However, these methods struggle with the drifting drawback, as the error accumulates through time across the whole video. Methods in the second category [1,3,9–12] apply face detection/tracking and facial landmark localization successively at each frame, which are also similar to most existing image-based face alignment methods [7,13–18]. While these methods eliminate drifting to some extent, these two models are trained individually and utilized in a serial manner. As a result, the performance of face alignment is restricted, which may cause low accuracy fittings under a poor generated bounding box. To address this, Khan *et al.* [19] proposed a synergistic approach to perform landmark localization by using different detection and tracking initializations, which partially utilizes the correlation between the bounding box generation and the face alignment. However, they only employed a separate tracking model to generate bounding boxes, which is not optimized together with the alignment model during training.

Deep Reinforcement Learning: Reinforcement learning has been originated from humans' decision making process [20], which aims to enable the agent to make decisions from its experiences. Deep reinforcement learning, which is

a combination of deep learning and reinforcement learning, can be divided into two classes: deep Q learning [21–23] and policy gradient [24,25]. The goal of deep Q Networks is to learn a state-action value function given by a deep network. Policy gradient methods learn the policy which maximizes the expected future reward using gradient descent. Recently deep reinforcement learning has gained great successes in several computer vision applications. For example, Rao *et al.* [26] proposed an attention-aware deep reinforcement learning method for key-frame selection in video face recognition. Yu *et al.* [27] proposed a sequence generative adversarial networks via policy gradient. Yoo *et al.* [28] proposed a sequential visual tracker learned by policy gradient. Foerster *et al.* [29] and Sukhbaatar *et al.* [30] proposed multi-agent deep reinforcement learning methods to communicate message between different agents. Kong *et al.* [31] proposed a collaborative algorithm to localize multiple objects via multi-agent reinforcement learning. Unlike these methods which have a common network architecture, we propose a dual-agent deep reinforcement learning (DADRL) method which is equipped with a dual-agent process: face bounding box generation and facial landmark detection.

3 Approach

In this section, we first present the Bayesian formulation of deformable face tracking to introduce the dual learning scheme. Then we propose the settings of Markov decision process (MDP) to show how to utilize deep reinforcement learning. Lastly, we detail the architecture of the proposed DADRL and the training procedure.

3.1 Problem Formulation

Suppose we have a face video consisting of K frames, $\{I_k\}_{k=1:K}$. For the k-th frame $I_k \in \mathbb{R}^{w\times h\times 3}$, we have the tracked bounding box $B_{k-1} \in \mathbb{R}^{2\times 2}$ and the shape vector with L landmarks $V_{k-1} \in \mathbb{R}^{L\times 2}$ of previous frame. The purpose of deformable face tracking is to predict the bounding box B_k and facial shape V_k for the current frame I_k. This task aims to learn a joint probability of face bounding box generation and face landmark detection. Following the Bayesian formulation, the joint probability are derived as follows:

$$p(B_k, V_k|I_k, B_{k-1}, V_{k-1}) = p(B_k|I_k, B_{k-1}, V_{k-1})p(V_k|B_k, I_k, B_{k-1}, V_{k-1}) \quad (1)$$

Since the joint probability $p(x, y)$ can be computed in two equivalent ways: $p(x, y) = p(x)p(y|x) = p(y)p(x|y)$, ideally the conditional probabilities in deformable face tracking problem should satisfy the following equality (we omit B_{k-1}, V_{k-1} for simplicity):

$$p(B_k|I_k)p(V_k|B_k, I_k) = p(V_k|I_k)p(B_k|V_k, I_k) \quad (2)$$

We call this *probabilistic duality*, which is a necessary condition for the optimality of the learned dual models.

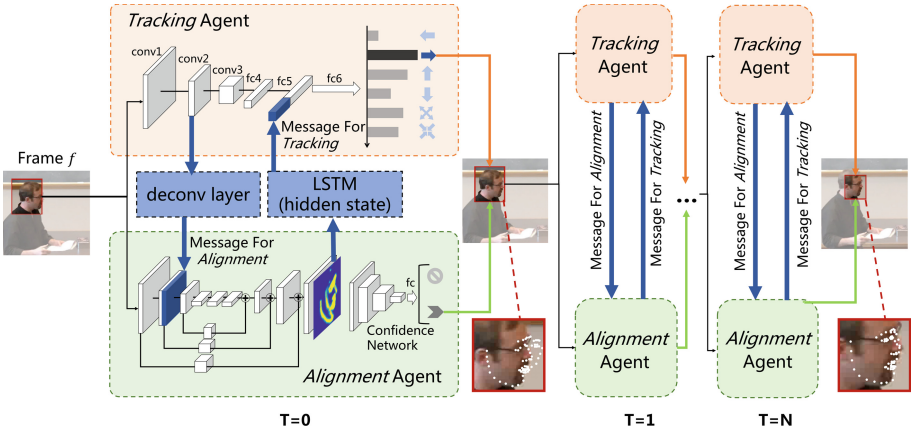


Fig. 2. The architecture of our proposed DADRL. Our DADRL consists of two agents: a tracking agent and an alignment agent. Each agent has a discrete action set. The communicated messages are encoded by a deconvolution layer and a LSTM unit, respectively. These two agents decide a sequence of actions to adjust the target face's bounding box and regress the facial landmarks simultaneously. The agents go to the next frame until the detected facial landmarks are finalized. Note that, T denotes the iteration number of MDP, rather than time-stamps number of the video.

