Learning Deep Sharable and Structural Detectors for Face Alignment

Hao Liu, Jiwen Lu, *Senior Member, IEEE*, Jianjiang Feng, *Member, IEEE*, and Jie Zhou, *Senior Member, IEEE*

***Abstract*— Face alignment aims at localizing multiple facial landmarks for a given facial image, which usually suffers from large variances of diverse facial expressions, aspect ratios and partial occlusions, especially when face images were captured in wild conditions. Conventional face alignment methods extract local features and then directly concatenate these features for global shape regression. Unlike these methods which cannot explicitly model the correlation of neighbouring landmarks and motivated by the fact that individual landmarks are usually correlated, we propose a deep sharable and structural detectors (DSSD) method for face alignment. To achieve this, we firstly develop a structural feature learning method to explicitly exploit the correlation of neighbouring landmarks, which learns to cover semantic information to disambiguate the neighbouring landmarks. Moreover, our model selectively learns a subset of sharable latent tasks across neighbouring landmarks under the paradigm of the multi-task learning framework, so that the redundancy information of the overlapped patches can be efficiently removed. To better improve the performance, we extend our DSSD to a recurrent DSSD (R-DSSD) architecture by integrating with the complementary information from multi-scale perspectives. Experimental results on the widely used benchmark datasets show that our methods achieve very competitive performance compared to the state-of-the-arts.**

***Index Terms*— Face alignment, deep learning, biometrics.**

I. Introduction

FACE alignment (*a.k.a.* facial landmark detection) aims at localizing multiple facial landmarks for a given facial image, which is a key step for many facial analysis tasks, such as face verification [1], [2], face recognition [3] and facial attribute analysis [4]. While extensive efforts have been devoted, face alignment still remains a challenging problem due to large variations of facial expressions, aspect ratios and diverse partial occlusions.

Conventional face alignment methods can be categorized into two classes: discriminative fitting-based and cascaded regression-based. Discriminative fitting-based methods [5]–[7] estimate facial landmarks by maximizing the joint posterior probability over all landmarks for a given face image. However, these methods are usually slow in terms of the efficiency compared with the improvements in the accuracy. To address this, cascaded regression-based methods [8]–[12] learn to seek a series of simple feature-to-shape mappings to refine the initial shape to the final shape progressively. While these methods have achieved fast alignment speed [9]–[11], their performance is still not satisfying because these methods usually employ linear feature-to-shape mappings, so that they are not powerful enough to directly model the nonlinear relationship between face samples and facial shapes. Moreover, the features employed in these methods are hand-crafted, which requires strong prior knowledge by hand. To address both issues, deep learning has been employed for face alignment [13]–[21], which formulates face alignment as a cascaded regression problem and employs the deep neural networks to exploit the complex and nonlinear image-to-shape mappings. However, previous face alignment methods may incur the ambiguity between neighbouring landmarks due to the lack of local discriminative information, because the concatenation of these local features cannot explicitly model the correlation of neighbouring landmarks. Moreover, the cropped pose-index patches are usually overlapped because of the correlated landmarks, which occurs some redundancy information for facial landmark localization.

Unlike conventional face alignment methods which cannot explicitly exploit the correlation of neighbouring landmarks and motivated by the individual facial landmarks are usually correlated especially for densely mark-up facial landmarks, in this paper, we propose a deep sharable and structural detectors (DSSD) method, where both the structural and sharable information are exploited for facial landmark localization. Specifically, we first develop a structural feature learning method to learn discriminative features directly from raw local patches, which enlarges the window sizes to cover the semantic information, *e.g.*, facial part-based details covering eyes, nose, mouths and partial face counter, and disambiguates the positions of neighbouring landmarks. Moreover, our model employs a multi-task learning framework to selectively learn a subset of latent tasks shared across neighbouring landmarks

Manuscript received March 27, 2016; revised September 9, 2016 and December 12, 2016; accepted January 18, 2017. Date of publication January 22, 2017; date of current version February 17, 2017. This work was supported in part by the National Key Research and Development Program of China under Grant 2016YFB1001001, in part by the National Natural Science Foundation of China under Grant 61672306, Grant 61225008, Grant 61572271, Grant 61527808, Grant 61373074, and Grant 61373090, in part by the National 1000 Young Talents Plan Program, the National Basic Research Program of China under Grant 2014CB349304, in part by the Ministry of Education of China under Grant 20120002110033, and in part by the Tsinghua University Initiative Scientific Research Program. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Guoliang Fan. *(Corresponding author: Jiwen Lu.)*

H. Liu is with the Department of Automation, Tsinghua National Laboratory for Information Science and Technology, Tsinghua University, Beijing 100084, China (e-mail: h-liu14@mails.tsinghua.edu.cn.

J. Lu, J. Feng, and J. Zhou are with the Department of Automation, State Key Lab of Intelligent Technologies and Systems, Tsinghua National Laboratory for Information Science and Technology, Tsinghua University, Beijing 100084, China (e-mail: lujiwen@tsinghua.edu.cn; jfeng@tsinghua.edu.cn; jzhou@tsinghua.edu.cn).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TIP.2017.2657118

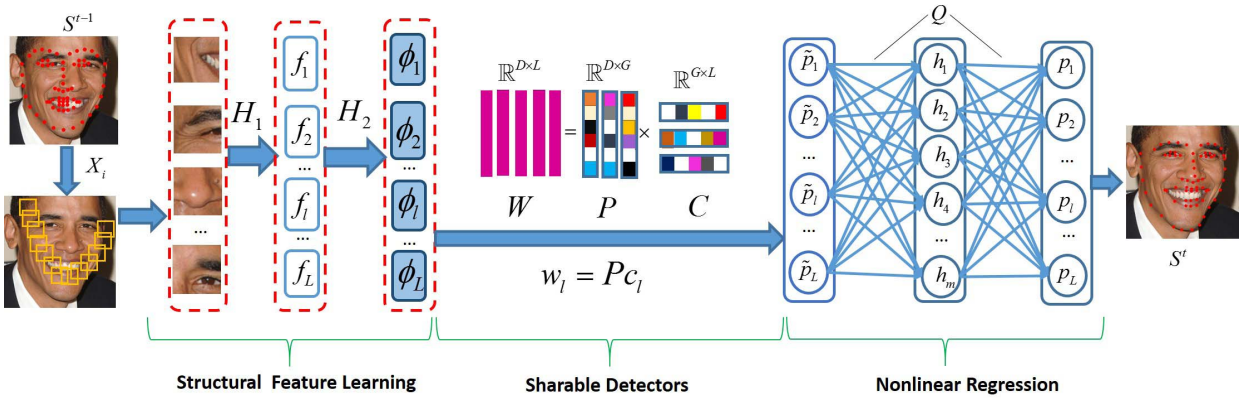


Fig. 1. The work-flow of the proposed DSSD. Shape-index patches cropped from each face image are fed into a tiny CNN deep network, which consists of two convolution layers and two fully-connected layers, respectively. As a result, each shape-index patch is encoded as a local feature f_l by the CNN feature extractor $\mathbf{H}_1$ and is further re-weighted with the fully connected layers $\mathbf{H}_2$ as ϕ_l by using the structural feature learning method. Face alignment aims to localize the landmarks by the shape regressor $\mathbf{W} = [w_1, w_2, \cdots, w_l, \cdots, w_L]$, where each landmark position $\mathbf{p}_l$ can be obtained by feeding ϕ_l to the landmark detector. In our work, the goal of the sharable detectors is to decompose $\mathbf{W}$ into two sparse matrix $\mathbf{P}$ and $\mathbf{C}$, where each landmark detector can be represented by a few sharable latent detectors. Lastly, we develop a nonlinear regression by using a feed-forward neural network $\mathbf{Q}$ to preserve the holistic structure of the facial shape.

in order to eliminate the redundancy information caused by the overlapped patches. Lastly, we employ a nonlinear regression following the outcomes of the sharable detectors, which preserves the global facial shape structure and addresses the face images versus diverse partial occlusions. To better improve the alignment performance, we extend our DSSD to a recurrent DSSD (R-DSSD) architecture, which shares a sequential series of parameters stage by stage and embeds the complementary information with multi-scale facial images for coarse-to-fine face alignment. In order to train the proposed model, both the structural feature learning and the sharable detectors are jointly learned under a unified deep convolutional neural networks (CNN) framework, where the parameters of the sharable detectors are obtained by the two-stage iteration method and those of the deep neural networks are optimized via back-propagation. Fig. 1 shows the main work-flow of our proposed model. Experimental results on the public 300-W dataset [22] show that the proposed models obtains very competitive performance compared with the state-of-the-art face alignment methods in terms of both the accuracy and efficiency.

The main contributions of this work are summarized as follows:

1) We develop a structural feature learning method to explicitly model the correlation of neighbouring landmarks, which learns to cover the semantic details to disambiguate the positions of neighbouring landmarks.
2) We propose a multi-task learning method to learn a subset of latent tasks shared across neighbouring landmarks, so that the redundancy information caused by the overlapped patches is efficiently removed during facial landmark localization.
3) To further improve the performance, our recurrent extension tackles the coarse-to-fine face alignment problem, where the capacity of the deep model is efficiently controlled and the complementary information is extracted from the multi-scale inputs.

