Received December 2, 2018, accepted December 19, 2018, date of publication January 1, 2019, date of current version January 29, 2019.

Digital Object Identifier 10.1109/ACCESS.2018.2890573

Facial Component-Landmark Detection With Weakly-Supervised LR-CNN

RUIHENG ZHANG[1,2], **CHENGPO MU**[1], **MIN XU**[2], **LIXIN XU**[1], **AND XIAOFENG XU**[3]

[1]School of Mechatronical Engineering, Beijing Institute of Technology, Beijing 100081, China
[2]GBDTC, Faculty of Engineering and IT, University of Technology Sydney, Ultimo, NSW 2007, Australia
[3]School of Computer Science and Engineering, Nanjing University of Science and Technology, Nanjing 210094, China

Corresponding author: Min Xu (min.xu@uts.edu.au)

This work was supported in part by China Scholarship Council (CSC) and in part by Beijing Institute of Technology Graduate School.

ABSTRACT In this paper, we propose a weakly supervised landmark-region-based convolutional neural network (LR-CNN) framework to detect facial component and landmark simultaneously. Most of the existing course-to-fine facial detectors fail to detect landmark accurately without lots of fully labeled data, which are costly to obtain. We can handle the task with a small amount of finely labeled data. First, deep convolutional generative adversarial networks are utilized to generate training samples with weak labels, as data preparation. Then, through weakly supervised learning, our LR-CNN model can be trained effectively with a small amount of finely labeled data and a large amount of generated weakly labeled data. Notably, our approach can handle the situation when large occlusion areas occur, as we localize visible facial components before predicting corresponding landmarks. Detecting unblocked components first helps us to focus on the informative area, resulting in a better performance. Additionally, to improve the performance of the above tasks, we design two models as follows: 1) we add AnchorAlign in the region proposal networks to accurately localize components and 2) we propose a two-branch model consisting classification branch and regression branch to detect landmark. Extensive evaluations on benchmark datasets indicate that our proposed approach is able to complete the multi-task facial detection and outperforms the state-of-the-art facial component and landmark detection algorithms.

INDEX TERMS Weakly-supervised, facial landmark, generative adversarial network, region-based convolutional neural network.

I. INTRODUCTION

Facial component and landmark detection are important procedures in a multitude of face analysis tasks including face recognition [1], [2], facial expression analysis [3], face reconstruction [4], and face enhancement [5]–[7]. With the enormous advancement of deep learning, the performance of many computer vision tasks, e.g. facial component and landmark detection, have been improved significantly. Generally, the success of applying deep learning to facial component and landmark detection relies on a reliable deep architecture with optimal parameters, which are trained and finely tuned using a large amount of training data with accurate and detailed annotations. Without enough quantity and quality of fully labeled training samples, however, detecting facial components and landmarks in images with severe occlusions is a formidable challenge.

Most of the traditional facial component detection algorithms rely on shallow models, such as SVM [8], Gabor Wavelet [9], and Bag-of-Words [10], which may fail to combine with facial alignment methods effectively. On the other hand, facial landmark detection approaches can be categorized into three types, the template fitting approach [11], [12], cascaded shape regression based methods [13], [14] and deep-learning-based models [15], [16]. Traditional methods, such as template fitting approaches and regression-based models, heavily rely on prior knowledge and artificial design feature, which might not be able to extract essential features for face alignment. Recently, deep-learning-based facial landmark detection methods [15]–[21] have achieved remarkable results. Sun *et al.* [19] propose a cascaded convolutional neural network model with 23 layers, which requires a huge amount of computational power during training and testing. Kumar *et al.* [20] design a coarse-to-fine framework, of which the input is not only raw pixels but a set of given landmarks. The model has been trained four times with input images at different scales. Kowalski *et al.* [21]

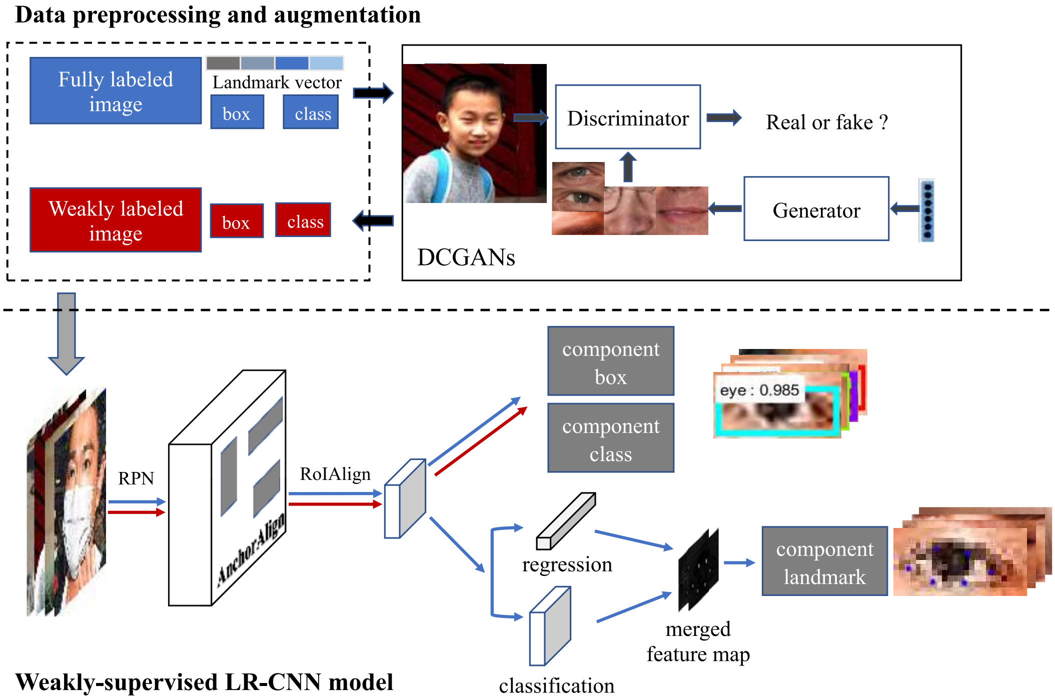


FIGURE 1. The whole training pipeline mainly consists of two parts: (1) data preprocessing and augmentation, (2) weakly-supervised LR-CNN model. We put fully labeled data into DCGANs to generate weakly labeled data. Fully labeled training data include component bounding box, component class and landmark. Weakly labeled training data only contain component bounding box and component class. All these training data are sent to LR-CNN model for weakly-supervised training. Blue boxes and lines represent fully labeled data and fully-supervised learning processing, while red boxes and lines show weakly labeled data and weakly-supervised learning. When testing, test images are put into LR-CNN model to make predictions.

present a Deep Alignment Network trained by entire face images, which is robust to large variations in difficult initializations and head poses.

Although these algorithms have good performance in laboratory environment, they fail in some cases. First, lack of enough training data with detailed annotations will lead to poor generalization of deep learning models. In the case of fully supervised learning for locating facial landmarks, training images with corresponding pixel-level landmark annotations are highly demanded. However, it is often difficult to obtain pixel-level annotations, which are expensive and time-consuming. Second, existing face detectors fail to localize facial landmark in the real-world conditions owing to severe occlusions. When large occlusion areas occur, existing face detectors may fail to detect faces and miss the responding landmarks. In addition, since these methods detect landmarks in a whole face image, occlusion areas as the uninformative pixels effect the detection results of unblocked areas. Third, it lacks an end-to-end deep learning framework for facial component and landmark detection.

In this paper, our pipeline mainly consists of two parts: data preprocessing and augmentation, and weakly-supervised LR-CNN model. In the first part, we utilize DCGANs to generate facial component images with weak labels and convert facial landmark into a landmark vector. Then, we propose a weakly-supervised LR-CNN (landmark-region-based CNN) facial component and landmark detection algorithm, which firstly detects visible facial components (i.e. eyebrow, eye, nose and mouth), followed by estimation of landmarks based on the component location and classification results. The whole pipeline is shown in Figure 1. Overall, the main contributions of this paper are three-fold:

(1) In order to cope with lack of training data with detailed annotations, we consider to replace pixel-level annotated data with easily generated weakly labeled data. We propose a DCGAN-based data preprocessing and augmentation to generate facial component samples with weak labels effectively. After weakly-supervised learning on above data, our LR-CNN model has a better landmark detection result, compared to just with fully-supervised learning.

(2) The proposed LR-CNN pipeline can tackle the large occlusion problem through detection of the visible facial components instead of a whole face in an image. AnchorAlign, RoIAlign, and a two-branch landmark detection model are presented in LR-CNN architecture, so that our method can detect facial components and landmarks simultaneously. The two-branch framework includes pixel-level classification, and landmark regression.

(3) This work is the first attempt to propose a comprehensive end-to-end framework, which firstly locates facial

components and then infers corresponding landmark coordinates. The experimental results indicate that our algorithm outperforms state-of-the-art methods.

The rest of this paper is organized as follows. Section II briefly reviews relevant work on facial component detection, facial alignment and R-CNN approach. In Section III, the main ideas and details of our framework are proposed, including data preprocessing and weakly-supervised LR-CNN model. Experiments and analysis are presented in Section IV. Finally, we conclude this paper in Section V.

II. RELATED WORK

A. FACIAL COMPONENT DETECTION

Detection of facial components as a significant step of face analysis is aimed at detecting facial components like eyebrows, eyes, mouth, nose, either given a known face detection or under the assumption that there is only a single face in the image. Yi *et al.* [23] use mapping-based localization to detect eyes and lip region with a fixed face structure. This method fails to facial deformation or expression change. Urschler *et al.* [24] present an algorithm for detecting face and facial component candidates, and for robustly voting for the best face and eyes. But it can only detect eye and mouth regions. Naruniec *et al.* [25] employ Discrete area filters for face detection and facial feature detection, which focus on fiducial point detection of facial components with complex computing. Sudhakar and Nithyanandam [9] use Gabor filter to detect facial components including left eye, right eye, nose, mouth, and also detect facial points in each component area. But this method is unable to distinguish eyebrow and eye region. These algorithms can only detect and guess all the components if some components are obscured, by using shallow models with fixed component structural relation.

