Multimed Tools Appl (2019) 78:13131–13148
https://doi.org/10.1007/s11042-018-5609-1

Joint face alignment and segmentation via deep multi-task learning

Yucheng Zhao[1,2] · Fan Tang[1,2] · Weiming Dong[1] · Feiyue Huang[3] · Xiaopeng Zhang[1]

Received: 31 July 2017 / Revised: 23 November 2017 / Accepted: 3 January 2018 /
Published online: 12 January 2018
© Springer Science+Business Media, LLC, part of Springer Nature 2018

Abstract Face alignment and segmentation are challenging problems which have been extensively studied in the field of multimedia. These two tasks are closely related and their learning processes are supposed to benefit each other. Hence, we present a joint multi-task learning algorithm for both face alignment and segmentation using deep convolutional neural network (CNN). The proposed multi-task learning approach allows CNN model to simultaneously share visual knowledge between different tasks. With a carefully designed *refinement residual module*, the cross-layer features are fused in a collaborative manner. To the best of our knowledge, this is the first time that face alignment and segmentation are learned together via deep multi-task learning. Our experiments show that learning these two related tasks simultaneously builds a synergy between them, improves the performance of each individual task, and rivals recent approaches. Furthermore, we demonstrate the effectiveness of our model in two practical applications: virtual makeup and face swap.

Keywords Face alignment · Face segmentation · Multi-task learning · Virtual makeup · Face swap

This work was supported by National Natural Science Foundation of China under nos. 61672520, 61702488, 61501464 and 6120106003, by Beijing Natural Science Foundation under No. 4162056, by National Key Technology R&D Program of China under No. 2015BAH53F02, and by CASIA-Tencent YouTu jointly research project.

✉ Weiming Dong
weiming.dong@ia.ac.cn

Xiaopeng Zhang
xiaopeng.zhang@ia.ac.cn

[1] NLPR-LIAMA, Institute of Automation, Chinese Academy of Sciences, Beijing, China

[2] University of Chinese Academy of Sciences, Beijing, China

[3] YouTu Lab, Tencent, Shanghai, China

1 Introduction

Facial analysis technique is frequently used in various applications such as emotion recognition [13], style transfer [9], virtual makeup [20] and face swap [17], etc. As a fundamental problem in multimedia and computer vision, a large number of processing and analysis methods have been proposed to perceive and understand multi-modal information for comprehension and understanding of face images and videos. Great progress has been achieved in facial analysis tasks such as face detection and face attributes recognition by taking advantage of recent developments in deep learning. In this paper, we generate deep representations for face alignment and face segmentation utilizing convolutional neural network.

Face alignment and segmentation are two basic tasks in facial image analysis, aiming at revealing semantic and structural information of a face image. Face alignment, also known as face landmark localization, tries to localize a set of semantic points precisely on a face image, such as eye corners, nose tip, etc. Face segmentation, which is a special case of semantic segmentation that labels each pixel in an image to a semantic class, aims to segment an input face image into semantic regions, such as nose, eyes, and lips. Although the two tasks are closely related as they all seek to extract local features on images, they are different in some aspects: face alignment focuses more on locating the positions of the organs while face segmentation cares more about the contours of the regions. In this paper, we present a novel encoder-decoder convolutional neural network for jointly learning face alignment and segmentation tasks based on the following two observations:

– Learning multiple correlated tasks simultaneously builds a synergy and improves the performance of each individual task as is shown in [28];
– Combining upsampling operation with low-level knowledge from low layers is beneficial to detecting finer structures at high resolution [21].

The proposed multi-task learning framework is shown in Fig. 1.

Most of recent deep architectures for end-to-end pixel-wise labeling contain the encoder-decoder pair to generate desired outputs. The encoder produces low resolution representation and decoder maps the low resolution representation to pixel-wise predictions. In our multi-task framework, the encoder branch is used to regress face landmarks and the decoder branch is for face segmentation. The two branches share the same features. When employing deep learning for semantic segmentation, the recognition of major objects in an image