II. Related Work

In this section, we briefly review conventional face alignment and face alignment by deep learning, respectively

A. Conventional Face Alignment

Conventional face alignment methods can be mainly classified into two categories: discriminative fitting based and cascaded regression based methods. The discriminative fitting based methods [5], [6], [23], [24] build a holistic fitting template to fit the facial shape for a given input image. Representative methods in this class include active shape model (ASM) [5], active appearance model (AAM) [6], constrained local model (CLM) [7] and Guass-Newton deformable part model (GN-DPM) [24]. However, since they adopted multiple SVR regressors (or SVM classifiers), their speed was usually slow and their computation load were heavy. The cascaded regression based approaches [8]–[12], [23], [25], [26] learn a series of linear feature-to-shape mappings to refine the predicted shape progressively. For example, Cao *et al.* [9] proposed an explicit shape regression method (ESR) for face alignment by using the boosting tree-based feature selection approach. Xiong *et al.* [8] proposed a supervised decent method (SDM) to learn a sequence of feature-to-shape mapping functions to refine the facial shapes. Recently, Zhu *et al.* [12] developed a coarse-to-fine shape searching approach (CFSS) to gradually narrow down the possible shape space, which exhibits a superior performance on the alignment accuracy. Besides, Zhu and Ramanan [27] showed that face detection, landmark detection and pose estimation can be jointly addressed under a unified framework. Zhang *et al.* [17] formulated face detection and face alignment as a multi-task learning problem. Nevertheless, the features employed in these methods are hand-crafted, which are arguably weak to represent local patches and require strong prior knowledge by hand. Moreover, their feature-to-pose mapping functions are either linear or ferns/trees based indexing, which are insufficient to handle realistic facial variations. To address this, our

model automatically extracts useful and discriminative features directly from pixels by integrating the sharable and structural information under a unified deep learning architecture.

B. Face Alignment by Deep Learning

Recently, deep learning has been an active topic in machine learning and computer vision, which shows superior performance in many visual analysis tasks such as handwritten digit recognition [28], [29], object detection [30], [31], visual tracking [32] and scene labeling [33], [34]. More recently, deep learning has been adopted to face alignment [13], [14], [16], [19], [20], [35], [36]. For example, Sun *et al.* [13] proposed a deep convolutional networks cascaded method (DCNC) for facial shape refinement, which consists of one shape initialization stage and two shape refinement stages. However, the cascaded CNN method performs individual refinement for each landmark, which is sensitive to the previous prediction. To address this, several works [16], [19], [36], [37] have been proposed to concatenate the shape-index features to estimate the landmark locations in a coarse-to-fine manner. For example, Zhang *et al.* [16] presented a coarse-to-fine auto-encoder networks (CFAN) method to refine the landmark locations iteratively. Lai *et al.* [36] proposed a deep cascaded regression method for face alignment (DCRFA) under the encoder-decoder (deconvolution) neural networks. Trigeorgis *et al.* [35] developed a mnemonic descent method (MDM) by employing a recurrent process which was applied for end-to-end face alignment.[1] However, these models fail to extract the structural information because the extracted features are simply concatenated together to learn the deep models and the correlation of neighbouring landmarks cannot be explicitly exploited. In contrast to these previous methods, we propose structural and sharable detectors in this work to explicitly achieve the correlation of neighbouring landmarks, which reduces the ambiguity for the positions of the neighboring landmarks. Moreover, the proposed model efficiently eliminates the redundancy information caused by the overlapped patches. Experimental results show the effectiveness of the proposed methods in comparisons with most of the state-of-the-art face alignment approaches.

III. Proposed Method

In this section, we describe our methods DSSD and R-DSSD in details.

A. Deep Sharable and Structural Detectors

Conventional cascaded regression methods [8]–[10] employ hand-crafted feature representations and linear feature-to-shape mapping functions, which are not powerful enough to model the complex and nonlinear relationship between face samples and facial shapes. To address this limitation, deep learning has been applied to face alignment [13], [15], [16], which aims to seek a series of deep nonlinear feature-to-shape mappings to model the nonlinear relationship between face images and facial shapes. Nevertheless, the detection accuracy usually degrades due to the correlation of neighbouring landmarks. Moreover, the cropped local patches are usually overlapped, which causes the redundancy information located at the overlapping regions across the adjacent patches. As a result, the redundancy might be ambiguous for facial landmark localization, especially for neighbouring landmarks. Fig. 2 demonstrates the intuitive examples. To address both issues, in this paper, we propose a *structural feature learning* and the *sharable detectors*, where the structural and sharable information across neighbouring landmarks are exploited for facial landmark localization. We detail the proposed model in the following parts.

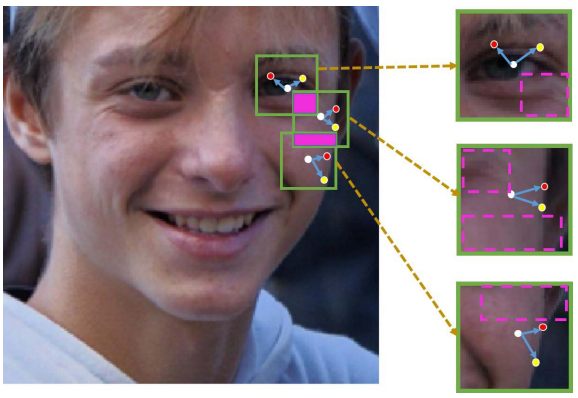


Fig. 2. For the left face image, the white points indicate the current locations and the red points denote the ground-truth. Both the right hand views are the enlarged windows cropped from the left face image, which show the insufficiency for the local regressor that predicts the local landmarks independently without considering the relationship among landmarks. Intuitively, based on the local patch inputs, the local detectors may incorrectly adjust current position onto yellow points which are close to the true landmarks in both appearance and location. Moreover, the concatenating pose-index features-based methods cannot work well due to the redundancy information (marked in magenta rectangles) which is caused by the correlated and overlapped patches (best viewed in the color pdf file).

Suppose that we have a training set $\{(\mathbf{X}_i, \mathbf{S}_i)\}_{i=1}^N$ containing N training samples and an initial shape $\mathbf{S}_i^0$, where $\mathbf{X}_i$ denotes the ith face image consisting of a set of shape-index patches $[\mathbf{x}_1, \mathbf{x}_2, \cdots, \mathbf{x}_l, \cdots, \mathbf{x}_L]$, where $\mathbf{x}_i$ denotes the ith patch of D pixels, and $\mathbf{S}_i = [\mathbf{p}_1, \mathbf{p}_2, \cdots, \mathbf{p}_l, \cdots, \mathbf{p}_L]_i \in \mathbb{R}^L$ denotes the corresponding facial landmarks containing L landmarks, where $\mathbf{p}_l \in \mathbb{R}^2$ denotes lth landmark coordinates. In this work, we map the face image to the facial shape residual $(\mathbf{S}_i^* - \mathbf{S}_i^0)$, which is formulated as the following minimization objective function:

$$\min \sum_i^N \frac{1}{2} \left\| (\mathbf{S}_i^* - \mathbf{S}_i^0) - R(\mathbf{X}_i) \right\|_2^2 \tag{1}$$

where $R(\cdot)$ denotes the shape regression function, *e.g.*, a linear/nonlinear regression [8], [9], [16], which maps the facial shape coordinates from the appearance features. In terms of the appearance features $\mathbf{X}$ in (1), conventional methods [8]–[12], [15], [16] usually employ sampled local features as input and jointly displace these features in a shape-index order [22] to perform the global shape regression. While they have directly addressed the global shape constraint during landmark localization, they cannot explicitly model

[1] At the time writing, we do not have access to the full paper of Trigeorgis *et al.* [35] and therefore cannot take advantage of this work in our experimental comparisons.

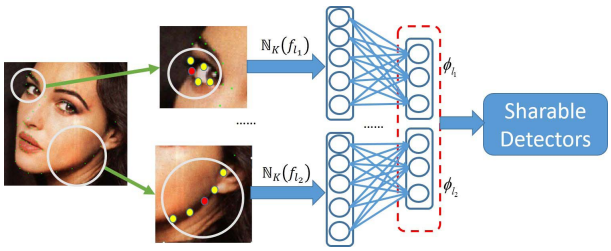


Fig. 3. The illustration of the structural feature learning, where two parts including eyes and partial facial counter are cropped by involving K neighbours. The red points denote the ground-truth landmarks while the white circles denote the k-nearest neighbours marked by yellow points. Let $\mathbb{N}_K(f_{l_1})$ and $\mathbb{N}_K(f_{l_2})$ denotes the l_1th and l_2th K-neighbour feature concatenations. With the proposed structural feature learning method, we see that the landmarks located surrounding the eye (the cropped patch on top) automatically involves the facial part-based details for localization, while those located along the facial counter (the cropped patch below) enlarge the window to cover more details accordingly. Having obtained the learned feature presentations ϕ_{l_1} and ϕ_{l_2}, we feed them to the proposed sharable detectors. (best viewed in the color pdf file).

the correlation of neighbouring landmarks in such cases that the overlapped patches encounter ambiguity during localizing especially for the neighbouring landmarks.

1) Structural Feature Learning: To model the correlation of neighbouring landmarks, we propose a structural feature learning method, which learns a set of discriminative local features from neighbouring landmarks. Specifically, for each individual landmark, our model aims to enlarge the cropping window size by finding K nearest neighbours, which provides more semantic details to disambiguate the positions of neighbouring landmarks. As illustrated in Fig. 3, the cropping windows are enlarged to cover the semantic facial parts such as the left eye part when the landmarks surround the eye parts, while to cover partial facial counter when the landmarks are located alongside the facial counter edge.