B. FACIAL LANDMARK DETECTION

Traditional landmark detection approaches with shallow models can be divided into two main categories, which is named as template fitting approaches and regression-based algorithms. (1) The former methods aim to learn a shape model from training set and to fit input pictures during testing. The pioneering works of template fitting algorithms are ASM [11] and AAM [12]. As for ASM, the shape of face is represented by the linear combination of basic shapes learning via PCA and appearance of face is modeled by different pre-trained templates. In AAM, the shape representation is similar with ASM while the appearance is modeled by PCA in regular coordinate system that eliminates shape changes. (2) Regression-based methods estimate landmark locations explicitly by regression using image features. Burgos-Artizzu *et al.* [13] and Cao *et al.* [14] use cascaded and random fern regression with pixel-difference features. Zhou *et al.* [17] employ random regression forest to cast votes for landmark location based on local image patches using Haar-like features. Different from them, our approach takes raw pixels as input and extracts the essential features of the input by LR-CNN model.

Recently, deep learning techniques are being widely applied to facial landmark detection, so that the accuracy is promoted undoubtedly. These methods usually regard landmark localization task as a regression problem. The common methods can be also divided into two types: one is given the initial position estimation, network learns the error between the true value and the estimation, and reduces the error between the output value and the real value through the iterative operation. The other is to predict the location of the key points directly. The most representative algorithm of the former method is proposed by Fan and Zhou [26]. They build an accurate and robust facial landmark localizer using deep learning tools, which includes two levels of convolutional neural network for course-to-fine prediction. The representation of the second approach is a multi-task learning algorithm for both facial attribute estimation and five-points landmark detection [27]. Another algorithm presented by Zhang *et al.* [28] is for simultaneous facial action unit recognition and facial landmark detection. Similarly, our algorithm can also be regarded as a multi-task method, i.e. we classify the bounding boxes of component location and estimate the landmarks of corresponding bounding boxes by using deep network model.

C. R-CNN FOR OBJECT DETECTION

CNN-based object detection is a good way to solve variability in illumination, viewpoint and occlusion problems. In recent years, CNN has made a breakthrough in the field of object detection with the advantages of high-level features in the extraction of images. Girshick *et al.* [29] and Girshick [30] propose R-CNN and Fast R-CNN, and Ren *et al.* [31] present Faster R-CNN. Modern object detectors predominantly follow the R-CNN (Region-based Convolutional Neural Network) framework: first an object proposal algorithm generates proposals with high probability containing objects, then a CNN classifier estimate the classification of each proposal, at last employed the NMS (Non-Maximum Suppression) algorithm to merge and filtrate proposals as objects. The improved methods based on R-CNN had more fluent pipelines and also become faster such as Fast R-CNN and Faster R-CNN. Faster R-CNN achieves 73% of the mAP (mean Average Precision) on the VOC2007, when using VGG-Net [32] as the feature extractor. Moreover, it reaches a speed at 17fps compared with R-CNN at 47 seconds per image. The reason is that Faster R-CNN shares feature map from CNN to generate proposals through RPN model (Region Proposal Network) and projects all proposals to the uniform size feature, rather than use CNN to extract features of each proposal while testing. In order to complete instance segmentation, He *et al.* [33] extends Faster R-CNN to Mask R-CNN, by adding a branch for predicting an object mask in parallel with the existing branch for bounding box recognition.

All approaches with shallow models have a quite sophisticated, hand engineered image processing pipeline in

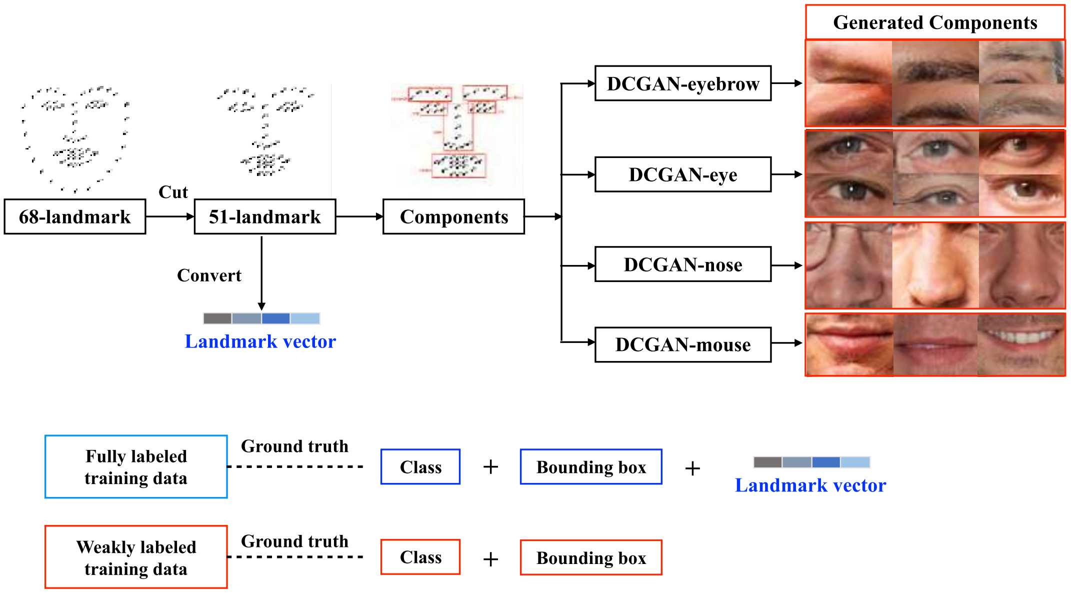


FIGURE 2. Data preprocessing and augmentation. We cut 68-landmark to 51-landmark, and convert 51-landmark into a landmark vector as finely labeled ground truth, referring to Table 1. According to landmark coordinates, component regions are cropped and put into DCGANs to generate four different 'fake' components as weakly labeled data.

common, not as good as the performance of deep learning methods. However, existing deep models for facial landmark detection require massive training data with finely labeled annotations, and fail to detect landmark accurately when it comes to large occlusion. This paper is the first attempt to detect facial components and landmarks simultaneously by using a weakly-supervised deep learning with a small amount of fully labeled training data.

III. METHODOLOGY

The whole pipeline mainly consists of two parts. The first part is data preprocessing and augmentation, which generates weakly labeled training data. The second part is our weakly-supervised LR-CNN framework for facial component-landmark detection.

A. DATA PREPROCESSING AND AUGMENTATION

Most facial landmark detectors require a large amount of training data with pixel-level annotations, such as 68-point landmarks. Since lack of training data with landmark-level labels, we consider to replace fully labeled data with weakly labeled data, so that our LR-CNN model can be trained by a small amount of fully labeled data and a large amount of weakly labeled data. In this section, we firstly generate massive weakly labeled data as training data preparation. Then, we design a landmark vector as the ground truth of fully labeled data, in order to achieve back propagation of fully-supervised part of LR-CNN model. The ground truth of weakly labeled data is component bounding-box coordinates and component class, while the ground truth of fully labeled data includes component bounding-box coordinates, component class, and landmark coordinates.

The proposed data preprocessing and augmentation is shown in Figure 2. One of the standard facial landmark benchmark has 68 points in one face, including jawline, eyebrow, eye, nose and mouth. We cut 68-landmark to 51-landmark by removing jawline. On one hand, each component region can be calculated by facial landmark coordinates. Maximum and minimum coordinates of landmarks in each component form a rectangular region, followed by proper amplification (1.25 times). Then, we put these real component images into 4 DCGAN models of different components to generate a large amount of 'fake' component images. After assign pseudo label to every 'fake' component image, weakly labeled data are ready for training. On the other hand, 51-landmark is converted into a 40-dimensional landmark vector which is regarded as fully-supervised labels, according to Table 1.

TABLE 1. The landmark indexes and the vector length of each facial component in the 68-landmark annotation.

	jawline	eyebrow	eye	nose	mouth
number	1-17	18-22, 23-27	37-42, 42-48	28-36	49-68
length	—	10	12	18	40

1) WEAKLY-SUPERVISED TRAINING DATA

Weakly-supervised training data are the main training set for LR-CNN model. Existing GAN-based data augmentation are directly generating face images, and they still require manually labeling. In addition, since the size of face image is larger than that of component image, GAN model for face images is often unable to converge, and GAN training is not well controlled. Therefore, we decide to generate different components respectively and automatically marking with weak labels. Considering the features and sizes of face components are different, in Figure 2, four DCGANs [35] to generate four categories of 'fake' images containing four different components, i.e. eyebow, eye, mouth and nose. Each DCGAN model is independent of each other and has different hyper-parameters to generate different facial components. When training DCGANs, we update the generator G three times when updating the discriminator D once, other than original settings of DCGAN. After several experiments, we train the generator G to perform better than the discriminator D. In order to learn the generator's distribution p_g over each type of components, we define a prior on input random noise variables $p_z(z)$. Variable z obey the standard normal distribution $N(0, 1)$. Then we represent a mapping to data space as $G(z; \theta_g)$, where G is a differentiable function represented by a full convolutional neural network with parameters θ_g. We also define a second full convolutional neural network $D(x; \theta_d)$ that outputs a single scalar. $D(x)$ represents the probability that x came from the data rather than p_g. We train D to maximize the probability of assigning the correct label to both training examples and samples from G. We simultaneously train G to minimize $log(1 - D(G(z)))$. In other words, D and G play the following two-player minmax game with value function $V(G, D)$:

$$\begin{aligned} minmaxV(D, G) = E_{x\sim p_{data}(x)}[logD(x)] \\ + E_{z\sim p_z(z)}[log(1 - D(G(z))))] \end{aligned} \tag{1}$$

Four DCGANs are trained with SGD (stochastic gradient descent) in a mini-batch size of 64. All of weights are initialized from a zero-centered normal distribution with standard deviation 0.02. In the LeakyReLU, the slope of the leak was set to 0.2 in all models. We leverage the Adam optimizer with tuned hyper-parameters to accelerate training, and momentum is 0.5. The learning rate is 0.0002. For training DCGANs, the training sets of real face images include Helen [40], IBUG [41], AFW [42], and LFPW [43] dataset. When 'fake' components are generated, we replace real components with 'fake' components in real face images, and mark them with weak labels (component class and bounding box) automatically. For class label, it is obvious that four DCGANs generate the images of eyebrow, eye, nose and mouth respectively, i.e. the DCGAN-eye can just output eye images. For bounding box label, we directly replace real component images with 'fake' component images in real face images. Thus, the ground truth of 'fake' components are the ground truth of real components. Finally, we generate 60,000 weakly-labeled data (ground truth: bounding box and category).