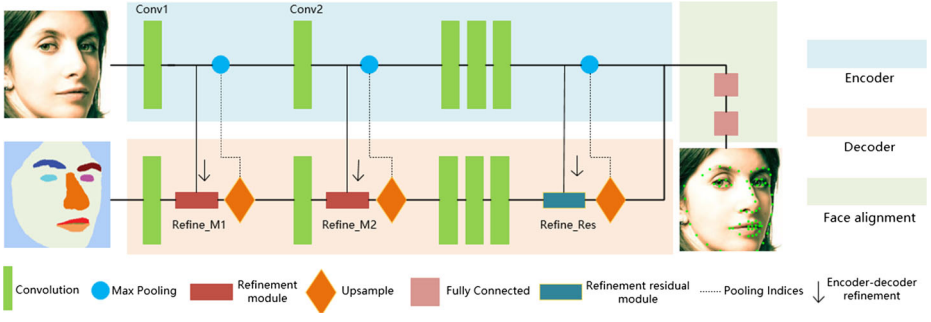


Fig. 1 Overall structure of our method. The framework contains face alignment and segmentation branches

such as face or nose region is realized at higher levels of the neural network. These high-level layers work on a coarse scale and are translation invariant (via pooling operations). However, face segmentation also require pixel-exact classification of small parts, such as eyes and lips, which typically only can be found in lower levels of the network. This trade-off is typically solved by using skip-connection from lower layers to higher layers, which increases the resolution at layers close to the output. Previous works have shown that combining high-level information comes from upsample layer and low-level knowledge from the corresponding encoder layer is helpful to detect finer structure [21]. Hence we incorporate *refinement module* proposed in [27] into our multi-task learning framework to help parse the contours of facial parts more exactly. Moreover, we propose *refinement residual module* at the joint of the two branches to let the layer fit a residual mapping. The optimization of the residual mapping is easier than that of the original, unreferenced mapping.

We evaluate the performance of face segmentation and alignment, experimental results show that the proposed model performs better than corresponding individual task. Incorporating *refinement residual module* provides additional performance boost compared to only using *refinement module*.

In summary, this paper makes the following contributions:

- To the best of our knowledge, we are the first to present a deep multi-task learning framework that solving face alignment and segmentation tasks together;
- We propose *refinement residual module* at the joint of the two tasks to enhance cross-layer feature sharing;
- We demonstrate the effectiveness of our face alignment and segmentation results in two practical applications: virtual makeup and face swap.

The rest of paper is organized as follows: in Section 2, related works on face alignment, face segmentation and multi-task learning are reviewed. In Section 3, we present our multi-task encoder-decoder framework for face alignment and segmentation in detail. Section 4 describes the implementation of the proposed approach, in which we also compare the obtained results versus the face segmentation method proposed recently. Section 5 provides two applications based on our face alignment and segmentation results, virtual makeup and face swap. Section 6 concludes the paper and explores the future works.

2 Related works

Face alignment Many approaches of face alignment have been proposed in the literature, including both regression-based [6, 36] and model-based methods [19, 24]. The former learn shape increment given a mean initial shape while the latter train an appearance model to predict positions of key points. Zhu et al. [42] presented a mixture of tree models with a shared pool of parts for landmark localization and pose estimation. Bao et al. [2, 3] used sparse representation to pre-process face images. Xiong et al. [36] proposed a supervised descent method (SDM) for face alignment.

CNN-based localization models have also been proposed in recent years and have achieved remarkable performance, but it is still a challenging problem to align face image with appearance variations in pose and expression. Sun et al. [34] used a cascade deep convolutional network based on a three-level CNN to locate facial landmarks. Zhang et al. [38] presented a coarse-to-fine auto-encoder network (CFAN) with multi-scale inputs for face alignment.

Shao et al. [31] used a coarse-to-fine training algorithm (CFT) based on CNN for face landmark localization. Sheng et al. [32] used a listwise ranking-based method to evaluate face alignment results without ground truth.

Face segmentation Due to immense variation of facial appearances in unconstrained images, it is challenging to obtain facial segmentation at pixel level. Luo et al. [22] segmented face images by a hierarchical CNN-based parsing network based on a hierarchical part detection process. The model generated masks of facial components such as eyes and mouth, but couldn't separate face region from background. Smith et al. [33] proposed an example-based approach for face region and component segmentation, but the method required sufficient visibility of face region.

Liu et al. [20] trained a face parsing model for both the before-makeup images and reference faces, but the contours of regions sometimes were not smooth.

Saito et al. [30] proposed a two-stream deconvolution network for face segmentation and used the outputs to capture 3D facial performance .