Most existing deformable face tracking methods model the joint probability as the decomposed form in Eq. 1. Since these two models are learned individually, there is no guarantee that the probabilistic duality will hold. To tackle this problem, we propose to explicitly reinforce the empirical probabilistic duality of these two models. We consider the learning objectives of bounding box generation and facial landmark detection as two conditional probabilities. Then the connections between these two tasks can be formulated as two marginal distributions. To satisfy the probabilistic duality during the training, one possible solution is to design a term in the loss function as an appropriate constraint, such as the regularization term in [32], and train the dual models by using standard supervised learning techniques. However, as the ground-truth marginal distributions are not available, the empirical marginal distributions are usually utilized to fulfill the constraint. This is a sub-optimal strategy as the marginal distributions are fixed during training.

Inspired by the observation that marginal distributions should be learned, we propose a deep reinforcement learning framework for deformable face tracking. These two tasks are considered as dual agents. The communicated messages between them are regarded as alternatives of two marginal distributions to satisfy the probabilistic duality. The learning of message channels is facilitated by using the deep Q-learning algorithm.

Our proposed DADRL is different from the following two learning schemes, as illustrated Fig. 3: (1) Tracking-by-detection focuses on single-task learning which has no guarantee to hold the probabilistic duality; (2) Multi-task learning has

an assumption that these two tasks share the same input space and coherent feature representation, which is too strong in practical application. Different from these learning schemes, our DADRL disentangles the connection of two tasks and explicitly exploits the synergy between them.

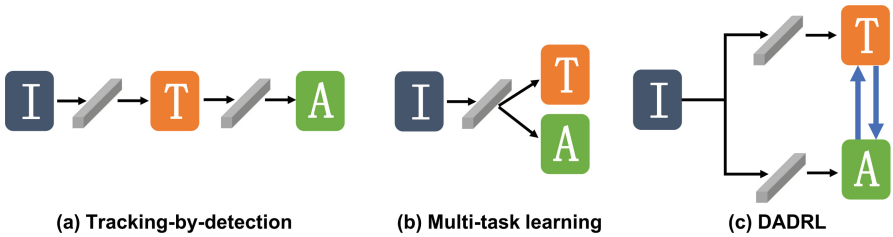


Fig. 3. Comparisons of three strategies for deformable face tracking. I, T, and A denote an image frame, bounding box tracking, and face alignment, respectively. (a) Tracking-by-detection processes deformable face tracking in a serial manner, which has no guarantee to hold the probabilistic duality. (b) Multi-task learning assumes that two tasks share the same input space and coherent feature representation, which is too strong in many real applications. (c) Our DADRL explicitly exploits the synergy between these two tasks.

3.2 Dual-Agent Deep Reinforcement Learning

Our DADRL consists of two agents: a tracking agent and an alignment agent. Each agent has a discrete action set. The basic pipeline is as follows: for each frame in the video, firstly the state is initialized by the terminal state of the previous frame. Then, based on the observed state and the received message, these two agents decide a sequence of actions to adjust the target face's bounding box and regress facial landmark coordinates simultaneously. Lastly, the agents go to the next frame until the detected facial landmarks are finalized. Figure 2 illustrates the pipeline of our method.

We formulate our strategy as MDP for each frame in the video. We start by introducing the state definition, which is shared by two agents, followed by the other respective definitions of two agents. We omit the subscript k when we describe MDP in each frame for simplicity.

State: s_t is defined as the current image region extracted by the bounding box, which is resized to a fixed size. Given the frame I and the current bounding box B, the state s_t is formulated as follows:

$$s_t = \phi(B, I) \tag{3}$$

where ϕ denotes the patch-extracting function.

Action: Based on the state s_t, each agent outputs an action a_t. There are totally eight types of actions for two agents, including *movement* actions and *stop/continue* actions, as shown in Fig. 4.

Tracking Agent: The tracking agent aims to produce *movement* actions to change the current observed region. Specifically, the set of actions are defined as: $\{left, right, up, down, scale\,up, scale\,down\}$.

Alignment Agent: The alignment agent produces *stop/continue* actions to determine whether the iteration should be terminated. Thus, the termination of the search process is in light of face alignment quality, rather than the tracked bounding box result.

Fig. 4. The defined actions of two agents. Left: *movement* actions for the tracking agent. Right: *stop/continue* actions for the alignment agent.