To achieve this, we feed the shape-index local patches to a designed CNN to extract the immediate feature representation $\mathbf{F}_i = [f_1, f_2, \cdots, f_l, \cdots, f_L]_i$ for the ith face. For each landmark, we seek K nearest neighbours and re-weight the features of the neighbouring landmarks as $\mathbf{\Phi}_i = [\phi_1, \phi_2, \cdots, \phi_l, \cdots, \phi_L]_i$ by using a feed-forward neural network, which is formulated as follows (ignoring bias):

$$f_l = \text{pool}(\text{ReLU}\,(\mathbf{H}_1 \otimes x_l)) \tag{2}$$

$$\phi_l = \sigma\,(\mathbf{H}_2 \cdot [\mathbb{N}_K\,(f_l)]_{vec}) \tag{3}$$

where $\mathbb{N}_K(\cdot)$ denotes the features in the K-nearest-neighbours, $[\cdot]_{vec}$ performs vector concatenating operation on these neighbouring features, $\otimes$ denotes the CNN convolution, pool$(\cdot)$ operates max pooling, and ReLU$(\cdot)$ denotes the nonlinear function [38]. $\mathbf{H}_2$ denotes the re-weighting parameters with the feed-forward neural networks and σ denotes the nonlinear functions (we employ the *tanh* function). Having obtained the learned features $\mathbf{F}_i$, we directly feed them to the sharable detectors $\mathbf{P}$ and $\mathbf{C}$ to predict the final shape. Furthermore, we deploy a nonlinear regression parameterized by $\mathbf{Q}$ on the outcomes of the sharable detectors (predicted landmarks) to preserve the global shape structure, where the occluded facial landmarks can be estimated by utilizing the non-occluded face part via the global shape regression.

There are two strengths for the proposed structural feature learning method: 1) The correlation of neighbouring landmarks is exploited in the learned features, which reduces the ambiguity for localizing neighbouring landmarks. 2) With the designed K-neighbour structure, the cropping windows are learned to enlarge and cover more semantic details to enhance the discriminativeness of the learned features. Nevertheless, the correlated and overlapped local patches incur some redundancy information to the learned feature representation, which may degrade the face alignment performance.

2) Sharable Detectors: To eliminate the redundancy information, our model aims to learn a subset of latent tasks shared by neighbouring landmarks under the paradigm of the multi-task learning framework [39]. To achieve this, we enforce the sparsity constraint to learn the sharable detectors instead of the global regression mapping. As a result, the overlapped patches are represented by a subset of sharable detectors which shares some visual common pattern according to the appearance information, and then the multiple landmarks are accurately localized.

Let $\mathbf{W} = [w_1, w_2, \cdots, w_l, \cdots, w_L] \in \mathbb{R}^{D\times L}$ denote the parameters of the whole shape regression $R(\cdot)$, which is incorporated with L coefficients. Let $\mathbf{P} = [P_1, P_2, \ldots, P_G] \in \mathbb{R}^{D\times G}$ denote the matrix of the sharable basis, where each column represents a latent component in $\mathbb{R}^D$ and G is the number of latent sharable tasks, $\mathbf{C} = [c_1, c_2, \ldots, c_L] \in \mathbb{R}^{G\times L}$ denotes the weights to represent $\mathbf{W}$ based on the sharable basis $\mathbf{P}$. Hence, the lth coefficient w_l of the global detector can be represented as:

$$w_l = \mathbf{P}c_l \tag{4}$$

To combine all weights for simplicity, we rewrite (4) as the following matrix form:

$$\mathbf{W} = \mathbf{PC} \tag{5}$$

To share knowledge across multiple landmarks, the goal of the sharable detectors aims to learn $\mathbf{P}$ and $\mathbf{C}$ instead of directly learning $\mathbf{W}$. To achieve this, we apply the ℓ_1 norm to force more elements in $\mathbf{P}$ and $\mathbf{C}$ to be 0. In this way, the latent tasks only respond to a few particular feature patterns and the remaining parameters are expected to represent the semantic facial parts. Therefore, the redundancy information is removed by employing the proposed sharable detectors.

3) Formulation: Based on the above discussions, we formulate the proposed goals as minimizing the following objective function:

$$\begin{aligned}\min_{\{\mathbf{P},\mathbf{C},\mathbf{H},\mathbf{Q}\}} J &= J_1(\mathbf{P},\mathbf{C},\mathbf{H},\mathbf{Q}) + J_2(\mathbf{P},\mathbf{C}) \\ &= \sum_j^G \sum_i^N \frac{1}{2}\left\|\mathbf{S}_i^* - \mathbf{S}_i^0 - \mathbf{Q}\left[\left(\mathbf{P}c_j\right)^T \mathbf{\Phi}_i\right]\right\|_2^2 \\ &\quad + \left(\gamma\,\|\mathbf{C}\|_1 + \beta\,\|\mathbf{P}\|_1 + \mu\,\|\mathbf{P}\|_F^2\right)\end{aligned} \tag{6}$$

where $\mathbf{Q}[\cdot]$ denotes the model parameters of two-layer feed-forward neural networks, G is the number of the latent sharable detectors, and γ, β and μ are used to balance the triple regularization terms compared with regression loss.

$\|\cdot\|_1$ denotes the entry-wise ℓ_1 norm of the sparsity of the matrices $\mathbf{P}$ and $\mathbf{C}$, which performs on the latent tasks where some coefficients of the parameters $\mathbf{P}$ and $\mathbf{C}$ are enforced to be 0. $\|\cdot\|_F^2$ denotes the penalty of the Frobenius norm on $\mathbf{P}$ to Preduce the ℓ_2 norm and avoid overfitting.

For (6), $\mathbf{\Phi}_i$ in term J_1 learns the discriminative and structural features, where the correlation of neighbouring landmarks and semantic information from nearer landmarks is explicitly exploited. $\mathbf{P}$ and $\mathbf{C}$ in term J_1 learns a subset of sharable detectors, which reduces the redundancy information caused by the overlapped local patches. $\mathbf{Q}$ in term J_1 employs a nonlinear regression function by using a feed-forward neural network, which infers the occluded facial part from the non-occluded facial part due to the global shape detector. The sparsity regularization on both $\mathbf{P}$ and $\mathbf{C}$ in J_2 enforces the latent tasks to be selectively shared by each coefficient of the global regression. The Frobenius norm on $\mathbf{P}$ in J_2 controls the model complexity and prevents the learned model from overfitting.

4) Optimization: To optimize (6), we employ the standard back-propagation [40] method for the model parameters $\mathbf{H}$ and $\mathbf{Q}$ the with the learning rate λ (ignoring the offset term):

$$\mathbf{H} \leftarrow \mathbf{H} - \lambda \frac{\partial J}{\partial \mathbf{H}}; \quad \mathbf{Q} \leftarrow \mathbf{Q} - \lambda \frac{\partial J}{\partial \mathbf{Q}}. \tag{7}$$

Moreover, we optimize $\mathbf{P}$ and $\mathbf{C}$ by the standard back-propagation algorithm with fixed $\mathbf{H}$ and $\mathbf{Q}$. By ignoring $\mathbf{H}$ and $\mathbf{Q}$ for simplicity, we can re-write (6) as follows:

$$\begin{aligned}\min_{\{\mathbf{P},\mathbf{C}\}} J &= J_1(\mathbf{P},\mathbf{C}) + J_2(\mathbf{P},\mathbf{C}) \\ &= \sum_j^G \sum_i^N \frac{1}{2} \left\| \mathbf{S}_i^* - \mathbf{S}_i^0 - \left(\mathbf{P}c_j\right)^T \mathbf{\Phi}_i \right\|_2^2 \\ &\quad + \left(\gamma \|\mathbf{C}\|_1 + \beta \|\mathbf{P}\|_1 + \mu \|\mathbf{P}\|_F^2\right).\end{aligned} \tag{8}$$

Obviously, (8) is not convex in $\mathbf{C}$ and $\mathbf{P}$ simultaneously. Alternatively, we optimize $\mathbf{P}$ by fixed $\mathbf{C}$ and optimize $\mathbf{C}$ by fixed $\mathbf{P}$, iteratively. Since the ℓ_1 norm regularization on $\mathbf{P}$ and $\mathbf{C}$ encounters the non-smooth optimization problem, we introduce the accelerated proximal gradient method (APG) [41] approach to address this. Following [41], we optimize $\mathbf{P}$ and $\mathbf{C}$ in two steps:

Step 1: Optimizing $\mathbf{P}$ by fixed $\mathbf{C}$.

$$\begin{aligned}\min_{\{\mathbf{P}\}} J &= J_1(\mathbf{P}) + J_2(\mathbf{P}) \\ &= \sum_j^G \sum_i^N \left(\frac{1}{2} \left\| \mathbf{S}_i^* - \mathbf{S}_i^0 - \left(\mathbf{P}c_j\right)^T \mathbf{\Phi}_i \right\|_2^2 \right) \\ &\quad + \beta \|\mathbf{P}\|_1 + \mu \|\mathbf{P}\|_F^2\end{aligned} \tag{9}$$

Since $J_1(\mathbf{P})$ is convex but $J_2(\mathbf{P})$ is non-smooth convex, we employ the following update scheme [42] to solve (9):

$$\mathbf{P}^\tau = T_{\frac{\mu}{V}} \left((\hat{\mathbf{P}}^\tau) - \frac{1}{V} \nabla_{\mathbf{P}} f(\hat{\mathbf{P}}^\tau) \right) \tag{10}$$

where T_α is the shrinkage operator defined as:

$$T_\alpha(\|z\| - \alpha)_+ sgn(z) \tag{11}$$

Algorithm 1 DSSD

Input: Training set: $\mathbf{D} = \{(\mathbf{\Phi_i}, \mathbf{S_i})\}$, iterative number Γ, τ, and convergence error ε
Output: Weights: Model parameters $\mathbf{P}, \mathbf{C}, \mathbf{H}, \mathbf{Q}$.
// Parameter Initialization
Initialize $\{\mathbf{P}^{(m)}, \mathbf{C}^{(m)}, \mathbf{H}, \mathbf{Q}\}$.
// Optimization by back-prorogation:
for $ii = 1, 2, \cdots, \Gamma$ **do**
  Randomly select a batch of the face samples $\mathbf{X}$.
  *// Optimize **C** and **P** in two-step iterations:*
  Step 1: Optimize $\mathbf{C}$ with fixed $\mathbf{P}$
  for $\tau_1 = 1, 2, \cdots$ **do**
    $\hat{\mathbf{C}}^{\tau_1} = T_{\frac{\mu}{V}} \left((\mathbf{C}^{\tau_1 - 1}) + \frac{\mathbf{C}^{\tau_1 - 2} - 1}{\mathbf{C}^{\tau_1 - 1}} (\mathbf{C}^{\tau_1 - 1} - \mathbf{C}^{\tau_1 - 2}) \right)$
    $p^{\tau_1} = \frac{1 + \sqrt{1 + 4(p^{\tau_1 - 1})^2}}{2}$
  end
  Step 2: Optimize $\mathbf{P}$ with fixed $\mathbf{C}$
  for $\tau_2 = 1, 2, \cdots$ **do**
    $\hat{\mathbf{P}}^{\tau_2} = T_{\frac{\mu}{V}} \left((\mathbf{P}^{\tau_2 - 1}) + \frac{\mathbf{P}^{\tau_2 - 2} - 1}{\mathbf{P}^{\tau_2 - 1}} (\mathbf{P}^{\tau_2 - 1} - \mathbf{P}^{\tau_2 - 2}) \right)$
    $p^{\tau_2} = \frac{1 + \sqrt{1 + 4(p^{\tau_2 - 1})^2}}{2}$
  end
  // τ_1 and τ_2 stop until converged.
  *// Learning Parameters **H** and **Q***
  Perform forward propagation.
  Perform back propagation.
  Update $\{\mathbf{H}^{(m)}, \mathbf{Q}^{(m)}\}$ according to (7).
  Calculate J_{ii} using (6).
  If $ii > 1$ and $|J_{ii} - J_{ii-1}| < \varepsilon$, Converged.
end

where V denotes the Lipschitz constant, which is computed by the back-tracking line searching method. APG performs the gradient of smooth part $\nabla_{\mathbf{P}} f(\hat{\mathbf{P}}^\tau)$ given the search point $\hat{\mathbf{P}}^\tau$ for the τth iteration, and APG performs the linear combination of two previous points for the next iteration. To be specific, given two previous points $\mathbf{P}^{\tau-1}$ and $\mathbf{P}^{\tau-2}$, the search point $\hat{\mathbf{P}}^{\tau-1}$ at the τth iteration is $\mathbf{P}_{\tau-1} + \left(\frac{p_{\tau-2}-1}{p_{\tau-1}}\right)\left(\mathbf{P}^{\tau-1} - \mathbf{P}^{\tau-2}\right)$. p is initialized as 1 and updated by a fast iterative shrinkage-thresholding algorithm [42], which is computed as $p_\tau = \frac{1+\sqrt{1+4(p_{\tau-1})^2}}{2}$.

Step 2: Optimizing $\mathbf{C}$ by fixed $\mathbf{P}$.

$$\begin{aligned}\min_{\{\mathbf{C}\}} J &= J_1(\mathbf{C}) + J_2(\mathbf{C}) \\ &= \sum_j^G \sum_i^N \left(\frac{1}{2} \left\| \mathbf{S}_i^* - \mathbf{S}_i^0 - \left(\mathbf{P}c_j\right)^T \mathbf{\Phi}_i \right\|_2^2 \right) \\ &\quad + \gamma \|\mathbf{C}\|_1\end{aligned} \tag{12}$$

where $\mathbf{C}^m$ is obtained by the gradient of the smooth part $\nabla_{\mathbf{C}} f(\hat{\mathbf{C}}^m)$ given the search point $\hat{\mathbf{C}}^m$ as

$$\mathbf{C}^\tau = T_{\frac{\mu}{V}} \left((\hat{\mathbf{C}}^\tau) - \frac{1}{V} \nabla_{\mathbf{C}} f(\hat{\mathbf{C}}^\tau) \right). \tag{13}$$

Algorithm 1 summarizes the detailed optimization procedure of DSSD.

B. Recurrent-DSSD

Existing methods [9], [10], [13], [15], [16] have employed the coarse-to-fine manner to refine the predicted shape progressively. Differently, we extend our DSSD to a recurrent architecture (R-DSSD), which shares the parameters across different stages by exploiting the feature representations from multi-scale perspective. To be specific, our R-DSSD model consists of five stages: two coarse stages and three refinement stages. For the coarse stage, we apply the random difference feature (RDF) [9], which shows superior efficiency on face alignment. For the refinements stages, we extract the discriminative feature representations to perform the shape refinements. As a result, the recurrent architecture significantly reduces the number of parameters and enhances the generalization ability of our method.

Let R denote the shape regressor, which specifically predicts the landmark residual utilizing the face image $\mathbf{X}$ based on the initial shape $\mathbf{S}^0$ up to the last stage (ignore the index i for the ith face image).

$$R\left(\mathbf{X}\right) = \mathbf{S}^0 + \sum_{t=1}^{T} R^t\left(\mathbf{X}, \mathbf{S}^{t-1}\right) \tag{14}$$

where the whole shape is updated incrementally

$$\mathbf{S}^t = \mathbf{S}^0 + R^t\left(\mathbf{X}, \mathbf{S}^{t-1}\right) \tag{15}$$

for the stage $t = 1, 2, \ldots, T$. The formulation to be optimized can be summarized as follows:

$$J = J^1\left(R_c, \Phi_{rdf}\right) + J^2\left(R_f, \Phi_{cnn}\right) + J^3\left(R_f, \Phi_{cnn}\right) \tag{16}$$

where R_c denotes the coarse stage with the random difference-based feature Φ_{rdf} and R_f denotes the refinements stage with the deep CNN feature Φ_{cnn}.

To jointly learn the model parameters in (16), we optimize $\mathbf{H}$ and $\mathbf{Q}$ by using back-propagation with fixed $\mathbf{P}$ and $\mathbf{C}$. Based on the optimization procedure of DSSD described in Algorithm 1, the parameters $\mathbf{P}$ and $\mathbf{C}$ can be optimized by APG [41]. To perform the joint learning via back-propagation, the derivatives of the shape w.r.t. the loss and the derivatives of the image w.r.t. the loss are computed for both the coarse and refinements stages, respectively. Take one landmark for simplicity, the derivatives of landmark $\mathbf{p}$ w.r.t. loss are calculated as:

$$\frac{\partial J}{\partial \mathbf{p}} = \frac{\partial \mathbf{X}}{\partial \mathbf{p}} \frac{\partial J}{\partial \mathbf{X}} \tag{17}$$

The scalar random pixel difference feature is computed as follows:

$$\epsilon = \Phi(\mathbf{p}, \mathbf{X}) = \mathbf{X}(\mathbf{p} + d_1) - \mathbf{X}(\mathbf{p} + d_2), \tag{18}$$

where $I(\mathbf{p})$ denotes the pixel value located at the landmark $\mathbf{p}$.

Furthermore, we have the derivatives w.r.t. landmark $\mathbf{p}$ as:

$$\frac{\partial \epsilon}{\partial \mathbf{p}} = \nabla\mathbf{X}(\mathbf{p} + d_1) - \nabla\mathbf{X}(\mathbf{p} + d_2). \tag{19}$$

where $I(\mathbf{p})$ denotes the 2D image gradient vector at the point $\mathbf{p}$.

Algorithm 2 R-DSSD

Input: Training set $\mathbf{X}$, stage number T (Typically, T is 3.), iterative number Γ, and convergence error ε.
Output: Weights: $\{\mathbf{H}^{(m)}, \mathbf{Q}^{(m)}\}$.
Initialize stage-wise $\{\mathbf{H}^{(m)}, \mathbf{Q}^{(m)}\}$.
// Optimization by back-prorogation:
for $ii = 1, 2, \cdots, \Gamma$ **do**
  Randomly select a batch of $\mathbf{X}$.
  // Forward propagation
  for $1, 2, \cdots, T$ **do**
    Perform forward propagation according to (14).
  end
  // Computing gradient
  for $T, T-1, \cdots, 1$ **do**
    Obtain gradients by back-propagation on $\mathbf{H}$ and $\mathbf{Q}$. Meanwhile, we perform **Algorithm 1** to optimize $\mathbf{P}$ and $\mathbf{C}$ by fixing $\mathbf{H}$ and $\mathbf{Q}$.
  end
  // Back propagation
  Perform summation for stage-wise gradients.
  for $m = 1, 2, \cdots, T$ **do**
    Update $\{\mathbf{H}^{(m)}, \mathbf{Q}^{(m)}\}$ by (7) by averaging gradients
  end
  Calculate $J_i i$ using (6).
  If $ii > 1$ and $|J_{ii} - J_{ii-1}| < \varepsilon$, go to **Return**.
end
Return: $\{\mathbf{H}^{(m)}, \mathbf{Q}^{(m)}\}$.

Likewise, we have the derivatives w.r.t. I as:

$$\frac{\partial \epsilon}{\partial \mathbf{X}} = \delta(\mathbf{p} + d_1) - \delta(\mathbf{p} + d_2) \tag{20}$$

where δ denotes the pulse response $0, 1$, which takes values 1 at each landmarks $\mathbf{p}$ and 0 at other positions.

The derivatives of landmark $\mathbf{p}$ w.r.t. loss are computed as:

$$\frac{\partial \mathbf{X}}{\partial \mathbf{p}} = \nabla(\mathbf{X}(\mathbf{p} + d)) \tag{21}$$

where ∇ is the gradient-image w.r.t the cropped image patch.

Since the derivatives of the pose-image are not strictly differentiable for 2D images, the value is approximated by the gradient of the image. Specifically, the $\nabla(\mathbf{X}(\mathbf{p} + d))$ is calculated by the Sobel operator in size of $d \times d$ which is convolved on the image patches. The final result is summed up by all of the gradients of each landmark. The derivatives of the image w.r.t. the loss are:

$$\frac{\partial J}{\partial \mathbf{X}} = \frac{\partial \epsilon}{\partial \mathbf{X}} \frac{\partial J}{\partial \epsilon} \tag{22}$$

where $\frac{\partial J}{\partial \mathbf{X}}$ is useless when the output is propagated to the image. However, the derivatives of the image w.r.t. the loss can help the visualization of the back-propagation procedure.