Multi-task learning Multi-task learning is a natural approach for problems that require simultaneous solutions of several related problems [7]. Existing methods either assume there is an optimal shared parameter space for all tasks and regularize the parameter space by estimating the latent relationships between tasks [41], or focus on correlating relevant features jointly [37]. Zhang et al. [40] optimized facial landmark localization with subtly related tasks such as pose estimation and facial attributes inference to improve detection robustness. Gkioxari et al. [11] proposed a CNN for the tasks of pose estimation and action classification. Rajeev et al. [28] applied deep model to jointly addressing the tasks of face detection, pose estimation, landmark localization, pose estimation and gender recognition, the method fused intermediate layers of a deep model using a separate CNN followed by a multi-task learning approach that operated on the fused features. Our work focuses on addressing face alignment and segmentation problems simultaneously by a multi-task learning mechanism.

3 Method

3.1 Overview

In order to simultaneously learn face alignment and segmentation tasks, we propose an encoder-decoder multi-task network by sharing the low-level features of the two tasks and incorporate cross-layer features fusing to help parse the contours of facial components more exactly. The overall multi-task facial analysis model is shown in Fig. 1. Unlike traditional segmentation method by using straightforward encoder-decoder mechanism, we push the encoder branch to learn low-level representations related to face landmark regression task, then decode the low-level features for face segmentation. The total loss of our multi-task deep convolutional network is computed as the weighted sum of the two individual losses as:

$$Loss_{full} = Loss_A + \lambda \cdot Loss_S, \tag{1}$$

where $Loss_A$ is the loss of face alignment, and $Loss_S$ is the loss of face segmentation. The weight λ is decided according to the importance of tasks in the overall loss. We set $\lambda = 5$ in our experiments.

In the following subsections, we start with the encoder branch of face landmark regression, then move to face segmentation decoder branch, and finally expose the encoder-decoder refinement trade-offs.

3.2 Encode face alignment

We add face landmark regression task into the encoder-decoder pipeline to stimulate the encoder branch to encode face representations more effectively. The fully connected layers are added after the low-level representations and their outputs predict face landmark coordinates.

Architecture The encoder network is illustrated in Fig. 1 (*Encoder* and *Face Alignment* module). It consists of eight learned layers, the first five are convolutional and the remaining three are fully connected. The first convolutional layer filters the $224 \times 224 \times 3$ input image with 96 kernels of size 7×7 with a stride of 2 pixels. The second convolutional layer takes as input the output of first convolutional layer and filters it with 256 kernels of size 5×5. The third, fourth and fifth convolutional layers are connected to one another, each of which have 512 kernels of size 3×3 with a pad of 1 pixel. The fully connected layers have 4096 neurons each. We reduce the kernel size of the first convolutional layer and stride in the first two convolutional layers to reduce information loss. Each convolution in the encoder network produces a set of feature maps. Batch normalized [15] and an element-wise rectified-linear non-linearity(ReLU) is applied. Following that, max-pooling is used to achieve translation invariance over small spatial shifts of input image.

Encoder loss We use face landmark regression to regularize the encoder process. A face region can be characterized by x, y, w, h where (x, y) is the coordinate of center of the region and w, h are width and height of the region respectively. Each landmark point is shifted with respect to the region center and normalized by $(w/2, h/2)$ as:

$$(a_i, b_i) = (2 \cdot \frac{x_i - x}{w}, 2 \cdot \frac{y_i - y}{h}), \tag{2}$$

where (a_i, b_i) is treated as labels for training the landmark localization task using euclidean loss as:

$$Loss_A = \frac{1}{2N} \sum_{i=1}^{N} ((\hat{x_i} - a_i)^2 + (\hat{y_i} - b_i)^2), \tag{3}$$

where $(\hat{x_i}, \hat{y_i})$ is the i^{th} landmark location predicted by the network and N is the total number of the landmark.

3.3 Decode face segmentation

In this part, we describe the decoder module for face segmentation generation. We decode the low-level representations by upsample operations and supply a prediction of each pixel for the original input.

Architecture The decoder module of our network is shown in Fig. 1 (*Decoder* module). The role of decoder branch is to map the low resolution encoder feature maps to full input resolution feature maps which produce class probability for each pixel independently. Each encoder layer has a corresponding decoder layer.

We follow the work of Badrinarayanan et al. [1] to upsample the low resolution feature maps. Each decoder part upsamples its input using the transferred pooling indices from the corresponding encoder part to produce sparse feature maps, as depicted in Fig. 1 (*Upsample* module). As the convolutional operations in *Conv1* and *Conv2* layers downsample the feature maps, we upsample the corresponding decoder part by learning to deconvolve the input feature maps to produce the decoder outputs.