State Transition: Having decided the action at the state s_t, the next state s_{t+1} is obtained by the state transition function.

Tracking Agent: For the *movement* actions of the tracking agent, the new state s_{t+1} is obtained by shifting the bounding box with a discrete change, which is relative to the current size of the bounding box as follows:

$$\delta_w = \alpha(x_2 - x_1), \quad \delta_h = \alpha(y_2 - y_1) \tag{4}$$

where $\alpha \in [0, 1]$ denotes a scale vector, $\{x_1, y_1, x_2, y_2\}$ denotes the bounding box coordinates of top-left and bottom-right vertices. The bounding box B is updated by adding or removing δ_w or δ_h to the coordinates according to the output action. For example, if $left$ action is selected, the position of B moves to $\{x_1 - \delta_w, y_1, x_2 - \delta_w, y_2\}$ and *scale up* action changes B into $\{x_1 - \frac{1}{2}\delta_w, y_1 - \frac{1}{2}\delta_h, x_2 + \frac{1}{2}\delta_w, y_2 + \frac{1}{2}\delta_h\}$.

Alignment Agent: For the alignment agent, if a *stop* action is selected, the face alignment result is finalized as the target of the current frame, and the bounding box result is transferred to the initial state of the next frame. The *continue* action continues the iteration of MDP.

Reward: The rewards of the agent depend on the chosen action a_t at state s_t, which are determined by the function r_t.

Tracking Agent: The reward function r_t reflects the landmark detection accuracy improvements. The reward function measures the misalignment descent and is defined as follow:

$$r_t = -sign(d_{t+1} - d_t), \; d_t = \frac{\sum_{i=1}^{L} \|\hat{V}_{i,t} - V_i^*\|}{L \cdot \zeta} \tag{5}$$

where d_t denotes the normalized point-to-point distance for the t-th iteration of MDP, $\|\cdot\|$ specifies the ℓ_2 norm, ζ denotes the normalizing factor, $\hat{V}$, V^* denote the predicted landmarks points and the ground truth, respectively.

Alignment Agent: For the *continue* action, we use the same reward as the tracking agent. For the *stop* action, we use a different reward scheme because it leads to a terminate state, which is defined as:

$$r_t = \begin{cases} +\eta & \text{if } d_t < \tau \\ -\eta & \text{otherwise} \end{cases} \tag{6}$$

where η is empirically set to 3.0, and τ is a threshold that indicates the maximum error allowed to consider the predicted alignment result as a positive one.

3.3 Network Architecture

The DADRL network consists of three parts: the tracking agent, the alignment agent, and communicated message channels. The tracking agent is a VGG-M model followed by a one-layer Q network. The alignment agent is designed as a combination of stacked hourglass network and a confidence network. Two communicated messages are encoded by a deconvolution layer and a Long Short-Term Memory (LSTM) unit respectively. In this section, we detail communicated message channels and the confidence network, and will detail tracking agent and stacked hourglass network in Sect. 4.2.

Communicated Message Channels: The communicated messages explicitly encode the synergic information flows between these two agents. For the message passed from the tracking agent to the alignment agent, we aim to provide prior additional textural information for the alignment agent to improve the robustness. We select the output feature map of the *conv*3 layer in the tracking agent, and concatenate it in depth axis with the feature map which is the output of the first down-sampling step in the hourglass network. We adopt a deconvolution layer as message channel to match the sizes of feature maps.

The message passed from the alignment agent to the tracking agent provides complementary 3D pose information for bounding box tracking. The primary goal is to produce auxiliary knowledge of facial pose for accurate tracking. To achieve this, we take the normalized coordinates of predicted landmark points as a representation of 3D pose information. We also adopt LSTM to memorize the pose variation through time series. The hidden state is not updated until the MDP of one frame is terminated for training stabilization.

Confidence Network: We observe that landmark prediction is usually formulated as a regression problem, which has no confidence score as estimated in classification problems. However, it is necessary for the alignment agent to judge the quality of predicted landmarks and determine whether to continue the adjustment process. For example, in cases that predicted landmarks is obviously

implausible due to an inaccurate bounding box, the regression result of the alignment agent should have a low confidence score and be considered as a failure. Inspired by this observation, we propose the confidence network to determine the termination of iterations for these two agents. The proposed confidence network takes the predicted heatmap and shape-indexed local patches as the input, and outputs a $L \times 1$ vector, which represents the confidence of each landmark. Followed by a one-layer fully connected Q-net, Q values of *stop*/*continue* actions are predicted for the alignment agent.

3.4 Network Training

As training via reinforcement learning directly from scratch is significantly slow to converge, we exploit a two-stage training procedure: firstly utilize supervised learning to pre-train main branch of the network, then train the other parts via reinforcement learning.