Algorithm 2 summarizes the detailed optimization procedure of R-DSSD.

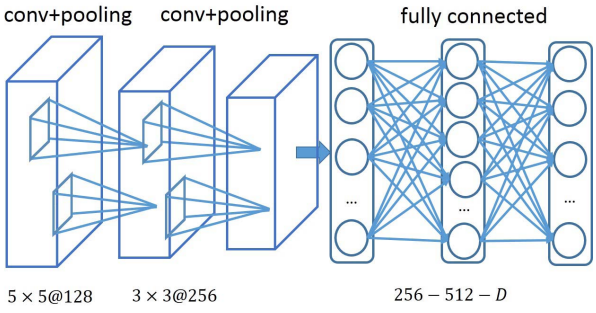


Fig. 4. Structure specification of the deep CNN feature extractor.

IV. Implementation Details

In this section, we describe the implementation details including data preparation and network architecture, respectively.

A. Data Preparation

For the face images to be aligned, we used the bounding box downloaded from the IBUG website and enlarged the downloaded bounding boxes by 30% on both the width and the height. We resized all face images into 200×200 first and subscaled them into 100×100 and 50×50, respectively. For each ground-truth, we normalized it into the range of $[0, 1]$. To better improve the performance, we applied the robust shape initialization (RSI) [36] approach to seek the specific initial shape from the shape space via clustering. Specifically, we firstly constructed 500 shape centers via the k-means method on the training set, which covered a wide range of the shapes including various poses, expressions and aspect ratios. During the learning procedure, we aligned the ground-truth to the cluster center as the initial shape. To enhance the generation of our deep model, we performed the data augmentation strategy [9] to generate more samples in the training procedure.

B. Network Architecture

As illustrated in Fig. 4, we fed the cropped image patch in size of 30×30 as the input to the CNN feature extractor. Hence, D for each shape-index patch was specified to 900. In the first layer, we convolved it with 128 different 5×5 filters. Each convolutional layer was followed by a nonlinear ReLU function [38]. The obtained feature maps were downsampled with a 2×2 max pooling operation. Similar operations were repeated in the second layer. Following the convolution layers, we applied 256-512 fully connections with a nonlinear *tanh* function. Lastly, we employed 20-dim vector as the feature for each local patch. For sharable detectors, we specified G to 20 and K to 5 by the experimental cross validations. We assigned the values of the weight decay, moment parameter and learning rate empirically to 0.0001, 0.9, and 0.01, respectively. The parameters of the designed networks were initialized by the normalized random initialization method [43] as follows: as follows:

$$\mathbf{W}^{(m)} \sim \mathrm{U}\left[-\frac{\sqrt{6}}{\sqrt{r^{(m)}+r^{(m-1)}}}, \frac{\sqrt{6}}{\sqrt{r^{(m)}+r^{(m-1)}}}\right] \quad (23)$$

where the bias $b^{(m)}$ of the mth layer was set as 0, and $r^{(m)}$ was the size of the mth layer.

To better initialize the model parameters employed in DSSD, we first trained a linear SVM regressor to obtain the global parameter $\mathbf{H}$, then we computed the singular value decomposition (SVD) for $\mathbf{H}$ to obtain $\mathbf{H} = \mathbf{ZPV}^{\mathbf{T}}$. The initialization of the matrix $\mathbf{P}$ was given by the first G columns of $\mathbf{Z}$. In terms of the balanced parameters γ, μ and β in (6), we conducted a cross validation to perform the parameter selection. The whole training procedure of R-DSSD converged in 15 iterations.

V. Experiments

In this section, we conducted facial alignment experiments to show the effectiveness of the proposed methods. The followings describe the details of experimental results and analysis.

A. Datasets

1) 300-W [22]: This dataset consists of various datasets for face alignment, including the LFPW [46], HELEN [47], AFW [27], XM2VTS [48] and IBUG [49] datasets with annotated 68 landmarks. For fair comparisons, we trained our model with the LFPW training set, the HELEN training set, the AFW dataset and tested it on the LFPW testing set, the HELEN testing set and the IBUG dataset, respectively. In addition, we investigated our approaches on the testing samples from the LFPW and HELEN datasets as the common set and the 135-image IBUG dataset as the challenging set, and the union of them as the full set (689 images in all).

2) COFW [44]: The Caltech Occluded Face in the Wild (COFW) dataset consists of 1345 training face images and 507 testing face images, which were collected from the Internet. All face images are annotated with 29 landmarks together with the visibility/invisibility information. We conducted experiments and evaluated our methods only on its testing set. Note that our model was trained on the training samples from the 300-W dataset, without using any training images from the COFW dataset.

B. Evaluation Protocols

Following the settings employed in [12], we utilized the normalized root mean squared error (NRMSE), which was normalized by the pupil distance to measure the error between the predicted positions and the ground-truths. Hence, the point-to-point NRMSE distance for each face image is computed as follows:

$$\mathrm{NRMSE} = \frac{\sum_{i=1}^{L} \left\| \mathbf{p}_i^{pred} - \mathbf{p}_i^{gt} \right\|_2}{\left\| \mathbf{p}_{leye} - \mathbf{p}_{reye} \right\|_2} \quad (24)$$

where $\mathbf{p}_i^{pred}$ and $\mathbf{p}_i^{gt}$ denote the ith landmark coordinates of the predicted and ground-truth facial landmark positions accordingly. $\mathbf{p}_{leye}$ and $\mathbf{p}_{reye}$ denote the pupil locations of the left eye and the right eye, respectively. Finally, we averaged the NRMSEs for all testing face samples in our experiments as the averaged error comparisons for evaluation.

TABLE I

COMPARISONS OF THE AVERAGED ERRORS WITH THE STATE-OF-THE-ART APPROACHES (THE BEST PERFORMANCE ARE QUOTED IN THE BOLD TYPE AND THE TOP-3 PERFORMANCE IN THE ITALIC TYPE). FOR MOST METHODS, WE DIRECTLY CROPPED THE REPORTED RESULTS FROM THE RELATED LITERATURES OR EVALUATED BASED ON THE RELEASED CODES. OUR MODEL ACHIEVES VERY COMPETITIVE PERFORMANCE COMPARED WITH THE STATE-OF-THE-ART FACE ALIGNMENT METHODS

Method	LFPW 68-pts	HELEN 68-pts	HELEN 192-pts	Common Set 68-pts	Challenging Set 68-pts	Full Set 68-pts
FPLL [27]	8.29	8.16	-	8.22	18.33	10.20
DRMF [23]	6.57	6.70	-	6.65	19.79	9.22
RCPR [46]	6.56	5.93	6.50	6.18	17.26	8.35
GN-DPM [24]	5.92	5.69	-	5.78	-	-
SDM [8]	5.67	5.50	5.85	5.57	15.40	7.50
CFAN [16]	5.44	5.53	-	5.50	-	-
ERT [11]	-	-	4.90	-	-	6.40
BPCPR [19]	-	-	-	5.24	16.56	7.46
ESR [9]	-	-	5.70	5.28	17.00	7.58
LBF [10]	-	-	5.41	4.95	11.98	6.32
LBF fast [10]	-	-	5.80	5.38	15.50	7.37
Deep Reg [15]	-	-	-	4.51	13.80	6.31
CFSS [12]	4.87	*4.63*	*4.74*	*4.73*	*9.98*	*5.76*
CFSS Practical [12]	4.90	4.72	4.84	4.73	10.92	5.99
TCDCN [47]	*4.57*	4.60	*4.63*	4.80	8.60	*5.54*
DCRFA [38]	*4.57*	*4.25*	-	*4.19*	**8.42**	**5.02**
R-DSSD*	4.77	4.31	4.95	4.57	10.86	5.91
R-DSSD	**4.52**	**4.08**	**4.62**	**4.16**	*9.20*	5.59

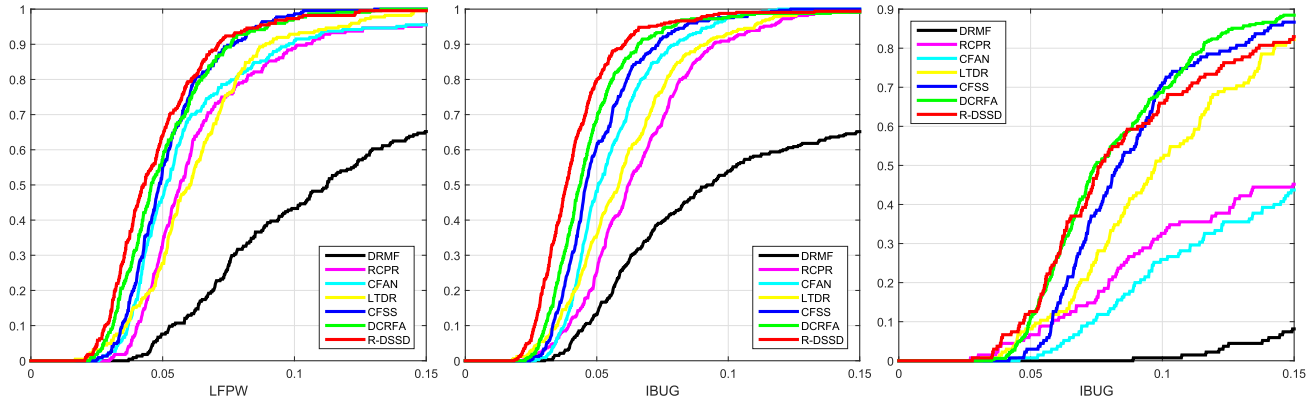


Fig. 5. CED curves of face alignment approaches tested on the LFPW, HELEN and IBUG datasets, respectively, where 68 landmarks were employed for evaluation.