Decoder loss We use softmax loss as the objective loss function:

$$Loss_S(C, C^*) = -\frac{1}{n}\sum_i C_i^* log(C_i), \tag{4}$$

where $C_i = e^{z_i} / \sum_c e^{z_{i,c}}$ is the class prediction at pixel i given the output of the final convolutional layer. The loss is summed up over all pixels in a mini-batch. As there is large variation in the number of pixels in each class of the training set, it is need to weight the loss differently on each class. We use median frequency balancing [8] and weight each pixel by $\alpha_c = median_freq/freq(c)$, where $freq(c)$ is the number of pixels in class c divided by the total number of pixels in images, and $median_freq$ is the median of these frequencies. The strategy implies that large classes in the training set usually have smaller weights than those of small classes.

3.4 Encoder-decoder refinement

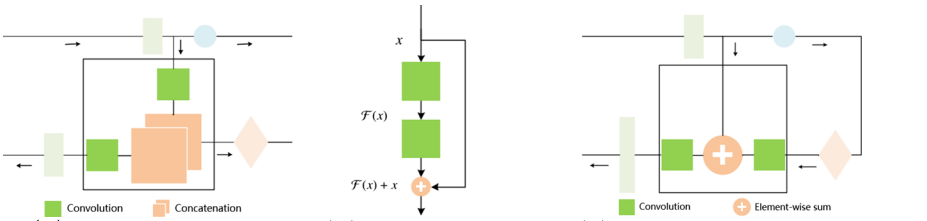
Although the two proposed losses are both considered when constructing the learning pipeline, the face alignment loss only influences the encoder branch when using back-propagating to update parameters. Hence, we enhance the decoder branch by adding additional connections to utilize information from the encoder branch (lines with arrow in Fig. 1).

Firstly, we give the content of *refinement* module in [27] and then we propose the *refinement residual* module.

Refinement module Options such as pooling or convolution in encoder may reduce the resolution of input, which is not beneficial for segmentation where boundary delineation is vital. Previous works have shown that using not only the information comes directly from the upsample layer, but also combines it with the low-level knowledge from corresponding low layer of the encoder is helpful to detect fine structure at high resolution. The *refinement module* is proposed in [27] to generate high-fidelity object masks and we add it in both *conv1* and *conv2* convolutional layers.

As is shown in Fig. 2a, each refinement module combines two streams as input. One is from previous upsample layer and the other is from the corresponding encoder convolutional layer. The two convolutional operations are used to transform the features from different layers so that they can be concatenated together. The kernel size of convolutional operation in refinement module is 3×3 with one-pixel padding.

Refinement residual module The refinement module enhances information sharing among different branches. Unlike pervious works focusing on single image-to-image learning task, we learn two related tasks simultaneously. The vital difference between our network and other encoder-decoder pipelines is that we add additional goals to the end of the encoder branch. To promote further feature sharing and boost the multi-task learning objective, we propose *refinement residual* module, as is shown in Fig. 2c.



(a) Refinement module in [27]. (b) Residual module in [14]. (c) Refinement residual module we proposed.

Fig. 2 Different refinement strategies

By incorporating refinement residual module, we aim to add strong connection at the joint of the two branches. Similar to refinement module, refinement residual module also takes two inputs: one is from the previous convolutional layer in encoder branch and the other from the first upsample layer in decoder. Differently, our refinement residual module is designed by taking residual learning into consideration. Fig. 2b illustrates the content of the basic residual module in [14]. Supposing the input signal is donated as x and the desired underlying mapping as $\mathcal{H}(x)$, they let the stacked nonlinear layers fit a residual mapping $\mathcal{F}(x) = \mathcal{H}(x) - x$ by adding *shortcut connections*. As for our refinement residual module, we also use shortcut connection to identify the output of the last convolutional layer in encoder branch, and use element-wise addition to realize the operation $\mathcal{F} + x$. Furthermore, a convolutional layer is adopted to transform feature maps before passing information to next layer. The kernel size of convolutional operation in our refinement residual module is 3×3 padded with one pixel. The reason of adopting such module is that we want the face alignment branch to push more influence on the decoder branch. By stack of layers and operations, this enhanced module is expected to transform the face alignment loss put on the encoder branch to the decoder branch.

Different from refinement module, the features from shallow layers are directly connected to the refinement residual module without additional convolutional operation. The reason is that the refinement residual module is located at the joint of the two branches, where the two inputs are quite similar. Furthermore, we use *sum* operation in refinement residual module instead of *concat* operation because it is easier to optimize the residual mapping in deep layers.