Supervised Learning Stage: For the supervised learning stage, two agents are trained separately and elements of message vectors are set to zero. For the tracking agent, training samples which consist of image patches $\{p_i\}$ and action labels $\{a_i^*\}$ are fed into the network. The image patches are sampled from the training dataset by adding Gaussian noise to the ground truth patches, which are the tightest bounding box of the annotated facial landmarks. The corresponding action label a_i^* is assigned by $a_i^* = \arg\max_a IoU(f(p_i, a), G)$, where $f(p_i, a)$ denotes the moved patch from p_i by the action a from the action set of tracking agent, G denotes the ground truth patch. The loss function for tracking agent is defined as follows,

$$L_{tracking} = CrossEntropy(\hat{a}_i, a_i^*) \tag{7}$$

where $\hat{a}_i$ denotes the predicted action of tracking agent.

For the alignment agent, the loss function of hourglass model is presented as:

$$L_{alignment} = \frac{1}{L}\sum_{n=1}^{L}(\sum_{ij}||h_n(i,j) - h_n^*(i,j)||_2^2) \tag{8}$$

where $h_n(i,j)$, $h_n^*(i,j)$ represent the predicted and the ground truth heatmap at pixel location (i,j) for the n-th landmark respectively.

Reinforcement Learning Stage: The reinforcement learning stage aims to train parameters of Q-nets, message channels and confidence network simultaneously. Following the Q-learning algorithm, each agent chooses an action according to the current estimation of the Q-function $Q(s,a)$ in an iterative fashion.

Based on $Q(s, a)$, the agent will choose the action that is associated to the highest reward. Q-learning iteratively updates the action-selection policy using the Bellman as follows:

$$Q(s, a) = r + \gamma \max_{a'} Q(s', a') \tag{9}$$

where s and a are the current state and action, γ represents the discount factor.

In our work, we approximate the Q-function by deep Q-network trained with reinforcement learning. In the dual-agent setting, deep Q-network also takes the message received from the other agent as input, formulated as $Q(s, a, m)$. In order to reach the Bellman optimality, we jointly perform sampling to these two agents and the samples are used to update all parameters by jointly minimizing the following loss,

$$L = \mathbb{E}[Q(s_t, a_t) - (r_t + \gamma \max_{a'} Q(s_{t+1}, a'))]^2 \tag{10}$$

The parameters related to message channels between these two agents are also updated because the messages are differential.

4 Experiments and Results

We evaluated the performance of the proposed DADRL on the large-scale face tracking dataset, the 300-VW test set [4], which is one publicly available large scale face tracking dataset. We compared our method with state-of-the-arts, and reported several analyses to investigate the importance of message passing in the dual-agent learning manner in Sect. 4.3. Our results demonstrate the effectiveness of interaction between two tasks.

4.1 Dataset and Settings

The 300-VW dataset consists of 3 categories: 1 (62,135 frames), 2 (32,805 frames), and 3 (26,338 frames). The Category 3 is by far the most challenging, and contains 14 videos in severe wild conditions and each video lasts around one minute (25–30 images per second). We conducted our experiments on Category 3 to study the improved performance of our method on severe conditions including large pose, heavily occlusion, etc. Results were reported for both the 49 inner points and the whole 68 points. Note that, there are several existing evaluation protocols and different versions of annotations for the dataset, such as [3,4]. For fair comparison, we followed the dataset and setting in the original 300VW competition of [4]. The other reported results also follow the same setting.

During supervised learning stage, the two agents were trained separately. We utilized all training data from the 300-W competition [34] to train the alignment agent, and the 300-VW training set to train the tracking agent. During reinforcement learning stage, the whole network was trained with the data of

300-VW training set. We noticed a newly set-up face tracking competition [35] with released 3D projected annotated facial landmarks of 300-VW dataset. We also trained another model with the 3D data and compared it with state-of-the-art methods in Sect. 4.3. For the evaluation protocols, we employed the standard normalized root mean squared error (RMSE) and cumulative error distribution (CED) curves.

4.2 Implementation Details

Our model was built based on the popular accelerated deep learning toolbox TensorFlow [36], which mainly operates on data flow graphs. The network of tracking agent is initialized by the pre-trained VGG-M model. The feature extracted by the pre-trained CNN is trained with ImageNet [37], which helps the parameters of the Q-Network to converge faster. The input fixed size of state s_t is 112×112. As illustrated in Fig. 2, the network consists of three convolutional layers $\{conv1, conv2, conv3\}$, which are identical to the convolutional layers in VGG-M model, and three fully connected layers $\{fc4, fc5, fc6\}$. $\{fc4, fc5\}$ layers are combined with ReLU and dropout layers, and the output of $fc5$ layer is concatenated with the message received from alignment agent. The final $fc6$ layer, without any activation function, predicts the Q value of the six *movement* actions, in order to determine the action of tracking agent for the current iteration.