In the weakly-supervised training processing on LR-CNN model, we incorporate the proposals generated by RPN [31] into LR-CNN network. In the 2,000 candidate bounding boxes of each training sample, we randomly select 64 candidate boxes as a batch, which contains 16 positive samples (IoU to ground truth larger than 0.5, IoU is Intersection over Union as shown in Eq.(2)) and 48 negative samples (IoU to ground truth larger than 0.1 and smaller than 0.5). For positive samples, their coordinates are converted into a vector (x, y, w, h) in relation to ground truth bounding box which each sample belongs to, in Eq.(3). The subscript s indicates the center coordinates of bounding box, and subscript g and t indicate the ground truth and training sample respectively. As for negative samples, we drop out of negative training samples and mark them with a background label for component classification.

$$IoU = \frac{DetectionResult \cap GroundTruth}{DetectionResult \cup GroundTruth} \tag{2}$$

$$(x, y, w, h) = (\frac{x_{gs} - x_{ts}}{w_s}, \frac{y_{gs} - y_{ts}}{h_t}, log\frac{w_g}{w_t}, log\frac{h_g}{h_t}) \tag{3}$$

2) FULLY-SUPERVISED TRAINING DATA

Besides weakly labeled training data, LR-CNN model also need a small amount of fully supervised for guidance, including Helen [40], IBUG [41], AFW [42], and LFPW [43] datasets. Fully labeled data extra contain landmark-level annotation compared to weakly-labeled data. 68-landmark is cut to 51-landmark, followed by converting into a landmark vector $(x_{1s}, y_{1s}, x_{2s}, y_{2s}, x_{ns}, y_{ns})$, as shown in Eq.(4). (x_i, y_i) is the coordinate of i-th landmark. In Table 1, length is set according to different number of facial components. In addition, each sample has four weight vectors determining validity of coordinates. As for positive samples, all related coordinates of the components are available, and other components coordinates are unavailable. The negative landmarks coordinates are only valid inside the component bounding boxes.

$$(x_{ti}, y_{ti}) = (\frac{x_i - x_{ts}}{w_t}, \frac{y_i - y_{ts}}{h_t}) \tag{4}$$

Therefore, the preparation of training data has been completed. The sum of preprocessed data for LR-CNN model is about 66,000, including around 6,000 fully-labeled data (ground truth: landmark, bounding box and category) and 60,000 weakly-labeled data (ground truth: bounding box and category).

B. WEAKLY-SUPERVISED LR-CNN FRAMEWORK FOR FACIAL COMPONENT-LANDMARK DETECTION

After data preparation, a novel architecture called weakly-supervised LR-CNN is presented in Figure 3, which mainly consists of region-based component detection and two-branch landmark detection. When training, the input of network includes two parts: weakly labeled data, and fully labeled data. When testing, the input of network is just face images to be predicted. Firstly, we leverage ResNet-50 [36] model to extract convolutional features of input image and share them to our RPN with AnchorAlign model, for calculating RoIs (Region of Interest). After RoIAlign layer, fixed size feature map is put into component detection model and landmark detection model simultaneously. The component detection model predicts component bounding box and category. For landmark detection, a two-branch

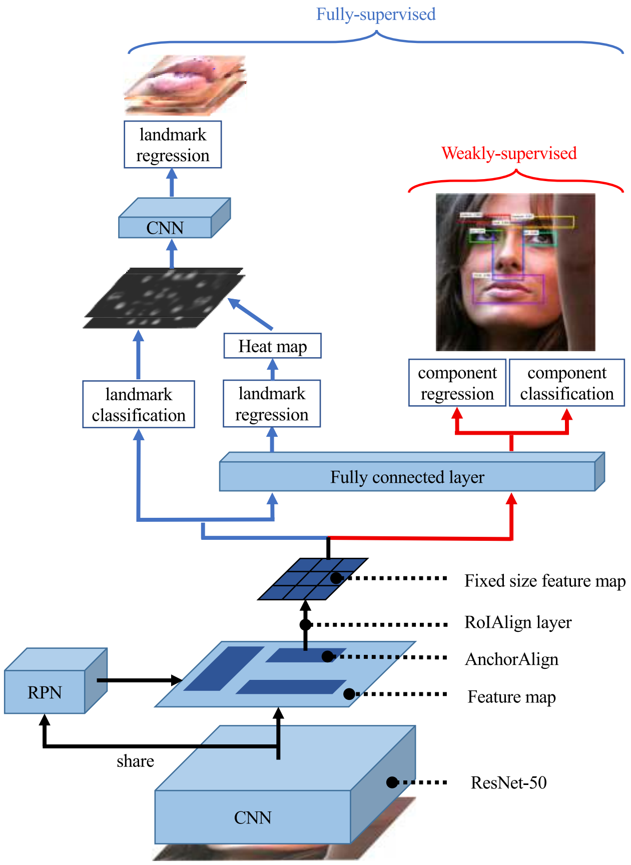


FIGURE 3. Our LR-CNN architecture. Convolutional features are extracted by Resnet-50 model and shared to RPN with AnchorAlign. After RoIAlign layer, RoIs with unified size are put into component detection and the two-branch landmark detection model. The outputs include component bounding boxes, categories and landmarks. When training, weakly-supervised data contain bounding box coordinates and component category, while the ground truth of landmark is extra included in fully supervised data.

landmark detection model is proposed, which consists of a landmark classification branch and a landmark regression branch. Finally, the whole framework outputs three parts: component bounding box coordinates indicating the offsets between ground truth and the RPN proposal; component category showing the category of the proposal region, i.e. eyebrow, eye, nose and mouth; component landmark demonstrating the landmarks distribution with the proposal belongs to corresponding category.

1) REGION-BASED COMPONENT DETECTION

In the initial part of the proposed framework, we adapt ResNet-50 model to extract feature maps of input images and share them with RPN(region proposal network) [31] to generate RoIs by using AnchorAlign, followed by fixing feature size through RoIAlign. Finally, the fixed feature map is put into fully connected layer to predict component class and bounding box. Next, we introduce the proposed AnchorAlign and RoIAlign model, for accurately localizing components and landmarks.

a: AnchorAlign

Anchor [31] is no longer a stranger in object detection area, which can address multiple scales and aspect ratios. Generally, we take 9 anchors for granted in detecting object of Faster R-CNN. However, for our facial component-landmark detection task, the situation has become different. Since the scales and ratios of components are different from normal object, we design AnchorAlign by changing multiple scales and aspect ratios to adapt to facial components, so that AnchorAlign model can improve the localization accuracy. Comparing to Anchor, AnchorAlign can be used in a specific application. Unlike Anchor settings of Faster-RCNN, the scales and ratios of AnchorAlign is not selected through manual experiments. It reasonably relies on component structure, since the bounding boxes of components are approximately {2:1, 1:2, 1:3} ratios for {eye, nose, eyebrow/mouse}. Additionally, the scales of components are also different from the object detection task. The areas of components occupying the entire face has a certain regularity below about 256x256, while ordinary objects randomly appear on the image and have unfixed sizes. Therefore, in the RPN, we design a specified AnchorAlign model as shown in Table 6 and gain a better result, compared with Anchor of Faster R-CNN and Mask R-CNN.

b: RoIAlign

RoIPool [31] is a standard operation for extracting a small feature map from each RoI, but it misalignments between the RoI and the extracted features. To address predicting pixel-accurate landmarks, we employ RoIAlign that removes the harsh quantization of RoIPool, properly aligning the extracted features with the input. This operation greatly increases the accuracy of landmark detection, while it may be not beneficial for component detection and classification.

In the output step, we leverage standard regression and classification method for object detection, as the same as Mask R-CNN. $\mathcal{L}_{reg}$ is loss of component bounding box regression, and $\mathcal{L}_{cls}$ is loss of component classification.

$$\mathcal{L}_{reg} = \sum_i class_i \cdot Smooth_{L1}(box_i, \hat{box_i}) \tag{5}$$

$$\mathcal{L}_{cls} = \sum_i Softmax(class_i, \hat{class_i}) \tag{6}$$

where, i is the index of a proposal region in an image. box_i and $\hat{box_i}$ are the predicted offset and true offset value between the i-th proposal region and its corresponding ground truth bounding box. $\hat{class_i}$ and $class_i$ indicate ground-truth and predicted classification of the proposal region. SmoothL1 and Softmax are shown in Eq.(7) and Eq.(8) respectively. Compared with traditional Euclidean distance, SmoothL1 can reduce the outlier effect, and in that way our model converges faster.

$$Smooth_{L1}(x) = \begin{cases} 0.5x^2, & |x| < 1 \\ |x| - 0.5. & others \end{cases} \tag{7}$$

$$P(i) = \frac{exp(\theta_i^T x)}{\sum_{k=1}^{K} exp(\theta_k^T x)} \tag{8}$$

2) TWO-BRANCH LANDMARK DETECTION

In Figure 3, the two-branch landmark detection model consists of landmark classification and landmark regression model.