4 Implementation and experiment

We evaluate our method and show that learning face alignment and segmentation simultaneously promotes the performance of individual task and clearly outperforms a recent face segmentation approach [30].

4.1 Implementation Details

Pre-training We use AFLW [18] dataset to train the face detection network, the structure of which is the same as encoder face alignment. Once trained, the learned parameters are used to initialize the encoder face alignment branch, decoder face segmentation and the multi-task framework, as the standard ImageNet initialization does not converge well. The selective search algorithm [35] is used to generate region proposals for faces in an image.

A region overlapping more than 0.5 area of the ground truth bounding box is considered as a positive sample ($l = 1$), while the candidate regions overlapping less than 0.35 area of the ground truth bounding boxes are treated as negative instances ($l = 0$). All other regions are ignored. In our experiments, we randomly select 50K positive and negative samples as the training dataset. Softmax loss is used to train the network:

$$Loss_D = -(1 - l) \cdot log(1 - p) - l \cdot log(p), \tag{5}$$

where p is the probability that the candidate region is a face.

Datasets We train the multi-task network using HELEN dataset [33], which contains 2330 face images with manually-annotated labels including eye (L/R), eyebrow (L/R), nose, lip (Up/Low), foundation, inner mouth and background, where "L", "R", "Up" and "Low" stand for "Left","Right","Upper" and "Lower" respectively. The dataset also contains landmark annotations following the Multi-PIE [12] 68 points mark-up, which is used for training face alignment task. The test set of face alignment task is composed of test set in LFPW [4] and HELEN, where are totally 554 images in general.

To prevent overfitting, we augment the HELEN dataset using random perturbations of translation, rotation. Skin toned objects such as hands and arms are commonly observed during gesticulations, which are difficult to parse because of the similar colors and fine structures with face, we further augment the training dataset with composited hands and arms. We first collect 500 hands images under different poses and light conditions and manually segment them. We then synthesize these hands images on top of the face images. In total, there are 31803 images used to train our network.

Training details We implement the multi-task network using Caffe framework [16]. Once trained, the learned Caffe model and corresponding deploy file are used for testing. We adopt MTCNN [39] method for face detection, the detected face is then resized to 224×224 as the input of the multi-task framework for training and testing. We train our models using stochastic gradient descent (SGD) with a batch size of 80 examples, momentum of 0.9, and weight decay of 0.0002. We use "Poly" learning policy and set the base learning rate to 10^{-3}. The model is trained on NVIDIA Titan X GPUs. We set the weighted parameter $\lambda = 5$, which achieves the best result on validation dataset. We will analyze the choice of λ to the performance of multi-task learning in evaluation part (Section 4.2).

4.2 Evaluation of face segmentation

We rely on mean intersection-over-union metric (mIoU) [10] as used in [30] to assess the performance of face segmentation, which is simply calculated as the average IoU over all classes:

$$IoU = \frac{TP}{TP + FP + FN},$$

where TP, FP and FN are the numbers of true positive, false positive and false negative pixels respectively, determined over the whole test set. The mIoU is a more stringent metric than class average accuracy since it penalizes false positive predictions. In our evaluation, we firstly qualify the effect of different refinement strategies on face segmentation performance, then compare our result with a recently proposed face segmentation approach [30].

Evaluation of decode refinement In Table 1, we show the qualitative results about the impact of different refinement strategies to face segmentation performance. FaceSeg-Basic is a simple encoder-decoder network without refinement module used in the network. FaceSeg-Refine_M1-M2 is constructed by adding Refine_M1 and Refine_M2 module to FaceSeg-Basic. FaceSeg-Refine_Res is constructed by adding Refine_Res to FaceSeg-Refine_M1-M2, as shown in Fig. 1 (decoder part). FaceSeg-Refine_Basic is similar to FaceSeg-Refine_Res but we replace the third refinement residual module to the form of Refine_M1. Figure 3 shows the face segmentation results among different refinement strategies. We observe that FaceSeg-Basic without any refinement strategy performs the worst and does not retain small classes. All other methods which add refinement module perform significantly better. This emphasizes the significance of refining the segmentation contours of small parts such as eyes and lips, which is achieved by combining information presented in lower layers in encoder and corresponding decoder part. Particularly, FaceSeg-Refine_Res is superior to FaceSeg-Refine_Basic, which validates the effectiveness of the proposed refinement residual module. The residual module enhances more impact on the decoder branch from encoder face alignment part.