The basic network of alignment agent is designed based on stacked hourglass network [38]. The original signals are branched out before each down-sampling step and combined before each up-sampling step. Features from the original size to $1/2^n$ size are able to extracted for n scale hourglass model. The output of hourglass model is a set of heatmaps, each of which represents the probability of one keypoint's presence at each pixel. We choose $n = 2$ for the trade-off of accuracy and speed.

For Confidence Network, we concatenate the extracted shape-index patches and the predicted heatmaps, and resize them to 26×26 as input. Then we deploy two convolutional layers (3×3 kernel size, 1×1 stride) with 128 and 512 kernels. By following the convolution layers, we append a two-layer fully connections, where the parameters are 512×512 and $512 \times L$ vector matrices ($L = 68$ for 300-VW dataset). The output vector is fed into a one-layer fully connection to predict the Q value of *stop*/*continue* actions.

For hyper-parameters during training process, we specified the learning rates to 0.001, the discounted factor to 0.9, and mini-batch size to 20. For parameters in MDP, the scale vector α was set to 0.2, the threshold τ was set to 0.06, ε was set to 0.7. A replay buffer [33] is used for reinforcement learning stage.

4.3 Results and Analysis

Comparison with State-of-the-Arts: In this section, we compared DADRL with state-of-the-arts for both the 49 inner points and 68 points. For 49 inner points, we compared DADRL with 5 state-of-the-art methods including the two

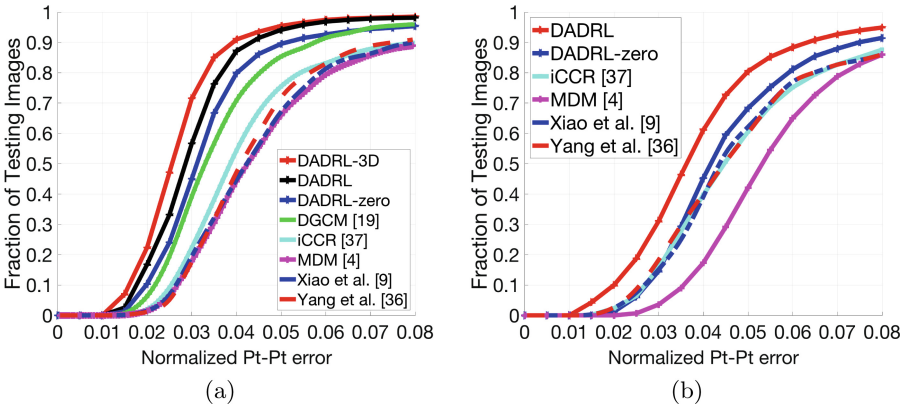


Fig. 5. (a) Comparison between DADRL and state-of-the-arts on Category 3 of 300-VW for 49 inner points. (b) Comparison between DADRL and state-of-the-arts on Category 3 of 300-VW for 68 points.

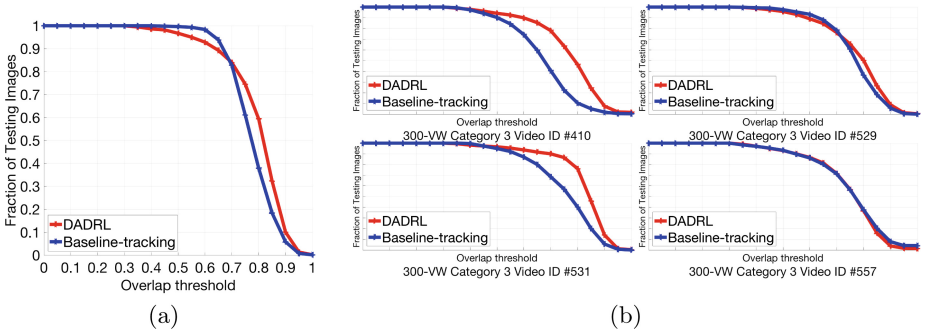
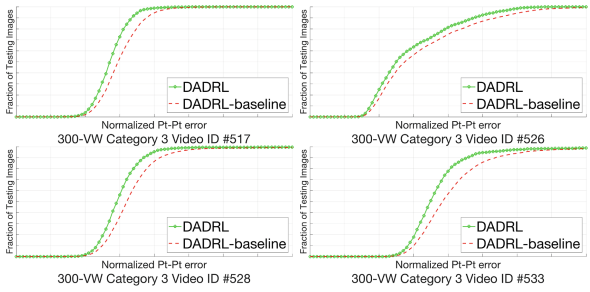


Fig. 6. Bounding box tracking comparison. (a) Success plots for all videos on Category 3 of 300-VW. (b) Success plots for several videos with extreme pose variation on Category 3 of 300-VW.