For the classification branch, each of the landmark of a component is a one-hot $m * m$ binary key-point where only a single pixel is labeled as foreground. As we know, Mask R-CNN is a framework for instance segmentation, through extending Faster R-CNN detector with a mask branch. This mask branch motivates us to classify every pixel of the fixed size feature map as a one-hot mask. If the pixel is landmark, the model output 1 for this pixel. If not, the model outputs 0. $landmark_{icls}$ is pixel-to-pixel classification of i-th proposed component region, as shown in Eq.(9).

$$landmark_{icls} = \sum_i crossentropy(p_{i(cls)}, p_{i(\hat{comp})}) \tag{9}$$

where $p_{i(\hat{comp})}$ is the ground truth of landmark for i-th proposal. $p_{i(cls)}$ is the predicted landmark relative coordinates according to the i-th proposal, calculated by classification. In the cross-entropy function t and o represent $p_{i(cls)}$ and $p_{i(\hat{comp})}$, in Eq.(10).

$$crossentropy(t, o) = -[t * log(o) + (1 - t) * log(1 - o)] \tag{10}$$

For the regression branch, fully connected layer is used to infer component landmark as a regression task. The landmark vector derived from Table 1 can be regressed along with bounding boxes regression. Then, for every component, we use Gaussian model to generate the heat map on the basis of key-point regression results. $landmark_{ireg}$ is landmark vector regression of i-th proposed component region, as defined in Eq.(11).

$$landmark_{ireg} = \sum_i Smooth_{L1}(p_{i(reg)}, p_{i(\hat{comp})}) \tag{11}$$

where $p_{i(reg)}$ is the predicted landmark relative coordinates according to the i-th proposal, calculated by regression. SmoothL1 function is shown in Eq.(7).

Finally, since each branch of facial component feature map is local spatial related, the feature maps of two branches are stacked over as a fused local receptive field. Then, the fused feature maps are fed into a CNN model for learning local spatial structures. The final output of the two-branch model is also inferred by landmark regression model. All the components' landmarks are combined together to form a whole facial landmark.

Our loss for landmark detection is defined in Eq.(12):

$$\mathcal{L}_{land} = \sum_i Smooth_{L1}[f_{i(comp)}(landmark_{icls}, landmark_{ireg}), p_{i(\hat{comp})}] \tag{12}$$

$f_{i(comp)}(landmark_{icls}, landmark_{ireg})$ and $p_{i(\hat{comp})}$ are the predicted and ground truth landmark relative coordinates according to i-th proposed component region. $landmark_{icls}$ is calculated in cross-entropy, as shown in Eq.(9). $landmark_{ireg}$ regression function is in Eq.(11).

3) WEAKLY-SUPERVISED LEARNING AND LOSS

Our learning system consists of two parts, a main part of weakly-supervised learning and a small amount of fully supervised learning, as shown in Figure 2. For weakly-supervised learning, we only utilizes our weakly labeled training data generated by DCGANs. The ground truth of weakly labeled data is component bounding box and class. For fully-supervised learning, a small amount of fully labeled data are also used for guiding back-propagation of neural network while training. Comparing to weakly-supervised data, the ground truth of fully-supervised data extra includes landmark vectors. In total, the number of weakly-supervised training data is much larger than that of fully-supervised data. Though weakly-labeled data without landmark coordinates ground truth, the result of landmark detection is also enhanced greatly. This is because weakly-supervised learning make component localization and recognition results more accurate. Since landmark detection results strongly rely on the predicted component detection results, the improvement of component detection results have a positive influence on landmark detection results. This is the core idea of our landmark detection strategy based on the facial component regions.

We use a multi-task loss $\mathcal{L}$ on each weakly labeled data and fully labeled data to jointly train. Our loss function for an image is defined as:

$$\mathcal{L} = \mu(\lambda_1 \mathcal{L}_{reg} + \lambda_2 \mathcal{L}_{cls}) + (1 - \mu)\mathcal{L}_{land} \tag{13}$$

The hyper-parameter μ, λ_1, λ_1 in Eq.(13) control the balance among the three task losses. μ represents weakly-supervised weight, and it is determined by the number of weakly-supervised samples. λ_1 and λ_1 represent the loss weight of component bounding box regression $\mathcal{L}_{reg}$ and component classification $\mathcal{L}_{cls}$. Each of three terms has a loss weight indicated to adjust the affect of each loss part. All experiments use $\lambda_1 = \lambda_2 = 0.5$, $\mu = 1/3$, to make our network focus on landmark detection task.

4) TRAINING

The LR-CNN framework can be trained end-to-end by back-propagation and SGD. We follow the "image-centric" sampling strategy from [30] to train our network. Each mini-batch arises from images that include positive and negative example anchors, which are defined in data preprocessing and augmentation.

The shared convolutional layers (ResNet-50) are initialized by pre-training a model for ImageNet 1000-class dataset [38], as is standard practice. We randomly initialize all other layers by drawing weights from a zero-mean Gaussian distribution with standard deviation 0.01. We tune all layers with weakly labeled and fully labeled data. Each mini-batch has 2 images per GPU and each image has 64 sampled RoIs, with a ratio of 1:3 of positive to negative [33]. We train on 4 GPUs (so effective minibatch size is 16) for 160k iterations, with a learning rate of 0.02 which is decreased by 10 at the 120k iteration. The weight decay is 0.0001 and momentum is 0.9. Our framework is also fast to train. Training with ResNet-50 on takes 30 hours in the synchronized 4-GPU implementation (0.98s per mini-batch = 16 samples).

The training set for LR-CNN consists of two part: (1) Helen [40], IBUG [41], AFW [42], and LFPW [43] as fully-supervised training data; (2) our generated weakly-supervised training data.

5) TESTING

At test time, the test face images are directly put into the trained LR-CNN model, without any data preprocessing. The steps of testing and training in LR-CNN are almost the same, as shown in Figure 3. The proposal number is 1000 for RPN. The scales and ratios of AnchorAlign is $\{64^2, 128^2, 256^2\}$ and {2:1, 1:2, 1:3}. For the two-branch model, we only compute landmarks on the top 100 component detection boxes. We test on Helen and LFPW test sets. In addition, 300-W test set [45] is used as evaluation dataset due to its challenging for their variability in illumination, viewpoint and occlusion.

IV. EXPERIMENTS AND DISCUSSIONS

In the experiment, several benchmark datasets are used to train and test on our model. Firstly, we introduce our datasets and evaluation measurements. Then, we compare our method with other state-of-the-art algorithms in both facial component detection task and landmark detection task. In the ablation experiments, we discuss the performance of each proposed method in detail.

We implement the Caffe [39] framework for all training, inference, and testing, in a regular PC (3.2-GHz 8-core CPU, 32G RAM, 4×12G GPU and Ubuntu 14.04). The whole training costs 30 hours on four NVIDIA TITAN X Pascals. Our algorithm reaches a speed at 0.21s per image while testing.

A. DATASET AND EVALUATION MEASUREMENT

The sum of training data for LR-CNN is about 66,000, including around 6,000 fully labeled data (ground truth: landmark, bounding box and category) and 60,000 weakly labeled data (ground truth: bounding box and category). Helen [40], IBUG [41], AFW [42], and LFPW [43] are fully-supervised training data, while our generated weakly labeled data are weakly-supervised training data. We evaluate our method on Helen, LFPW and 300-W test sets.

Our facial component and landmark detection algorithm is a multi-task method. The performance of methods is measured by two indexes, which is average precision (AP) for component detection and average error distance for landmark detection, shown in Eq.(14) and Eq.(15) respectively.

$$AP = \int_0^1 p(x)dx \tag{14}$$

Average precision computes the average value of p(x) over the interval from x=0 to x=1 and is the area under the precision-recall curve.

For facial landmark detection, the normalized error rate is used to represent the good or bad of an algorithm in Eq.(15).

$$e = \frac{1}{N}\sum_{i=1}^{N} \frac{\frac{1}{M}\sum_{j=1}^{M} |p_{i,j} - g_{i,j}|_2}{|le_i - re_i|_2}. \tag{15}$$

Here, N is the number of test samples and M is the number of landmarks ($M = 51$ in our experiment). $p_{i,j}$ and $g_{i,j}$ are the predicted coordinates and real coordinates of j-th landmark ground truth of i-th test sample respectively. le_i and re_i are the center coordinates of the left eye and the right eye of the i-th test sample, respectively.

B. COMPARISON WITH OTHER STATE-OF-THE-ART METHODS

To better understand the advantage of the proposed method, the experimental results of component detection and landmark detection are compared to other state-of-the-art algorithms separately, including both shallow models and deep models.