Following we use FaceSeg-Refine_Res network as the decoder branch to train our multi-task encoder-decoder network as is shown in Fig. 1.

Evaluation of multi-task framework We compare our multi-task face segmentation results with the results generated by the Two-Stream-ConvNet model (re-tranied using our traning set) in [30]. The qualitative results are shown in the last two rows of Table 1 and apparently our network outperforms the Two-Stream-ConvNet model. To be fair, we also compare the facial region IoU in terms of the segmentation result with Two-Stream-ConvNet, which is designed to segment the whole facial region. Our result (86.9) outperforms the result reported in [30] (81.4). Figure 4 shows comparison of the segmentation results between the two methods, which shows the ability of our proposed multi-task network to parse small classes more precisely and produce a smoother segmentation contours.

4.3 Evaluation of face alignment

The evaluation metric of face alignment is the mean error of facial landmarks, which is calculated as:

$$\frac{1}{n}\sum_{i=1}^{n}\frac{\|x_i - x_i^*\|^2}{d},$$

where d denotes the inter-pupil distance, which is estimated by averaging the coordinates of the landmarks around eyes, and n is the number of facial landmarks.

Table 1 Quantitative comparison of face segmentation performance

Network	mIoU	face IoU
FaceSeg-Basic	20.5	81.0
FaceSeg-Refine_M1-M2	65.3	84.2
FaceSeg-Refine_Basic	65.0	84.2
FaceSeg-Refine_Res	67.5	85.6
Face-Multi_task	70.2	86.9
Two-Stream-ConvNet [30]	61.4	81.4

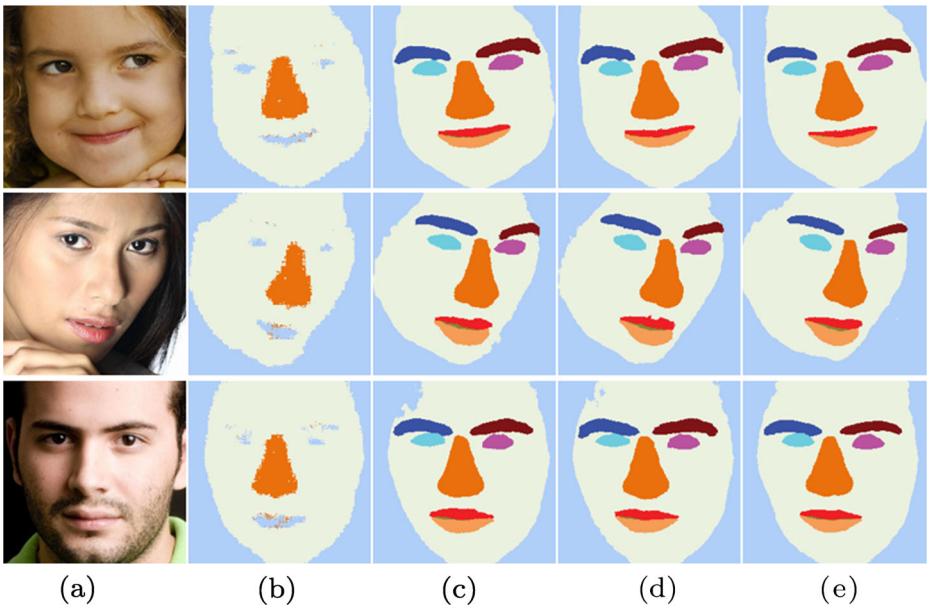


Fig. 3 Comparison of face segmentation results among different refinement strategies. The **a** input image, **b** the results segmented by FaceSeg-Basic, **c** by FaceSeg-Refine_M1-M2, **d** by FaceSeg-Refine_Basic and **e** by FaceSeg-Refine_Res network are shown sequentially

Evaluation of face alignment We compare the face landmark localization result between the encoder face alignment and the multi-task framework. Other state-of-the-art methods used for comparison include Zhu et al. [42], SDM [36], CFAN [38] and CFT [31]. The results are shown in Table 2, which indicates the proposed multi-task framework is superior