best performing methods of the 300-VW competition [2,39], the state-of-the-art face alignment method of [16], the state-of-the-art tracker of [40] and a synergy method DGCM of [19]. We introduced a baseline, called 'DADRL-zero', where the communicated messages are set to zero during test phase. As the newly released 3D projected annotated landmarks in [35] has the same inner-points position as the previous 2D annotation, we also reported the result of the model trained by these 3D data, named 'DADRL-3D'. Figure 5(a) shows the obtained results on Category 3. The proposed 'DADRL-3D' is the best performing method, followed by 'DADRL', while the baseline DADRL-zero shows comparable performance with other methods. The large margin between 'DADRL' and other state-of-arts demonstrates the effectiveness of the interactive manner, as the intrinsic correlation between two agents could be held. It is reasonable

that 'DADRL-3D' outperforms all state-of-the-arts because the model is more robust to large pose by training with 3D data. The comparison of the proposed 'DADRL' and the baseline 'DADRL-zero' illustrates the importance of the communicated messages.

We also reported our results for the whole 68 points. As the 'DADRL-3D' has an output of 84 points, we did not consider it for the 68-points condition. Compared with 49-inner-points setting, the 68-points setting could better demonstrate the robustness of the methods, as the contour points are more sensitive to extreme conditions. As DGCM of [19] did not report results for 68 points, we did not compare our method with it. As illustrated in Fig. 5(b), our proposed DADRL outperforms other methods by a large margin.



Vi./Meth.	Baseline	DADRL
#517	3.29	**2.75**
#526	3.60	**2.92**
#528	3.70	**3.01**
#533	3.78	**3.68**

Fig. 7. CED curves (left) and averaged errors comparisons (100%) (right) of several videos with heavy occlusions and motion artifacts on Category 3 of 300-VW.

Analysis: In this section, we performed two analyses to illustrate how communicated messages in the dual-agent training manner improve the performance of both bounding box tracking and face alignment. As the comparison between DADRL and the conducted baseline 'DADRL-zero' has already illustrated the importance of communicated messages, we further investigated how two messages assist the respective agent. Two experiments demonstrate that the interaction between two tasks enhances the robustness to extreme conditions for deformable face tracking.

Tracking Experiments: The message passing from alignment agent to tracking agent aims to provide complementary 3D pose information for accuracy bounding box tracking. To verify the effectiveness, we trained another tracking network as baseline. This network has the same architecture as the tracking agent, except the output of the $fc5$ layer is no more concatenated with message code. The network was trained in the same manner as the tracking agent, namely, firstly pretrained by supervised learning, then fine-tuned by reinforcement learning. The baseline tracker also follows MDP and has the same action set as the tracking agent of DADRL. A similar sequence of bounding box shifting is predicted by the baseline network. As there is no *stop* action for this tracker, the selected

face region is fed into our DRDAL to determine whether the iteration should stop by our alignment agent. The only difference between this baseline bounding box tracker and our tracking agent is that there is no message input for the baseline. For comparison, we employed success rate as evaluation protocol. As there is no annotated bounding box in 300-VW dataset, we considered the tightest bounding box of the facial landmarks as ground truth. Success plots of Category 3 are shown in Fig. 6(b). We also illustrated success plots of several individual videos, shown in Fig. 6(a). Note that these videos contain faces which undergo extreme pose, even totally turn-around. The results show that the message from alignment agent is an effective complementary 3D information for accurate 2D tracking and can enhance the robustness of tracking agent to large pose. Examples of sequential actions decided by tracking agents are shown in Fig. 8(c).

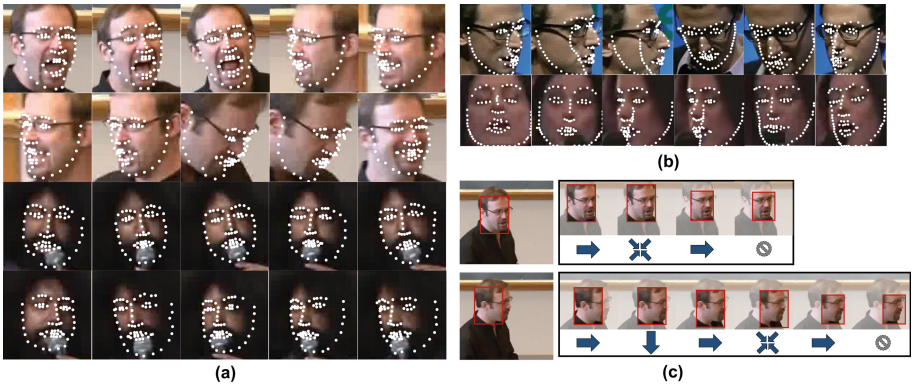


Fig. 8. (a)(b) Examples of alignment results for 68 points and 3D projected 84 points on Category 3 of 300-VW. (c) Sequential actions decided by tracking agent for two frames in Video #533 of 300-VW Category 3.