We also applied the cumulative error distribution (CED) curves of NRMSE to quantitatively evaluate the performance of our methods with previous face alignment methods. Let e denote the averaged error normalized by the pupil distance and CED at the error l is computed as follows:

$$\text{CED} = \frac{N_{e \leq l}}{n} \tag{25}$$

where $N_{e \leq l}$ is the number of images on which the error l is no less than e.

C. Results and Analysis

1) Comparisons With State-of-the-Art Methods: We compared our methods with fifteen state-of-the-art face alignment methods including FPLL [27], DRMF [23], RCPR [44], SDM [8], ESR [9], GN-DPM [24], ESR [9], LBF [10], ERT [11], CFSS [12], CFAN [16], BPCPR [19], LTDR [52], TCDCN [45], DFSM [20], Deep Reg [15] and DCRFA [36]. The standard implementations of other compared methods were provided by the original authors except DCRFA and DSFM because their codes have not been publicly released. We carefully implemented DCRFA by following the settings in [36] and we directly cropped the results for DFSM from the original paper. As for our methods, we deployed two architectures: R-DSSD* and R-DSSD. Specifically, R-DSSD* denotes the practical architecture with one RDF and two CNNs, and R-DSSD denotes the architecture including two RDF and three CNNs, which further improve the alignment performance in accuracy. Table I shows the NRMSEs of different face alignment approaches, which implicates that our methods achieve very competitive performance compared with the state-of-the-art methods. Fig. 5 shows the CED curves of several face alignment approaches tested on various datasets. According to the results, we see that our methods obtain the best performance on the LFPW and HELEN datasets, and even achieve very competitive results on the IBUG dataset. Moreover, we illustrated the aligned results in Fig. 6. From these example results, we see that our model exhibits superior

Fig. 6. Example results in each row are aligned results for the samples in the LFPW, HELEN and IBUG datasets, respectively, where 68 landmarks were employed for evaluation. According to these results, our model achieves superior performance even the face samples encounter diverse pose variations and varying facial expressions.

capability of handling difficult cases with large facial aspect ratios. In addition, we conducted experiments on the HELEN dataset, where 192 landmarks were employed for evaluation. As showed in Table I, we see that our methods achieve comparable performance with TCDCN [45], when denser landmarks were employed in our experiments. The aligned example results for the 192 landmarks are shown in Fig. 7.

2) Comparisons With Existing Deep Learning Based Face Alignment Methods: We also compared our R-DSSD with several recent deep learning based face alignment methods including TCDCN [45], CFAN [16], DCRFA [36] and DFSM [20]. Specifically, TCDCN embedded the tasks of face detection and face alignment with the multi-task learning method by CNN, CFAN mapped the local features to the shape space by utilizing deep auto-encoder networks, DCRFA applied the deconvolution neural networks based cascaded regression method for face alignment and DFSM modeled relationship of both local and holistic shape constraints via the restricted Boltzmann machine networks. Table II shows the failure rate of the percentage of images that were correctly detected, where the average error is 0.05 and 0.1, respectively. We see that our method significantly outperforms the other compared deep learning based methods on the LFPW and HELEN datasets, even our model achieves comparable results on the IBUG dataset.

Fig. 7. Example results in each row are aligned results for the samples in the HELEN dataset, where 192 landmarks were employed for evaluation. From these results, we see that our model obtains accurate landmark localization in such cases that more challenging correlation of neighbouring landmarks of denser 192 annotations occurs than 68 annotations.

TABLE II

COMPARISONS OF PERCENTAGES OF IMAGES WHERE THE AVERAGE ERROR IS 0.05 AND 0.1 OF DIFFERENT DEEP LEARNING BASED FACE ALIGNMENT METHODS. FOR CFAN [16] AND TCDCN [45], WE INVESTIGATED THE EVALUATIONS BASED ON THE RELEASED CODES. FOR DCRFA [36], WE CAREFULLY IMPLEMENTED THE METHOD BY FOLLOWING THE SETTING IN [36]. SINCE THE CODE OF DFSM [20] HAS NOT BEEN RELEASED, WE CROPPED THE RESULTS DIRECTLY FROM THE ORIGINAL PAPER. FROM THESE RESULTS, WE SEE THAT OUR R-DSSD OBTAINS PROMISING RESULTS COMPARED WITH THE STATE-OF-THE-ART METHODS ON THE LFPW AND HELEN DATASETS

Dataset	**Methods**	**Architectures**	**NRMSE** (68-pts)	**CED**$_{(\text{NRMSE}\leq 0.05)}$	**CED**$_{(\text{NRMSE}\leq 0.1)}$
LFPW	CFAN [16]	DAE	5.44	29.02%	90.18%
	DFSM [20]	RBM	-	-	96.39%
	TCDCN [47]	CNN-MTL	4.57	50.89%	92.86%
	DCRFA [38]	DeConv	4.57	60.25%	93.99%
	R-DSSD	Recurrent	**4.52**	**64.13%**	**97.62%**
HELEN	CFAN [16]	DAE	5.53	49.39%	97.27%
	DFSM [20]	RBM	-	-	89.91%
	TCDCN [47]	CNN-MTL	4.60	79.94%	97.58%
	DCRFA [38]	DeConv	4.25	78.23%	96.52%
	R-DSSD	Recurrent	**4.08**	**80.24%**	**98.78%**

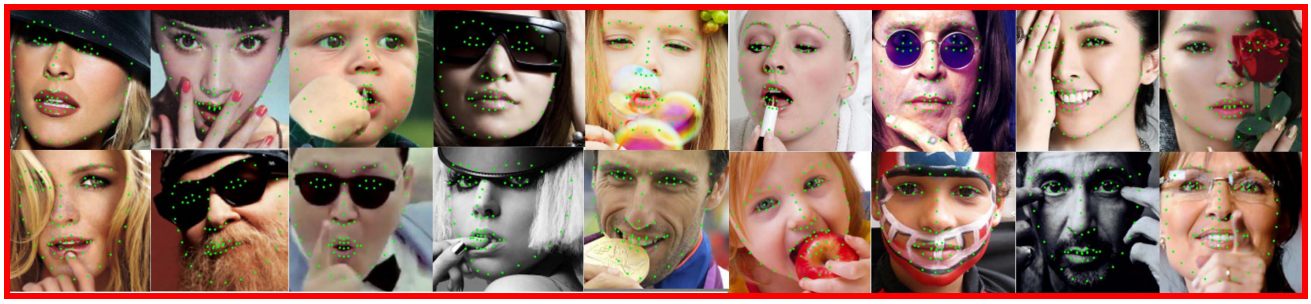


Fig. 8. Example results in each row are aligned results for the samples in the COFW dataset, where 68 landmarks were employed for evaluation. Our model achieves robustness to largely variations of face images caused by diverse partial occlusions, which benefits from the employed nonlinear global regression mapping.

3) Evaluation on COFW [44]: To investigate the effectiveness of our methods versus various occlusions on the COFW dataset, where the facial images are occluded by the invisible parts. Table III shows the averaged errors and failure rates [44]. From these results, we see that our method achieves better performance than that of the state-of-the-art methods. Some sample alignment results on the COFW dataset are shown in Fig. 8. We see that our method accurately detects the occluded landmarks even with the heavy occlusions. This is because our model achieves the robustness to the face images versus partial occlusions by exploiting both the local and global information in the learned features, which verifies the robustness of our framework on the occluded face images.

4) Analysis of the Sharable Detectors: To investigate the importance of the sharable detectors, we conducted experiments and compared the performance of our methods with and without the sharable detectors. In terms of without the sharable detectors, we improved CFAN by employing CNN architecture instead of auto-encoder networks as the baseline method, because CFAN [16] were trained by the joint displacement of all landmarks without explicitly considering the correlated relationship among neighbouring landmarks. The parameters were configured according to [16] and [19]. Table IV tabulates the averaged results. From these results, we see that the sharable detectors significantly improve the landmark detection accuracy, especially in the challenging

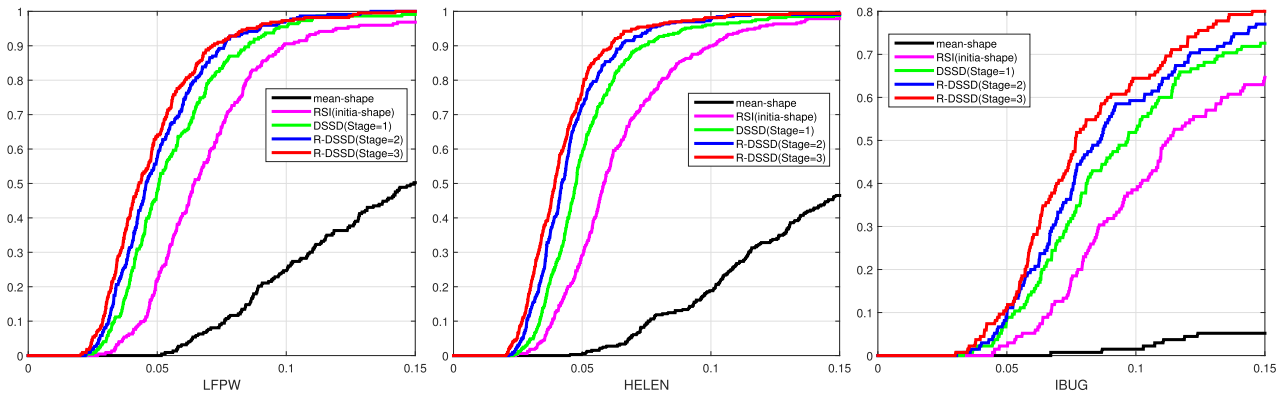


Fig. 9. CED curves of different stages in our R-DSSD tested on the LFPW, HELEN and IBUG datasets, respectively, where 68 annotated landmarks were employed for evaluation.