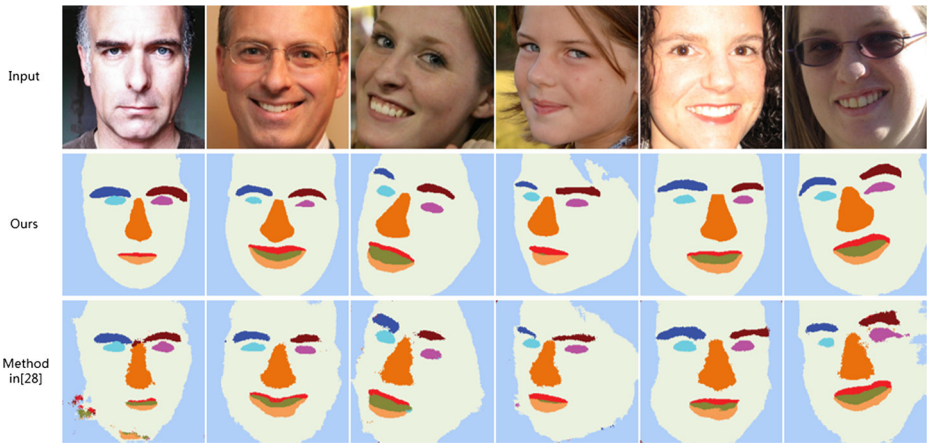


Fig. 4 Comparison of face segmentation results between our method and the method proposed in [30]. The images in the first two columns are near frontal, images in the middle two columns are large pose variation and the last two columns are large illumination variation. Our method is more robust to different situations and the segmentation contours are smoother than the model in [30]

Table 2 Quantitative comparison of face alignment performance with state-of-art methods

Method	mean Error
Zhu et al. [42]	8.22
SDM [36]	5.57
CFAN [38]	5.50
Our face-encoder-align	6.33
Our face-multi-task	5.10
CFT [31]	4.82

to the encoder face alignment network (mean error 5.10 vs. 6.33), method in [42], SDM and CFAN. Our multi-task network is slightly inferior to CFT that adds supervisory signal to each max-pooling layer, which makes the model more complex than ours. On the other side, CFT method takes about 65ms to process an image on an Intel Core i7 CPU, while our encoder face alignment model only takes 23ms, which is 3 times faster. The CFT network is more complex than our face alignment network and thus needs more time for one forward propagation. In detail, the CFT network contains eight convolutional layers and two fully connected layers, while our face alignment network contains five convolutional layers and three fully connected layers. Figure 5 shows the face landmark localization results. We can see that the proposed multi-task network has a better generalization ability than the encoder network and can correct the position of landmarks.

We also demonstrate the successful face alignment and segmentation results with complex occlusions such as hands and arms as is shown in Fig. 6. Our convolutional network effectively predicts a dense probability map revealing the alignment and segmentation results, even when the faces are blocked by objects with similar skin tones such as hands.

Analysis of parameter We analyze the face alignment and segmentation results in terms of weighted parameter λ and the results are shown in Table 3. We use mIoU as the evaluation metric of face segmentation and use mean Error for face alignment. A larger λ is beneficial for good segmentation result, which is in accordance with face alignment. The best performance is achieved when $\lambda = 5$.

5 Application

Virtual makeup Virtual makeup is a technique used to change people's appearance with special cosmetics such as foundation, lip gloss and eye shadow. In most cases, especially for females, makeup makes people more attractive and gain self-confidence. There are

(a) Multi-task network (b) Encoder network

Fig. 5 Comparison of face alignment results between the encoder face alignment and multi-task network

Fig. 6 Examples of face alignment and segmentation results with complex occlusions such as hands and arms

commercial makeup systems in market increasingly. Virtual makeup website TAAZ[1] offers some pre-determined cosmetic elements such as lipsticks and eye shadow. Face alignment and segmentation are key techniques for digital face makeup, which help synthesize the final visual effect of makeup for given images.

Based on the face segmentation result, only the eye shadow mask need to be aligned to eye region in the test image. To illustrate this clearly, the mask is firstly scaled and rotated according to the landmark in eye corner predicted by the encoder face alignment branch. Then, Thin-Plate-Spline warpping method [5] is used to adjust the mask. Color transfer [29] is used to change the appearance of the lip region in source (before-makeup) image according to the color pattern of a target (after-makeup) image. The final synthesis result is generated by Alpha blending of testing image I and template T of lip and eye shadow to make the makeup color natural, which is given as:

$$\gamma(x, y) = \alpha \cdot I + (1 - \alpha) \cdot T,$$

where $\alpha \in [0, 1]$ and is set to 0.6 in our experiments.

We compare our virtual makeup results with the results of TAAZ in Fig. 7. We can see the significance of accurate face alignment and segmentation results to the elegant virtual makeup. The contrast of lip region is preserved by using color transfer method. Moreover, our makeup synthesis process is fully automatic but TAZZ requires manual intervention.