1) PERFORMANCE OF FACIAL COMPONENT DETECTION

In order to illustrate the result of component detection, we use average precision (AP) as a rule to compare each facial component detection precision and mean average precision (mAP) to all facial component detection precisions. We compare against several the-state-of-art object detection methods, such as Mask R-CNN [33], Faster R-CNN [31], SSD [46], YOLO [47] and YOLOv2 [48], shown in Table 2. All the methods are trained by generated weakly-supervised training set, and evaluated on the test set mixed by Helen, LFPW, and 300-W test set. As we can see, our algorithm has the best performances in every category of facial component, especially by 0.919 and 0.921 in nose and mouse. Actually, for component detection task, our method is fully-supervised learning. However, for landmark detection, our method is weakly-supervised learning. The mainly reason why our method outperforms other algorithms is our data

TABLE 2. The component detection result (AP) compared with other methods.

Algorithm	mAP	eyebrow	eye	nose	mouth
Faster R-CNN [31]	0.751	0.613	0.762	0.809	0.821
Mask R-CNN [33]	0.765	0.640	0.776	0.812	0.832
SSD [46]	0.764	0.631	0.784	0.808	0.834
YOLO [47]	0.646	0.487	0.674	0.727	0.696
YOLOv2 [48]	0.732	0.596	0.747	0.781	0.803
LR-CNN	**0.861**	**0.704**	**0.898**	**0.919**	**0.921**

TABLE 3. The landmark detection results (average error distance) compared with other methods, on Helen, LFPW and 300-W test set separately.

	Algorithm	Helen	LFPW	300-W	300-W	300-W
				common	challenge	full
Non-deep models	RCPR [13]	5.93	6.56	6.18	17.26	8.35
	CFAN [15]	5.53	5.44	5.50	–	–
	SDM [49]	5.50	5.67	5.57	15.40	7.50
	CDM [50]	12.86	24.68	10.10	19.54	11.94
	GN-DPM [51]	5.69	5.92	5.78	–	–
	CFSS [52]	4.63	4.87	4.73	9.98	5.76
Deep models	RAR [16]	–	–	**4.12**	8.35	4.94
	LDDR [20]	4.76	4.67	–	–	–
	TCDCN [53]	4.60	–	4.80	8.60	5.54
	CFT [54]	4.75	–	4.82	10.06	5.85
	DAN [21]	–	–	4.42	**7.57**	5.03
	LR-CNN(51L)[1]	3.03	3.12	3.02	6.65	4.25
	LR-CNN(51L)[1]+RAR(17L)[2]	**3.71**	**4.07**	4.61	8.56	5.32
	LR-CNN(51L)[1]+RAR(17L)[2]+300W[3]	4.86	4.79	**4.12**	8.26	**4.92**

[1] LR-CNN(51L) mean that 51-landmark is the result of our method, trained on a small fully supervised and a weakly-supervised training set.

[2] RAR(17L) means that we use RAR method to detect another 17-landmark of jawline, for a clear and fair comparison with 68-landmark results.

[3] 300W means that we additionally train on a training set with only 300-W, in order to compare with other methods which only focus on 300-W, like RAR and DAN.

preprocessing, which auto-annotate and augment training data effectively. AnchorAlign also play an important role in component detection, because other models are not suitable for facial component detection. In addition, compared with other R-CNN methods, LR-CNN employs batch normalization after convolutional layers to avoid over-fitting problem. Comparing to Yolo, we use batch normalization in CNN model, and dropout in fully connected layers. In Figure 6, we observe that the proposed method is robust to faces with large pose variation, lighting, and severe occlusion.

2) PERFORMANCE OF FACIAL LANDMARK DETECTION

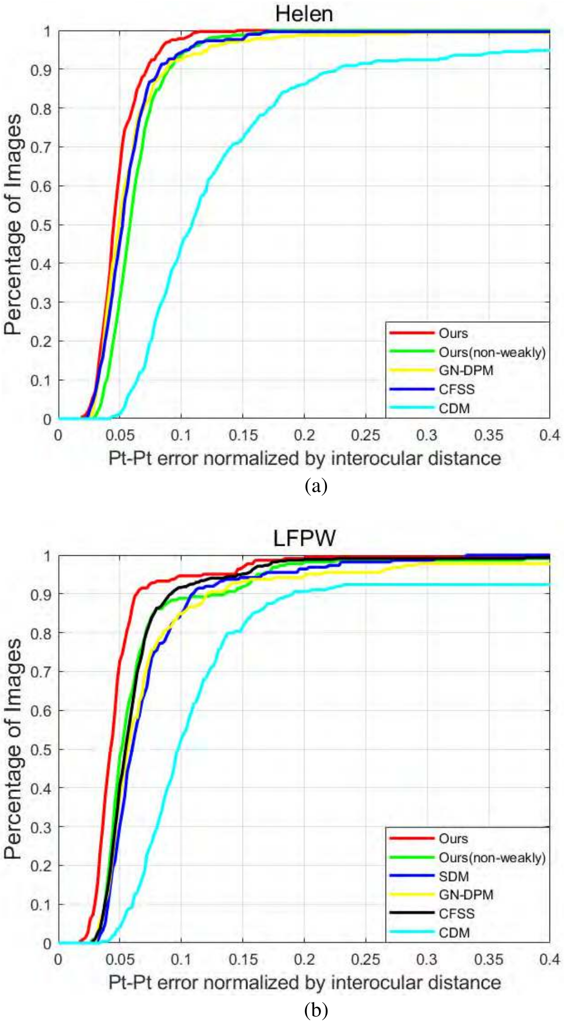
As mentioned before, several benchmark test sets are used to evaluate performance of different methods, including Helen, LFPW, and 300-W. We compare to non-deep models: (1) Robust Cascaded Pose Regression (RCPR) [13] using the publicly available implementation and parameter settings; (2) Coarse-to-Fine Auto-Encoder Networks (CFAN) [15], which focuses on real-time face alignment; (3) Supervised Descent Method (SDM) [49]; (4) Cascaded Deformable Shape Model (CDM) [50]; (5) Gauss-Newton Deformable Part Models (GN-DPM) [51]; (6) Coarse-to-fine shape searching (CFSS) [52]. And we also compare against deep models: (7) Recurrent Attentive-Refinement Networks (RAR) [16]; (8) Local Deep Descriptor Regression (LDDR) [20]; (9) Tasks-Constrained Deep Convolutional Network (TCDCN) [53]; (10) Coarse-to-fine training algorithm (CFT) [54]. (11) Deep alignment network (DAN) [21]. Given that our method is aimed at 51-point landmark, we combine LR-CNN of 51-point (eye, eyebrow, nose, mouse) with RAR algorithm of 17-point (jawline) for testing on 68-point landmark detection, for comparison to other approached listed above. As shown in Table 3, average error distances of all algorithms are measured by 68-point landmark detection result.

a: EVALUATION ON HELEN

It is obvious that deep learning models produces a superior performance to shallow models on Helen test set, in Table 3. And LR-CNN(51-landmark)+RAR(17-landmark) outperforms all other state-of-the-art methods, far below CFSS and TCDCN. The proposed model perform best on Helen test set with average error distance less than 3. Figure 4 (a) shows several algorithms' cumulative error curves. As we can see, our algorithm performs better than other state-of-art algorithms. Figure 6 shows several examples of our detection, including component and landmark. We observe that the proposed method is robust to lighting and severe occlusion. It is worth pointing out that the size of input images is non-restricted, which means that LR-CNN can cope with both low-resolution and high-resolution images. We make a comparison between our method and LDDR tested on images with extreme illuminations and occlusions, as shown in Figure 7. The results of other approaches are unreliable and rely on unfounded guesswork, while our method only detects the landmarks in visible component regions. Therefore, the detection result of our method is more reasonable and intuitive.

b: EVALUATION ON LFPW

In addition to Helen, we also tested on LFPW test set and observe similar trend as on the Helen test set. Figure 4 (b) also demonstrates the superiority of our method compared with some released code of other algorithm. Figure 6 and Figure 7 also indicate some detection examples using LR-CNN method.



(a)

(b)

FIGURE 4. **Cumulative error curves. All of those curves are produced by the released code of each algorithm. The red line (Ours) is our weakly-supervised method and the green one (Ours(non-weakly)) is our LR-CNN without generated weakly labeled data. Our algorithms performs better than other state-of-the-art algorithms, and weakly-supervised model performs much better than non-weakly-supervised model. (a) evaluation on Helen test set. (b) evaluation on LFPW test set.**

c: EVALUATION ON 300-W

We report the landmark detection results of LR-CNN method as well as results of current state-of-the-art methods on the 300-W testing set. Compared with the performance on Helen and LFPW test set, LR-CNN(51-landmark)+RAR (17-landmark) result on 300-W is barely satisfactory but still outperforms other state-of-art algorithms, except RAR and DAN. Because both RAR and DAN are trained by 300-W training set, while our model is only trained by limited fully labeled data. TCDCN pre-trains their facial landmark detection model on the Multi-Attribute Facial Landmark database which consists of 19,000 face images with multiple facial attributes information, and tunes their model on 300-W. On the other hand, the training set of our model doesn't contain 300-W data set. What's more, RAR and DAN only focus on 300-W data set and has no test on other benchmark dataset, and our model has wider applicability than those. Therefore, for fair comparison with RAR, TCDAN and DAN, we only use 300-W training set as fully supervised part to train our model, and gain a improvement on 300-W test by about 0.5 on 300-W common set and by 0.4 on 300-W full test set, as shown in the last line of Table 3. Since the original training set including Helen and LFPW training set are replaced by 300-W, the results on Helen and LFPW decline reasonably, but are also better than many other Non-deep models. The reason why DAN outperforms ours in 300-W challenge test set is that DAN is also a deep-learning-based algorithm and it is a robust alignment method of which network input are entire face images. And cascading complexity of DAN is higher than our method, undoubtedly has better performance than our straightforward regression algorithm. DAN are trained sequentially while ours is an end-to-end architecture and easily trained.