TABLE III

COMPARISON OF THE AVERAGED ERRORS AND THE FAILURE RATES ON THE COFW DATASET. OUR MODEL ACHIEVES ROBUSTNESS TO VARYING PARTIAL OCCLUSIONS

Method	NRMSE	Failure Rate
FPLL [27]	8.79	38.46%
ESR [9]	11.20	36.00%
RCPR [46]	8.50	20.00%
HPM [52]	7.46	13.24%
RPP [53]	7.52	16.20%
SDM [8]	8.77	24.32%
CFAN [16]	8.38	19.14%
TCDCN [47]	8.05	15.31%
DCRFA [38]	6.54	9.02%
R-DSSD	**6.17**	**8.23%**

TABLE IV

COMPARISONS OF AVERAGED ERRORS WITH AND WITHOUT SHARABLE DETECTORS ON THE LFPW, HELEN AND IBUG DATASETS, RESPECTIVELY, WHERE 68 ANNOTATED LANDMARKS WERE EMPLOYED FOR EVALUATION

	LFPW	HELEN	IBUG
R-DSSD without *sharable detectors*	4.99	5.14	11.32
R-DSSD with *sharable detectors*	**4.52**	**4.08**	**9.20**

TABLE V

COMPARISON OF AVERAGED ERRORS WITHOUT $\|\mathbf{P}\|_1$, WITHOUT $\|\mathbf{C}\|_1$ AND DSSD ON 300-W DATASET

	Common Set	Challenging Set	Full Set
without $\|\mathbf{P}\|_1$	5.69	13.97	6.98
without $\|\mathbf{C}\|_1$	5.38	13.76	6.55
DSSD	**5.21**	**12.85**	**6.13**

IBUG dataset, which were captured under the large variances of facial expressions and aspect ratios. In addition, we also evaluated the effects of the regularization terms $\|\mathbf{P}\|_1$ and $\|\mathbf{C}\|_1$. Table V shows the performance of DSSD of both penalties. According to the results, our method degrades significantly when the ℓ_1 norm constraints are removed.

5) Analysis of the Cascaded R-DSSD: To evaluate the effectiveness of the cascaded architecture, we first implemented the robust shape initialization to provide an initial shape. Moreover, we examined the performance of R-DSSD with different network depths. Fig. 9 shows the CED curves of R-DSSD on various datasets, where different stages were employed for evaluation. We see that our R-DSSD with multiple stages consistently outperforms R-DSSD with single stage because the complementary information can be exploited from multi-scale perspectives for accurate shape refinements. Moreover, our R-DSSD with single stage achieves better performance than that of the predicted initial shape, which shows the effectiveness of our carefully designed deep architectures. This is because the shape has been refined in a fixed size of local patch on different scales of images in a coarse-to-fine way.

TABLE VI

COMPARISON OF AVERAGED ERRORS OF SEQUENTIAL LEARNING AND JOINT LEARNING ON 300-W DATASET

	Common Set	Challenging Set	Full Set
Sequential	5.16	12.28	6.33
Joint	4.98	11.56	6.19
Recurrent	**4.41**	**11.17**	**5.82**

6) Comparisons With Different Learning Strategies: We investigated the effectiveness of the proposed recurrent architecture. In this experiment, we compared of our method with the sequential learning and the joint learning strategies. Specifically, the sequential learning method separately learns the parameters for each stage and then directly combines them together, while the joint learning method jointly trains the model parameters in an end-to-end manner. Table VI tabulates the averaged errors of our recurrent learning with other learning strategies. From these results, we see that the recurrent architecture obtains better performance than that of other learning strategies, which exhibits the effectiveness of the designed recurrent manner.

7) Performance Effects for Different Parameters: We conducted experiments to analyze different factors of our methods. First, we set $G = \{5, 10, 20, 50, 68\}$ to the number of the latent tasks employed in the sharable detectors. Table VII tabulates the results. According to these results, we see that our method achieves the best performance when G was set

TABLE VII

AVERAGING ERRORS DENOTED BY VARIOUS G EMPLOYED IN THE SHARABLE DETECTORS TESTED ON THE 300-W FULLSET

G	5	10	**20**	50	68
NRMSE	8.93	8.77	**7.09**	7.45	7.99

TABLE VIII

AVERAGING ERRORS DENOTED BY VARIOUS K RELATED TO DSSD (SINGE STAGE) TESTED ON 300-W FULLSET

K	1	3	**5**	12	68
NRMSE	6.55	5.81	**5.59**	6.87	7.92

to 20. We have also evaluated the variances of K on the 300-W fullset, which was employed in the structural feature learning approach. Specifically, we set $K = \{1, 3, 5, 12, 68\}$ to the the number of neighbours for each landmark. Note that we created the baseline method by setting K to 1, which means the sharable detectors were directly fed with the extracted local features without any feature selection approach. Table VIII shows the averaged errors, respectively. According to the results, we see that our method obtains the best performance when K was set to 5, which also shows the effectiveness of the structural learning. Moreover, the performance deteriorates when K grows to $\{12, 68\}$. This is because more noise occurs during the landmark localization with a large range of locality.

8) Computational Time: Our approach was implemented on the Matlab platform with the DAGNN module of the MatConvnet [53] deep learning toolbox. Our model with the practical architecture R-DSSD* runs at around 40 frames per second (FPS) with the Intel(R) Core(TM) i5-6500 CPU @ 3.20GHz, which satisfies the real-time requirements.

VI. CONCLUSIONS AND FUTURE WORK

In this paper, we have proposed a deep feature learning method for face alignment, dubbed deep sharable and structural detectors (DSSD), where the semantic information and correlation of neighbouring landmarks are exploited for facial landmark localization. To further improve the alignment performance, we have extended our DSSD to a recurrent DSSD (R-DSSD) architecture, which integrates the learned pose-informative feature representations with multi-scale information and efficiently controls the capacity of the cascaded deep architecture. Experimental results on both the public benchmark datasets including the 300-W dataset and the COFW dataset verify the effectiveness of the proposed methods compared with the state-of-the-art face alignment methods.

There are two interesting directions for our further work:

1 Our proposed R-DSSD is built based on the recurrent architecture and it is interesting to apply it to video-based face alignment by using the feedback architecture to further improve the effectiveness.

2 Our proposed framework works well for the near-front facial images. How to apply it to detect the facial landmarks for the facial images which were captured in unconstrained conditions is the interesting future work.

REFERENCES

[1] Y. Taigman, M. Yang, M. Ranzato, and L. Wolf, "DeepFace: Closing the gap to human-level performance in face verification," in *Proc. CVPR*, Jun. 2014, pp. 1701–1708.

[2] J. Hu, J. Lu, and Y.-P. Tan, "Discriminative deep metric learning for face verification in the wild," in *Proc. CVPR*, Jun. 2014, pp. 1875–1882.

[3] Z. Huang, X. Zhao, S. Shan, R. Wang, and X. Chen, "Coupling alignments with recognition for still-to-video face recognition," in *Proc. ICCV*, 2013, pp. 3296–3303.

[4] N. Kumar, P. N. Belhumeur, and S. K. Nayar, "FaceTracer: A search engine for large collections of images with faces," in *Proc. ECCV*, 2008, pp. 340–353.

[5] T. F. Cootes, C. J. Taylor, D. H. Cooper, and J. Graham, "Active shape models-their training and application," *Comput. Vis. Image Understand.*, vol. 61, no. 1, pp. 38–59, Jan. 1995.

[6] T. F. Cootes, G. J. Edwards, and C. J. Taylor, "Active appearance models," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 23, no. 6, pp. 681–685, Jun. 2001.

[7] T. F. Cootes, M. C. Ionita, C. Lindner, and P. Sauer, "Robust and accurate shape model fitting using random forest regression voting," in *Proc. ECCV*, 2012, pp. 278–291.

[8] X. Xiong and F. D. la Torre, "Supervised descent method and its applications to face alignment," in *Proc. CVPR*, Jun. 2013, pp. 532–539.

[9] X. Cao, Y. Wei, F. Wen, and J. Sun, "Face alignment by explicit shape regression," in *Proc. CVPR*, 2012, pp. 2887–2894.

[10] S. Ren, X. Cao, Y. Wei, and J. Sun, "Face alignment at 3000 FPS via regressing local binary features," in *Proc. CVPR*, 2014, pp. 1685–1692.

[11] V. Kazemi and J. Sullivan, "One millisecond face alignment with an ensemble of regression trees," in *Proc. CVPR*, Jun. 2014, pp. 1867–1874.

[12] S. Zhu, C. Li, C. C. Loy, and X. Tang, "Face alignment by coarse-to-fine shape searching," in *Proc. CVPR*, Jun. 2015, pp. 4998–5006.

[13] Y. Sun, X. Wang, and X. Tang, "Deep convolutional network cascade for facial point detection," in *Proc. CVPR*, 2013, pp. 3476–3483.

[14] E. Zhou, H. Fan, Z. Cao, Y. Jiang, and Q. Yin, "Extensive facial landmark localization with coarse-to-fine convolutional network cascade," in *Proc. ICCVW*, Dec. 2013, pp. 386–391.

[15] B. Shi, X. Bai, W. Liu, and J. Wang. (2014). "Deep regression for face alignment." [Online]. Available: https://arxiv.org/abs/1409.5230

[16] J. Zhang, S. Shan, M. Kan, and X. Chen, "Coarse-to-fine auto-encoder networks (CFAN) for real-time face alignment," in *Proc. ECCV*, 2014, pp. 1–16.

[17] Z. Zhang, P. Luo, C. C. Loy, and X. Tang, "Facial landmark detection by deep multi-task learning," in *Proc. ECCV*, 2014, pp. 94–108.

[18] J. J. Tompson, A. Jain, Y. Lecun, and C. Bregler, "Joint training of a convolutional network and a graphical model for human pose estimation," in *Proc. NIPS*, 2014, pp. 1799–1807.