Alignment Experiments: For better understanding the effect of the message passing to alignment agent, we trained a separated stacked hourglass model as baseline which predicts landmarks without any received message. This baseline model was trained the same way as the supervised learning stage of DADRL. During test phase, we directly used the tightest bounding box of annotated landmarks as the input face region. Two models predicted landmarks with only one feed-forward pass. To verify that this message channel provides prior additional textural information for the alignment agent, we selected several videos which contain frames under occlusions or motion artifacts. The comparison of CED curves and averaged point-to-point error is shown in Fig. 7. We can see the alignment agent with a message input has about 2% performance improvement over the single hourglass model, which demonstrates the robustness to occlusions and motion artifacts of our DADRL. The results further show that the

message passing from tracking agent to alignment agent is able to decode the textural information, which is an effective prior information for face alignment under heavy inclusions or motion artifacts. Examples of alignment results are shown in Fig. 8(a) (b), for 68 points and 3D 84 points respectively. In summary, the results of these two experiments suggest that the communicated messages play an important role in our proposed method.

5 Conclusion

In this paper, we have proposed a dual-agent deep reinforcement learning (DADRL) method for deformable face tracking. In our method, we have explicitly exploited the interaction between bounding box generation and face alignment by following a Bayesian model and have proposed a unified framework to simultaneously perform these two tasks. By formulating the problem as MDP, we have defined these two models as dual agents to exploit the relationships and pass messages via an adaptive sequence of actions. The models are trained interactively via deep reinforcement learning. Experimental results have been presented to show the effectiveness of the proposed approach. How to automatically choose the message channels and to further improve the performance of our method seems to be an interesting future work.

Acknowledgements. This work was supported in part by the National Key Research and Development Program of China under Grant 2017YFA0700802, in part by the National Natural Science Foundation of China under Grant 61822603, under Grant 61672306, Grant U1713214, Grant 61572271, and in part by the Shenzhen Fundamental Research Fund (Subject Arrangement) under Grant JCYJ20170412170602564.

References

1. Wang, X., Yang, M., Zhu, S., Lin, Y.: Regionlets for generic object detection. In: ICCV, pp. 17–24 (2013)
2. Xiao, S., Yan, S., Kassim, A.A.: Facial landmark detection via progressive initialization. In: ICCVW, pp. 33–40 (2015)
3. Chrysos, G.G., Antonakos, E., Snape, P., Asthana, A., Zafeiriou, S.: A comprehensive performance evaluation of deformable face tracking "in-the-wild". IJCV **126**(2–4), 198–232 (2018)
4. Shen, J., Zafeiriou, S., Chrysos, G.G., Kossaifi, J., Tzimiropoulos, G., Pantic, M.: The first facial landmark tracking in-the-wild challenge: Benchmark and results. In: ICCVW, pp. 50–58 (2015)
5. Asthana, A., Zafeiriou, S., Cheng, S., Pantic, M.: Incremental face alignment in the wild. In: CVPR, pp. 1859–1866 (2014)
6. Peng, X., Zhang, S., Yang, Y., Metaxas, D.N.: PIEFA: personalized incremental and ensemble face alignment. In: ICCV, pp. 3880–3888 (2015)
7. Peng, X., Feris, R.S., Wang, X., Metaxas, D.N.: A recurrent encoder-decoder network for sequential face alignment. In: Leibe, B., Matas, J., Sebe, N., Welling, M. (eds.) ECCV 2016. LNCS, vol. 9905, pp. 38–56. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-46448-0_3