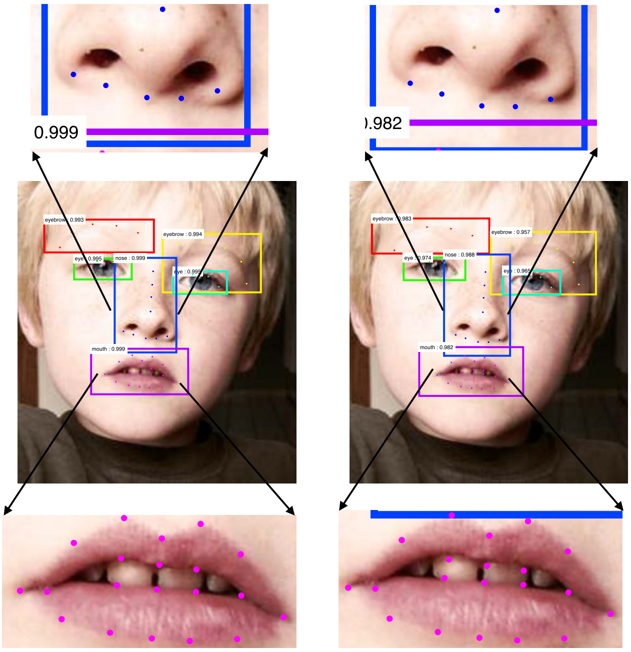


FIGURE 5. **Left image is weakly-supervised result, right image is non-weakly-supervised result.**

In addition, our method only predicts visible components and landmarks, while other algorithms guess the facial landmarks of which components are occluded. It is obvious that guesswork is unreliable and useless, as shown in Figure 7. In fact, this inaccurate estimation of landmarks is based on facial structure feature. Oppositely, our system is able to detect facial landmark precisely because our predicted landmark is based on our previous component detection results, which are trained effectively by weakly-supervised data.

C. ABLATION EXPERIMENTS

We run a number of ablations to analyze weakly-supervised LR-CNN. Results are shown in every subsections and discussed in detail next.

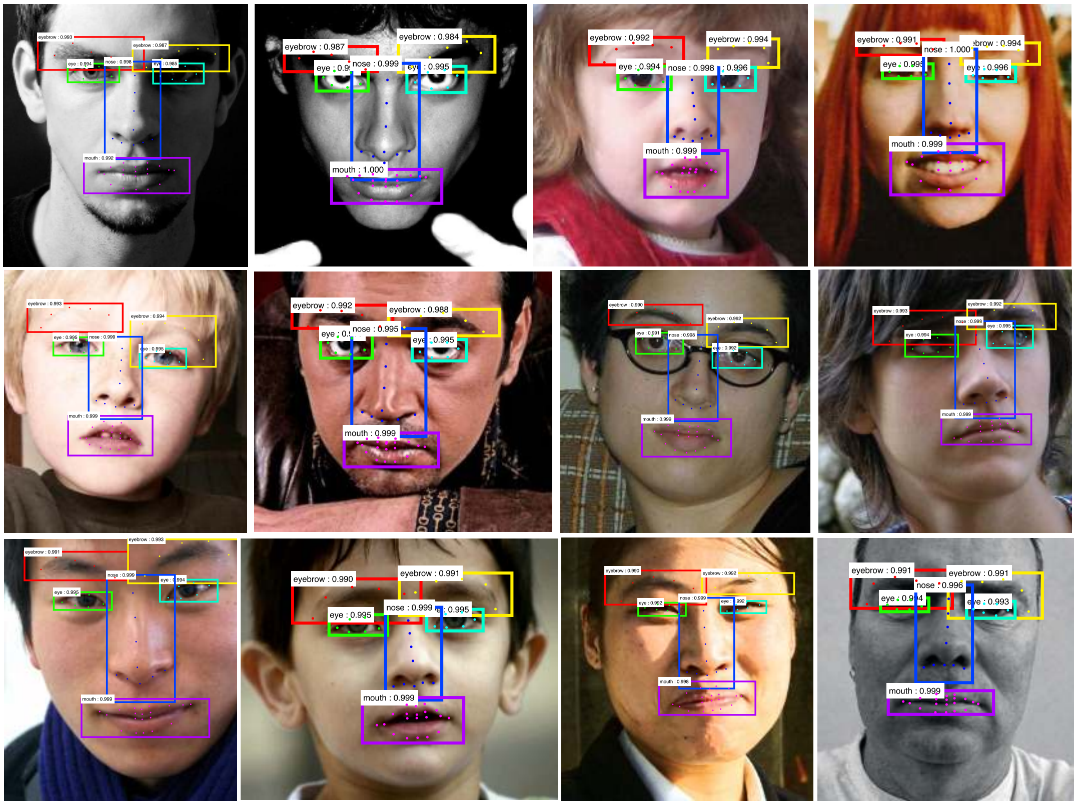


FIGURE 6. Some detection results on Helen testset (the first row), LFPW testset (the second row) and 300-W testset (the rest row). The different color bounding boxed show facial component detection and the text and number pairs denote the probabilities of bounding boxes belong to the corresponding categories. The predicted landmark coordinates are plotted by different color points corresponding their component.

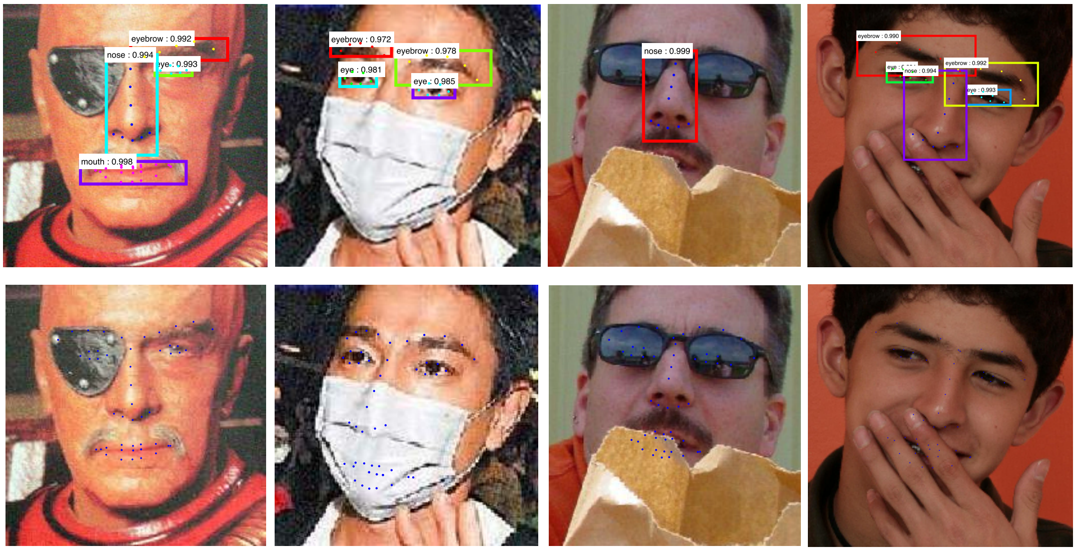


FIGURE 7. Some detection results of our method on extreme illuminations and occlusions in first row. The second row shows the result of LDDR [20].

1) WEAKLY-SUPERVISED V.S. NON-WEAKLY-SUPERVISED

To demonstrate the necessity of weakly-supervised learning in our architecture, we compare the results of two groups of experiments, which are with weakly-supervised method and without weakly-supervised method. In this controlled experiment, weakly-supervised method uses fully supervised data and weakly-supervised data to train our model while non-weakly-supervised method only uses fully supervised

TABLE 4. The component detection result (AP) compared with other methods.

Algorithm	mAP	eyebrow	eye	nose	mouth
Ours	**0.861**	**0.704**	**0.898**	**0.919**	**0.921**
Ours(non-weakly)	0.589	0.471	0.724	0.557	0.602
Ours(VGG [32])	0.846	0.680	0.889	0.909	0.907
Ours(NIN [37])	0.799	0.641	0.832	0.868	0.855
Ours(ZF [55])	0.833	0.664	0.868	0.901	0.898

TABLE 5. The 51-landmark detection result (average error distance) compared with our methods with variants on different test set.

Algorithm	Helen	LFPW	300-W full
Ours	**3.03**	**3.12**	**4.25**
Ours(non-weakly)	3.77	4.25	6.68
Ours(VGG [32])	3.10	3.25	4.69
Ours(NIN [37])	3.22	3.27	4.85
Ours(ZF [55])	3.18	3.24	4.76

data to train the same architecture. We test our method and ours(non-weakly) on facial component detection, and also on facial landmark detection with Helen, LFPW and 300-W as shown, as shown in Table 4 and Table 5 respectively. We can see that Ours outperforms Ours(non-weakly) in both facial component and landmark detection. For component detection, our generated weakly-supervised data can be regarded as data augmentation. As the number of training data increasing, our model improve a lot. Since landmark detection results strongly rely on the predicted component detection results, the improvement of component detection results have a positive influence on landmark detection results. In addition, Figure 4 shows their cumulative error curves. The red line (Ours) is our weakly-supervised method and the green one (Ours(non-weakly)) is our LR-CNN without generated weakly labeled data. Our algorithms performs better than other state-of-the-art algorithms, and weakly-supervised model performs much better than non-weakly-supervised model.

As shown in Figure 5, the left image is the detection result of our weakly-supervised method while the right one shows our pipeline with only fully supervised learning. Both ours and ours(non-weakly) can detect six parts of facial components, but localization result of ours(non-weakly) is much worse than that of weakly-supervised LR-CNN. For instance, weakly-supervised LR-CNN can locate landmarks of the nose accurately while fully supervised method provides an unsatisfied result, especially at the teeth position of the mouth.