[19] P. Sun, J. K. Min, and G. Xiong. (2015). "Globally tuned cascade pose regression via back propagation with application in 2D face pose estimation and heart segmentation in 3D CT images." [Online]. Available: https://arxiv.org/abs/1503.08843

[20] Y. Wu and Q. Ji, "Discriminative deep face shape model for facial point detection," *Int. J. Comput. Vis.*, vol. 113, no. 1, pp. 37–53, May 2015.

[21] C. N. Duong, K. Luu, K. G. Quach, and T. D. Bui, "Beyond principal components: Deep Boltzmann machines for face modeling," in *Proc. CVPR*, 2015, pp. 4786–4794.

[22] *300 Faces in-the-Wild Challenge*, accessed on Jul. 2013. [Online]. Available: http://ibug.doc.ic.ac.uk/resources/300-W/

[23] A. Asthana, S. Zafeiriou, S. Cheng, and M. Pantic, "Robust discriminative response map fitting with constrained local models," in *Proc. CVPR*, Jun. 2013, pp. 3444–3451.

[24] G. Tzimiropoulos and M. Pantic, "Gauss-Newton deformable part models for face alignment in-the-wild," in *Proc. CVPR*, 2014, pp. 1851–1858.

[25] M. F. Valstar, B. Martínez, X. Binefa, and M. Pantic, "Facial point detection using boosted regression and graph models," in *Proc. CVPR*, Jun. 2010, pp. 2729–2736.

[26] G. Tzimiropoulos, "Project-Out Cascaded Regression with an application to face alignment," in *Proc. CVPR*, Jun. 2015, pp. 3659–3667.

[27] X. Zhu and D. Ramanan, "Face detection, pose estimation, and landmark localization in the wild," in *Proc. CVPR*, Jun. 2012, pp. 2879–2886.

[28] D. C. Cireşan, U. Meier, L. M. Gambardella, and J. Schmidhuber, "Deep, big, simple neural nets for handwritten digit recognition," *Neural Comput.*, vol. 22, no. 12, pp. 3207–3220, 2010.

[29] S. Thomas, C. Chatelain, L. Heutte, T. Paquet, and Y. Kessentini, "A deep HMM model for multiple keywords spotting in handwritten documents," *Pattern Anal. Appl.*, vol. 18, no. 4, pp. 1003–1015, Nov. 2015.

[30] D. Erhan, C. Szegedy, A. Toshev, and D. Anguelov, "Scalable object detection using deep neural networks," in *Proc. CVPR*, 2014, pp. 2147–2154.

[31] Y. Zhang, K. Sohn, R. Villegas, G. Pan, and H. Lee, "Improving object detection with deep convolutional networks via Bayesian optimization and structured prediction," in *Proc. CVPR*, Jun. 2015, pp. 249–258.

[32] N. Wang and D.-Y. Yeung, "Learning a deep compact image representation for visual tracking," in *Proc. NIPS*, 2013, pp. 809–817.

[33] P. H. O. Pinheiro and R. Collobert, "Recurrent convolutional neural networks for scene labeling," in *Proc. ICML*, 2014, pp. 82–90.

[34] H. Noh, S. Hong, and B. Han, "Learning deconvolution network for semantic segmentation," in *Proc. ICCV*, 2015, pp. 1520–1528.

[35] G. Trigeorgis, P. Snape, M. A. Nicolaou, E. Antonakos, and S. Zafeiriou, "Mnemonic descent method: A recurrent process applied for end-to-end face alignment," in *Proc. CVPR*, 2016, pp. 4177–4187.

[36] H. Lai, S. Xiao, Z. Cui, Y. Pan, C. Xu, and S. Yan. (2015). "Deep recurrent regression for facial landmark detection." [Online]. Available: https://arxiv.org/abs/1510.09083

[37] Y. Wu, Z. Wang, and Q. Ji, "A hierarchical probabilistic model for facial feature detection," in *Proc. CVPR*, Jun. 2014, pp. 1781–1788.

[38] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Proc. NIPS*, 2012, pp. 1097–1105.

[39] A. Kumar and H. D. Daume, III, "Learning task grouping and overlap in multi-task learning," in *Proc. ICML*, 2012, pp. 1383–1390.

[40] G. E. Hinton, S. Osindero, and Y.-W. Teh, "A fast learning algorithm for deep belief nets," *Neural Comput.*, vol. 18, no. 7, pp. 1527–1554, 2006.

[41] P. Tseng, "On accelerated proximal gradient methods for convex-concave optimization," *J. Optim.*, pp. 1–20, May 2008.

[42] A. Beck and M. Teboulle, "A fast iterative shrinkage-thresholding algorithm for linear inverse problems," *SIAM J. Imag. Sci.*, vol. 2, no. 1, pp. 183–202, 2009.

[43] X. Glorot and Y. Bengio, "Understanding the difficulty of training deep feedforward neural networks," in *Proc. Aistats*, vol. 9. 2010, pp. 249–256.

[44] X. P. Burgos-Artizzu, P. Perona, and P. Dollár, "Robust face landmark estimation under occlusion," in *Proc. ICCV*, Dec. 2013, pp. 1513–1520.

[45] Z. Zhang, P. Luo, C. C. Loy, and X. Tang, "Learning deep representation for face alignment with auxiliary attributes," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 38, no. 5, pp. 918–930, May 2016.

[46] P. N. Belhumeur, D. W. Jacobs, D. J. Kriegman, and N. Kumar, "Localizing parts of faces using a consensus of exemplars," in *Proc. CVPR*, Jun. 2011, pp. 545–552.

[47] V. Le, J. Brandt, Z. Lin, L. D. Bourdev, and T. S. Huang, "Interactive facial feature localization," in *Proc. ECCV*, 2012, pp. 679–692.

[48] K. Messer, J. Matas, J. Kittler, J. Luettin, and G. Maitre, "XM2VTSDB: The extended M2VTS database," in *Proc. AVBPA*, vol. 964. 1999, pp. 965–966.

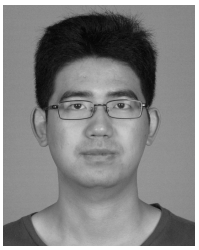
[49] C. Sagonas, G. Tzimiropoulos, S. Zafeiriou, and M. Pantic, "300 faces in-the-wild challenge: The first facial landmark localization challenge," in *Proc. ICCVW*, Dec. 2013, pp. 397–403.

[50] G. Ghiasi and C. C. Fowlkes, "Occlusion coherence: Localizing occluded faces with a hierarchical deformable part model," in *Proc. CVPR*, Jun. 2014, pp. 1899–1906.

[51] H. Yang, X. He, X. Jia, and I. Patras, "Robust face alignment under occlusion via regional predictive power estimation," *IEEE Trans. Image Process.*, vol. 24, no. 8, pp. 2393–2403, Aug. 2015.

[52] Z. Zhang, P. Luo, C. C. Loy, and X. Tang. (2014). "Learning deep representation for face alignment with auxiliary attributes." [Online]. Available: https://arxiv.org/abs/1408.3967

[53] A. Vedaldi and K. Lenc, "MatConvNet: Convolutional neural networks for MATLAB," in *Proc. ACM MM*, 2015, pp. 689–692.



Hao Liu received the B.S. degree in software engineering from Sichuan University, China, in 2011 and the Engineering Master degree in computer technology from the University of Chinese Academy of Sciences, China, in 2014. He is currently pursuing the Ph.D. degree with the Department of Automation, Tsinghua University. His research interests include face alignment, facial age estimation, and deep learning.

Jiwen Lu (S'10–M'11–SM'15) received the B.Eng. degree in mechanical engineering and the M.Eng. degree in electrical engineering from the Xi'an University of Technology, Xi'an, China, and the Ph.D. degree in electrical engineering from Nanyang Technological University, Singapore, in 2003, 2006, and 2012, respectively. From 2011 to 2015, he was a Research Scientist with the Advanced Digital Sciences Center, Singapore. He is currently an Associate Professor with the Department of Automation, Tsinghua University, Beijing, China. His current research interests include computer vision, pattern recognition, and machine learning. He has authored/co-authored over 140 scientific papers in these areas, where 36 of them are the IEEE Transactions papers. He serves/has served as an Associate Editor of Pattern Recognition Letters, Neurocomputing, and the IEEE Access, a Managing Guest Editor of Pattern Recognition and Image and Vision Computing, a Guest Editor of Computer Vision and Image Understanding, and an elected member of the Information Forensics and Security Technical Committee of the IEEE Signal Processing Society. He is/was a Workshop Chair/Special Session Chair/Area Chair for more than ten international conferences. He was a recipient of the National 1000 Young Talents Plan Program in 2015.

Jianjiang Feng received the B.S. and Ph.D. degrees from the School of Telecommunication Engineering, Beijing University of Posts and Telecommunications, China, in 2000 and 2007, respectively. From 2008 to 2009, he was a Post-Doctoral Researcher in the PRIP lab, Michigan State University. He is currently an Associate Professor with the Department of Automation, Tsinghua University, Beijing. He is also an Associate Editor of Image and Vision Computing. His research interests include fingerprint recognition and computer vision.

Jie Zhou (M'01–SM'04) received the B.S. and M.S. degrees from the Department of Mathematics, Nankai University, Tianjin, China, in 1990 and 1992, respectively, and the Ph.D. degree from the Institute of Pattern Recognition and Artificial Intelligence, Huazhong University of Science and Technology, Wuhan, China, in 1995. From 1995 to 1997, he served as a Post-Doctoral Fellow with the Department of Automation, Tsinghua University, Beijing, China. Since 2003, he has been a Full Professor with the Department of Automation, Tsinghua University. He has authored over 100 papers in peer-reviewed journals and conferences. Among them, over 40 papers have been published in top journals and conferences, such as PAMI, TIP, and CVPR. His current research interests include computer vision, pattern recognition, and image processing. He received the National Outstanding Youth Foundation of China Award. He is an Associate Editor of the *International Journal of Robotics and Automation* and two other journals.