8. Liu, H., Lu, J., Feng, J., Zhou, J.: Two-stream transformer networks for video-based face alignment. TPAMI (2017). https://doi.org/10.1109/TPAMI.2017.2734779
9. Black, M.J., Yacoob, Y.: Tracking and recognizing rigid and non-rigid facial motions using local parametric models of image motion. In: 1995 Proceedings of Fifth International Conference on Computer Vision, pp. 374–381. IEEE (1995)
10. Chrysos, G.G., Antonakos, E., Zafeiriou, S., Snape, P.: Offline deformable face tracking in arbitrary videos. In: ICCVW, pp. 1–9 (2015)
11. Decarlo, D., Metaxas, D.: Optical flow constraints on deformable models with applications to face tracking. IJCV **38**(2), 99–127 (2000)
12. Tzimiropoulos, G.: Project-out cascaded regression with an application to face alignment. In: CVPR, pp. 3659–3667 (2015)
13. Cao, X., Wei, Y., Wen, F., Sun, J.: Face alignment by explicit shape regression. IJCV **107**(2), 177–190 (2014)
14. Xiong, X., De la Torre, F.: Supervised descent method and its applications to face alignment. In: CVPR, pp. 532–539 (2013)
15. Zhang, Z., Luo, P., Loy, C.C., Tang, X.: Facial landmark detection by deep multi-task learning. In: Fleet, D., Pajdla, T., Schiele, B., Tuytelaars, T. (eds.) ECCV 2014. LNCS, vol. 8694, pp. 94–108. Springer, Cham (2014). https://doi.org/10.1007/978-3-319-10599-4_7
16. Trigeorgis, G., Snape, P., Nicolaou, M.A., Antonakos, E., Zafeiriou, S.: Mnemonic descent method: a recurrent process applied for end-to-end face alignment. In: CVPR, pp. 4177–4187 (2016)
17. Zhang, J., Shan, S., Kan, M., Chen, X.: Coarse-to-fine auto-encoder networks (CFAN) for real-time face alignment. In: Fleet, D., Pajdla, T., Schiele, B., Tuytelaars, T. (eds.) ECCV 2014. LNCS, vol. 8690, pp. 1–16. Springer, Cham (2014). https://doi.org/10.1007/978-3-319-10605-2_1
18. Kumar, A., Chellappa, R.: Disentangling 3D pose in a dendritic cnn for unconstrained 2D face alignment. arXiv preprint arXiv:1802.06713 (2018)
19. Khan, M.H., McDonagh, J., Tzimiropoulos, G.: Synergy between face alignment and tracking via discriminative global consensus optimization. In: ICCV 2017, pp. 3791–3799 (2017)
20. Littman, M.L.: Reinforcement learning improves behaviour from evaluative feedback. Nature **521**(7553), 445 (2015)
21. Gu, S., Lillicrap, T., Sutskever, I., Levine, S.: Continuous deep q-learning with model-based acceleration. In: ICML, pp. 2829–2838 (2016)
22. Mnih, V., et al.: Playing atari with deep reinforcement learning. arXiv preprint arXiv:1312.5602 (2013)
23. Mnih, V., et al.: Human-level control through deep reinforcement learning. Nature **518**(7540), 529 (2015)
24. Ammar, H.B., Eaton, E., Ruvolo, P., Taylor, M.: Online multi-task learning for policy gradient methods. In: ICML, pp. 1206–1214 (2014)
25. Silver, D., Lever, G., Heess, N., Degris, T., Wierstra, D., Riedmiller, M.: Deterministic policy gradient algorithms. In: ICML (2014)
26. Rao, Y., Lu, J., Zhou, J.: Attention-aware deep reinforcement learning for video face recognition. In: CVPR, pp. 3931–3940 (2017)
27. Yu, L., Zhang, W., Wang, J., Seqgan, Y.Y.: Sequence generative adversarial nets with policy gradient. arXiv preprint arXiv:1609.05473 **2**(3), 5 (2016)
28. Yoo, S.Y.J.C.Y., Yun, K., Choi, J.Y.: Action-decision networks for visual tracking with deep reinforcement learning (2017)
29. Foerster, J., Assael, Y., de Freitas, N., Whiteson, S.: Learning to communicate with deep multi-agent reinforcement learning. In: NIPS, pp. 2137–2145 (2016)

30. Sukhbaatar, S., Fergus, R., et al.: Learning multiagent communication with backpropagation. In: NIPS, pp. 2244–2252 (2016)
31. Kong, X., Xin, B., Wang, Y., Hua, G.: Collaborative deep reinforcement learning for joint object search. In: CVPR (2017)
32. Xia, Y., Qin, T., Chen, W., Bian, J., Yu, N., Liu, T.Y.: Dual supervised learning. arXiv preprint arXiv:1707.00415 (2017)
33. Lillicrap, T.P., et al.: Continuous control with deep reinforcement learning. arXiv preprint arXiv:1509.02971 (2015)
34. Sagonas, C., Tzimiropoulos, G., Zafeiriou, S., Pantic, M.: A semi-automatic methodology for facial landmark annotation. In: CVPRW, pp. 896–903 (2013)
35. Zafeiriou, S., Chrysos, G.G., Roussos, A., Ververas, E., Deng, J., Trigeorgis, G.: The 3D menpo facial landmark tracking challenge. In: ICCVW, vol. 5 (2017)
36. Abadi, M., et al.: TensorFlow: large-scale machine learning on heterogeneous systems (2015). tensorflow.org
37. Krizhevsky, A., Sutskever, I., Hinton, G.E.: Imagenet classification with deep convolutional neural networks. In: NIPS, pp. 1097–1105 (2012)
38. Newell, A., Yang, K., Deng, J.: Stacked hourglass networks for human pose estimation. In: Leibe, B., Matas, J., Sebe, N., Welling, M. (eds.) ECCV 2016. LNCS, vol. 9912, pp. 483–499. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-46484-8_29
39. Yang, J., Deng, J., Zhang, K., Liu, Q.: Facial shape tracking via spatio-temporal cascade shape regression. In: ICCVW, pp. 41–49 (2015)
40. Sánchez-Lozano, E., Martinez, B., Tzimiropoulos, G., Valstar, M.: Cascaded continuous regression for real-time incremental face tracking. In: Leibe, B., Matas, J., Sebe, N., Welling, M. (eds.) ECCV 2016. LNCS, vol. 9912, pp. 645–661. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-46484-8_39