2) COMPARISON WITH OTHER CASCADED CNNS

We also list detection results of several cascaded CNN models including VGG [32], NIN [37], ZF [55] and ResNet [33] as shown in Table 4 and Table 5. In Table 4, there is no doubt that ours with ResNet model outperforms other CNN models.

TABLE 6. Detection results of our algorithm on Helen test set using different settings of anchors. The network is ResNet-50.

settings	scales	ratios	mAP	Helen
Anchor	256^2	1:1	0.789	4.01
	256^2	{2:1, 1:1, 1:2}	0.806	3.88
	$\{128^2, 256^2, 521^2\}$	1:1	0.817	3.59
	$\{128^2, 256^2, 521^2\}$	{2:1, 1:1, 1:2}	0.829	3.35
Anchor Align	$\{64^2, 128^2, 256^2\}$	{2:1, 1:1, 1:2}	0.836	3.31
	$\{128^2, 256^2, 521^2\}$	{2:1, 1:2, 1:3}	0.857	3.10
	$\boldsymbol{\{64^2, 128^2, 256^2\}}$	**{2:1, 1:2, 1:3}**	**0.861**	**3.03**

3) THE ROLES OF ANCHORALIGN AND ROIALIGN

To investigate the behavior of AnchorAlign and RoIAlign, we conducted several ablation studies. First, we show the effect of different Anchors for component detection and landmark detection results. In this experiment, we use the ResNet-50 model with weakly-supervised learning, which is our standard settings. As shown in Table 6, our proposed AnchorAlign is compared with Anchor which is presented by Ren *et al.* [31]. Anchor changes in 4 kinds of settings, and AnchorAlign changes in 3 kinds of settings. As for the result of both component detection and landmark detection, the best performance of Anchor is still below the worst performance of AnchorAlign. By default we use $\{64^2, 128^2, 256^2\}$ scales and {2:1, 1:2, 1:3} ratios (0.961 mAP and 3.03 average error distance on Helen test set). The mAP is higher if using this kind setting of specified scales or ratios, the landmark detection has the same trend. What's more, the effect of ratio is larger than that of scale, the mAP and average error distance is 0.836 and 3.31 when we only change the ratios. But when we only change the scales, the result become much better, suggesting that scales and aspect ratios are not disentangled dimensions for the detection accuracy.

Next, we evaluate three kinds of RoI layer to demonstrate which operation is the best for our system. A comparison experiment of RoIPool, RoIWarp and RoIAlign layer is shown in Table 7. RoIAlign improves component detection mAP by about 2 points over RoIWarp. RoIAlign reduces Helen and LFPW landmark detection by about 0.7 below RoIWarp, with much of the gain coming at 300-W benchmark. RoIPool performs on par with RoIWarp and also much worse than RoIAlign. This also highlights that proper alignment is the key.

TABLE 7. Detection results with various RoI layers.

	mAP	Helen	LFPW	300-W full
RoIPool [31]	0.841	3.77	3.98	5.68
RoIWarp [56]	0.843	3.80	3.75	5.69
RoIAlign [33]	**0.861**	**3.03**	**3.12**	**4.25**

4) TWO-BRANCH V.S. ONE-BRANCH

In the architecture, we propose a two-branch model for landmark detection. This ablation experiment demonstrates

TABLE 8. Landmark detection results with various architectures.

	Helen	LFPW	300-W full
classification branch	4.31	4.58	7.17
regression branch	3.84	3.76	5.92
two-branch	**3.03**	**3.12**	**4.25**

the superiority of two branches, as shown in Table 8. We compare our two-branch model with two single-branch landmark detection models respectively. For the classification detection model, this branch directly outputs landmark result when we remove the regression branch. For the regression detection model, we eliminate the classification branch. And we train and test them separately. All these architectures are trained via weakly-supervised learning and standard settings. In Table 8, it is obvious that two-branch architecture outperforms other two models in three datasets. The regression model has better performance than the classification model by about 1.2 average error distance in 300-W full test set, which is much larger the gap between two-branch model and the regression branch. This illustrates that the two branches complement each other. It is also illustrates the regression branch is more suitable for difficult task than the classification branch, as 330-W is more challenging than Helen and LFPW. This is the reason why we also use regression method on the last layer after merging feature maps. The improvement of combination of two branches illustrates that these two method overlap and complement each other.

V. CONCLUSIONS AND FUTURE WORKS

In this paper, we propose an end-to-end weakly-supervised LR-CNN framework for facial component and landmark detection. Our presented method use DCGANs and automatic labeling to generate weakly-supervised training data, which solve the problem of small training set. Moreover, we design a two-branch architecture that makes it possible to detect facial components and predict facial landmarks simultaneously. For large area occluded faces, many existing face detectors are failed to detect any faces in the picture while ours could detect visible facial components and predict corresponding landmarks without any no sense guesswork. Experiments on benchmark datasets reveal that our method outperforms most of the state-of-art algorithms. One of the reason may be that our weakly-supervised framework is able to predict more accurate box coordinates, which thanks to weakly-supervised augmentation and data preprocessing by using generative models. This is also because that our two-branch architecture can extract more discriminative features by using classification and regression branch. We also discuss the advantages of our weakly-supervised learning compared with fully supervised learning. In addition, a comparison experiment among different AnchorAlign, RoIAlign and cascaded CNN models demonstrates the feasibility of our weakly-supervised algorithm successfully.

In the future work, we are planning to combine our weakly-supervised learning with semi-supervised learning to detect landmark of jawline. And we also hope to add some enactment and filtering algorithms [5]–[7] into face preprocessing stage to enhance face. Moreover, DCGAN and LR-CNN models could be shared features and reformed to an end-to-end model to accelerate training and testing.

REFERENCES

[1] R. R. Atallah, A. Kamsin, M. A. Ismail, S. A. Abdelrahman, and S. Zerdoumi, “Face recognition and age estimation implications of changes in facial features: A critical review study,” *IEEE Access*, vol. 6, pp. 28290–28304, 2018.

[2] Z. Xiang, H. Tan, and W. Ye, “The excellent properties of a dense grid-based HOG feature on face recognition compared to Gabor and LBP,” *IEEE Access*, vol. 6, pp. 29306–29319, 2018.

[3] C. Qi *et al.*, “Facial expressions recognition based on cognition and mapped binary patterns,” *IEEE Access*, vol. 6, pp. 18795–18803, 2018.

[4] J. Roth, Y. Tong, and X. Liu, “Adaptive 3D face reconstruction from unconstrained photo collections,” in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Las Vegas, NV, USA, Jun. 2016, pp. 4197–4206.

[5] A. Mustapha, A. Oulefki, M. Bengherabi, E. Boutellaa, and M. A. Algaet, “Towards nonuniform illumination face enhancement via adaptive contrast stretching,” *Multimedia Tools Appl.*, vol. 76, no. 21, pp. 21961–21999, 2017.

[6] I. Abdelhamid, A. Mustapha, and O. Adel, “Adaptive gamma correction-based expert system for nonuniform illumination face enhancement,” *J. Electron. Imag.*, vol. 27, no. 2, p. 023028, 2018.

[7] A. Oulefki, A. Mustapha, E. Boutellaa, M. Bengherabi, and A. A. Tifarine, “Fuzzy reasoning model to improve face illumination invariance,” *Signal, Image Video Process.*, vol. 12, no. 3, pp. 421–428, 2018.

[8] D. Xi and S.-W. Lee, “Face detection and facial component extraction by wavelet decomposition and support vector machines,” in *Proc. Int. Conf. Audio-Video-Based Biometric Person Authentication*, Guildford, U.K., Jun. 2003, pp. 199–207.

[9] K. Sudhakar and P. Nithyanandam, “An accurate facial component detection using Gabor filter,” *Bull. Elect. Eng., Inf. Technol.*, vol. 6, no. 3, pp. 287–294, Sep. 2017.

[10] B. A. Efraty, M. Papadakis, A. Profitt, S. Shah, and I. A. Kakadiaris, “Facial component-landmark detection,” in *Proc. IEEE Int. Conf. Autom. Face Gesture Recognit. Workshops*, Santa Barbara, CA, USA, May 2011, pp. 278–285.

[11] 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.

[12] 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.

[13] X. P. Burgos-Artizzu, P. Perona, and P. Dollár, “Robust face landmark estimation under occlusion,” in *Proc. IEEE Int. Conf. Comput. Vis.*, Sydney, NSW, Australia, Mar. 2013, pp. 1513–1520.

[14] X. Cao, Y. Wei, F. Wen, and J. Sun, “Face alignment by explicit shape regression,” *Int. J. Comput. Vis.*, vol. 107, no. 2, pp. 177–190, Apr. 2014.

[15] J. Zhang, S. Shan, M. Kan, and X. Chen, “Coarse-to-fine auto-encoder networks (CFAN) for real-time face alignment,” in *Proc. Eur. Conf. Comput. Vis.*, Zürich, Switzerland, Sep. 2014, pp. 1–16.

[16] S. Xiao, J. Feng, J. Xing, H. Lai, S. Yan, and A. Kassim, “Robust facial landmark detection via recurrent attentive-refinement networks,” in *Proc. Eur. Conf. Comput. Vis.*, Amsterdam, The Netherlands, Oct. 2016, pp. 57–72.

[17] E. Zhou, H. Fan, Z. Cao, Y. Jiang, and Q. Yin, “Extensive facial landmark localization with coarse-to-fine convolutional network cascade,” in *Proc. IEEE Int. Conf. Comput. Vis. Workshops*, Sydney, NSW, Australia, Dec. 2013, pp. 386–391.

[18] X. Wu, J. Zhou, and Y. Pan, “Initial shape pool construction for facial landmark localization under occlusion,” *IEEE Access*, vol. 5, pp. 16649–16655, 2017.

[19] Y. Sun, X. Wang, and X. Tang, “Deep convolutional network cascade for facial point detection,” in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Portland, OR, USA, Oct. 2013, pp. 3476–3483.

[20] A. Kumar, R. Ranjan, V. Patel, and R. Chellappa. (2016). “Face alignment by local deep descriptor regression.” [Online]. Available: https://arxiv.org/abs/1601.07950

[21] M. Kowalski, J. Naruniec, and T. Trzcinski, "Deep alignment network: A convolutional neural network for robust face alignment," in *Proc. Int. Conf. Comput. Vis. Pattern Recognit., Faces-Wild Workshop/Challenge (CVPRW)* Honolulu, HI, USA, Aug. 2017, pp. 88–97.

[22] J.-J. Lv, X.-H. Shao, J.-S. Huang, X.-D. Zhou, and X. Zhou, "Data augmentation for face recognition," *Neurocomputing*, vol. 230, pp. 184–196, Mar. 2017.

[23] Y. Yi, D. Qu, and F. Xu, "Face detection method based on skin color segmentation and facial component localization," in *Proc. 2nd Int. Asia Conf. Inform. Control, Automat. Robot. (CAR)*, Wuhan, China, Mar. 2010, pp. 64–67.

[24] M. Urschler, M. Storer, H. Bischof, and J. A. Birchbauer, "Robust facial component detection for face alignment applications," in *Proc. 33rd Workshop Austrian Assoc. Pattern Recognit. (AAPR/OAGM)*, At Stainz, Austria, May 2009, pp. 61–72.

[25] J. Naruniec, "Discrete area filters in accurate detection of faces and facial features," *Image Vis. Comput.*, vol. 32, no. 12, pp. 979–993, Dec. 2014.

[26] H. Fan and E. Zhou, "Approaching human level facial landmark localization by deep learning," *Image Vis. Comput.*, vol. 47, pp. 27–35, Mar. 2016.

[27] Y. Wu and Q. Ji, "Constrained joint cascade regression framework for simultaneous facial action unit recognition and facial landmark detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Las Vegas, NV, USA, Jun. 2016, pp. 3400–3408.

[28] 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.

[29] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Columbus, OH, USA, Jun. 2014, pp. 580–587.

[30] R. Girshick, "Fast R-CNN," in *Proc. IEEE Int. Conf. Comput. Vis.*, Santiago, Chile, Dec. 2015, pp. 1440–1448.

[31] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards real-time object detection with region proposal networks," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 6, pp. 1137–1149, Jun. 2017.

[32] K. Simonyan and A. Zisserman. (2014). "Very deep convolutional networks for large-scale image recognition." [Online]. Available: https://arxiv.org/abs/1409.1556

[33] K. He, G. Gkioxari, P. Dollár, and R. Girshick, "Mask R-CNN," in *Proc. IEEE Int. Conf. Comput. Vis.*, Venice, Italy, Oct. 2017, pp. 2980–2988.

[34] A. Khoreva, R. Benenson, J. Hosang, M. Hein, and B. Schiele, "Simple does it: Weakly supervised instance and semantic segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Honolulu, HI, USA, Jul. 2017, pp. 1665–1674.

[35] A. Radford, L. Metz, and S. Chintala. (2015). "Unsupervised representation learning with deep convolutional generative adversarial networks." [Online]. Available: https://arxiv.org/abs/1511.06434

[36] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Las Vegas, NV, USA, Jun. 2016, pp. 770–778.

[37] M. Lin, Q. Chen, and S. Yan. (2013). "Network in network." [Online]. Available: http://arxiv.org/abs/1312.4400

[38] O. Russakovsky *et al.*, "Imagenet large scale visual recognition challenge," *Int. J. Comput. Vis.*, vol. 115, no. 3, pp. 211–252, Dec. 2017.

[39] Y. Jia *et al.*, "Caffe: Convolutional architecture for fast feature embedding," in *Proc. 22nd ACM Int. Conf. Multimedia*, Saint Petersburg, Russia, Jul. 2014, pp. 675–678.

[40] V. Le, J. Brandt, Z. Lin, L. Bourdev, and T. S. Huang, "Interactive facial feature localization," in *Proc. Eur. Conf. Comput. Vis.*, Firenze, Italy, Oct. 2012, pp. 679–692.

[41] C. Sagonas, G. Tzimiropoulos, S. Zafeiriou, and M. Pantic, "A semi-automatic methodology for facial landmark annotation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW)*, Portland, OR, USA, Jun. 2013, pp. 896–903.

[42] X. Zhu and D. Ramanan, "Face detection, pose estimation, and landmark localization in the wild," in *Proc. IEEE Conf. IEEE Comput. Vis. Pattern Recognit. (CVPR)*, Providence, RI, USA, Jun. 2012, pp. 2879–2886.

[43] P. N. Belhumeur, D. W. Jacobs, D. J. Kriegman, and N. Kumar, "Localizing parts of faces using a consensus of exemplars," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 12, pp. 2930–2940, Dec. 2013.

[44] X. Tang, F. Guo, J. Shen, and T. Du, "Facial landmark detection by semi-supervised deep learning," *Neurocomputing*, vol. 297, pp. 22–32, Jul. 2018.

[45] C. Sagonas, G. Tzimiropoulos, S. Zafeiriou, and M. Pantic, "300 faces in-the-wild challenge: The first facial landmark localization challenge," in *Proc. IEEE Int. Conf. Comput. Vis. Workshops (ICCVW)*, Sydney, NSW, Australia, Jun. 2013, pp. 397–403.

[46] W. Liu *et al.*, "SSD: Single shot multibox detector," in *Proc. Eur. Conf. Comput. Vis.*, Amsterdam, The Netherlands, Oct. 2016, pp. 21–37.

[47] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Las Vegas, NV, USA, Jun. 2016, pp. 779–788.

[48] J. Redmon and A. Farhadi. (2016). "YOLO9000: Better, faster, stronger." [Online]. Available: http://arxiv.org/abs/1612.08242

[49] X. Xiong and F. De la Torre, "Supervised descent method and its applications to face alignment," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, Portland, OR, USA, Jun. 2013, pp. 532–539.

[50] X. Yu, J. Huang, S. Zhang, W. Yan, and D. N. Metaxas, "Pose-free facial landmark fitting via optimized part mixtures and cascaded deformable shape model," in *Proc. IEEE Int. Conf. Comput. Vis.*, Sydney, NSW, Australia, Dec. 2013, pp. 1944–1951.

[51] G. Tzimiropoulos and M. Pantic, "Gauss-Newton deformable part models for face alignment in-the-wild," in *Proc. IEEE Int. Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Columbus, OH, USA, Jun. 2014, pp. 1851–1858.

[52] S. Zhu, C. Li, C. C. Loy, and X. Tang, "Face alignment by coarse-to-fine shape searching," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Boston, MA, USA, Jun. 2015, pp. 4998–5006.

[53] Z. Zhang, P. Luo, C. C. Loy, and X. Tang, "Facial landmark detection by deep multi-task learning," in *Proc. Eur. Conf. Comput. Vis.*, Zürich, Switzerland, Sep. 2014, pp. 94–108.

[54] Z. Shao, S. Ding, Y. Zhao, Q. Zhang, and L. Ma. (2016). "Learning deep representation from coarse to fine for face alignment." [Online]. Available: http://arxiv.org/abs/1608.00207

[55] M. D. Zeiler and R. Fergus, "Visualizing and understanding convolutional networks," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, Zürich, Switzerland, Aug. 2014, pp. 818–833.

[56] J. Dai, K. He, and J. Sun, "Instance-aware semantic segmentation via multi-task network cascades," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Las Vegas, NV, USA, Jun. 2016, pp. 3150–3158.

RUIHENG ZHANG received the B.S. degree in information engineering from the Beijing Institute of Technology, China, in 2014, where he is currently pursuing the Ph.D. degree. Meanwhile, he joined the dual-Ph.D. degree with the Beijing Institute of Technology and the University of Technology Sydney, in 2017. He has authored more than six journal papers and one book chapter. His current research interests include deep learning, computer vision, and object detection.

CHENGPO MU received the B.S. degree from the Beijing Institute of Technology, the M.S. degree from Beijing Jiaotong University, and the Ph.D. degree from the Beijing Institute of Technology, where is currently an Associate Professor. He has published more than 60 journal papers and two books. His current research interests include deep learning, cyber security, and 3D simulation. He is a Guest Editor of Hindawi's *Journal of Healthcare Engineering*.

MIN XU received the B.E. degree from the University of Science and Technology of China, in 2000, the M.S. degree from the National University of Singapore, in 2004, and the Ph.D. degree from the University of Newcastle, Australia, in 2010. She is currently an Associate Professor with the University of Technology Sydney. She has published over 100 research papers in high-quality international journals and conferences. Her research interests include multimedia data analytics, pattern recognition, and computer vision.

LIXIN XU received the Ph.D. degree in information engineering from the Beijing Institute of Technology, where he is currently a Professor. He has published 100 journal and conference papers. His current research interests include deep learning, MEMS, and infrared imaging. He served as an Editor and a Reviewer for several international journals and conferences. He is on the editorial board of the *Journal of Detection and Control*.

XIAOFENG XU received the B.S. degree in computer science and technology from the Nanjing University of Science and Technology, China, in 2014, where he is currently pursuing the Ph.D. degree with the School of Computer Science and Engineering. Since 2017, he has been a Visiting Research Student with the Centre for Artificial Intelligence, University of Technology Sydney, Sydney, under the supervision of Prof. I. W. Tsang. His research interests include zero-shot learning, machine learning, and computer vision.