Face swap Swapping face means transferring a face from a source image onto a face appearing in a target image, attempting to generate realistic results. It is relevant in many scenarios including digital forensics [26], privacy preservation [25] and as a potential face specific data augmentation method [23].

[1] http://www.taaz.com/

Table 3 Comparison of face segmentation and alignment results in terms of weighted parameter λ

λ	0.2	0.4	1	3	5	7
mIoU	67.3	68.5	68.7	70.1	70.2	70.2
mean Error	5.27	5.32	5.36	5.14	5.10	5.13

Based on the face alignment result, we first calculate the affine transform matrix between the source image and target image so that:

$$\begin{bmatrix} \hat{x_i} \\ \hat{y_i} \end{bmatrix} = M \cdot \begin{bmatrix} x_i \\ y_i \\ 1 \end{bmatrix},$$

where $[\hat{x_i}, \hat{y_i}]^T$ is the coordinates in target image and $[x_i, y_i]^T$ is the corresponding coordinates in source image, M is the calculated affine transform matrix. Then, we move the source image face to the target based on the affine transform matrix. We also transform the color of the source face by normalizing cumulative distribution function (CDF) and create blur effect by box filter.

Fig. 7 Comparison of our virtual makeup results with the results of TAAZ. For each row, from left to right: before-makeup image, our results, TAAZ's results, local details of our results, local details of TAAZ's results

Fig. 8 Face swap results. The source faces are accurately mapped to the targets by using face alignment results

We show face swap results in Fig. 8. Our method is able to introduce noticeable changes to the appearance of the source faces while keeping head pose and gaze direction, and produce plausible face-swapped images.

6 Conclusion

In this paper, we propose a deep encoder-decoder neural network for jointly learning face alignment and segmentation tasks simultaneously via multi-task learning. We find a few crucial observations from our experiments. Firstly, the related tasks of face alignment and segmentation benefit from each other by using multi-task framework, which mainly due to the network's ability to learn more discriminative features. Secondly, combining low-level features from encoder part is responsible for parsing finer contours, as the features in CNN is hierarchically distributed and lower layers features contain edge information. Extensive experiments demonstrate the effectiveness of our method for the two tasks. For future work, we will generalize our method to video and other applications such as simultaneous face detection and pose estimation.

Acknowledgements The Titan X used for this research was donated by the NVIDIA Corporation.

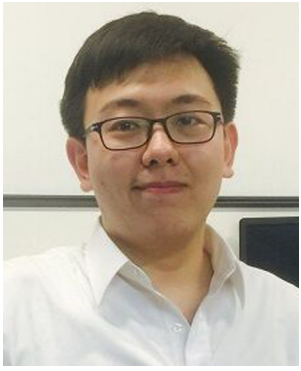
References

1. Badrinarayanan V, Kendall A, Cipolla R (2015) Segnet: a deep convolutional encoder-decoder architecture for image segmentation. arXiv:1511.00561
2. Bao BK, Liu G, Xu C, Yan S (2012) Inductive robust principal component analysis. IEEE Trans Image Process 21(8):3794–3800
3. Bao BK, Zhu G, Shen J, Yan S (2013) Robust image analysis with sparse representation on quantized visual features. IEEE Trans Image Process 22(3):860–871
4. Belhumeur PN, Jacobs DW, Kriegman DJ, Kumar N (2013) Localizing parts of faces using a consensus of exemplars. IEEE Trans Pattern Anal Mach Intell 35(12):2930–2940
5. Bookstein FL (1989) Principal warps: thin-plate splines and the decomposition of deformations. IEEE Trans Pattern Anal Mach Intell 11(6):567–585

6. Cao X, Wei Y, Wen F, Sun J (2014) Face alignment by explicit shape regression. Int J Comput Vis 107(2):177–190
7. Caruana R (1998) Multitask learning. In: Learning to learn. Springer, pp 95–133
8. Eigen D, Fergus R (2015) Predicting depth, surface normals and semantic labels with a common multi-scale convolutional architecture. In: Proceedings of the IEEE international conference on computer vision, pp 2650–2658
9. Elad M, Milanfar P (2017) Style transfer via texture synthesis. IEEE Trans Image Process 26(5):2338–2351
10. Everingham M, Eslami SA, Van Gool L, Williams CK, Winn J, Zisserman A (2015) The pascal visual object classes challenge: a retrospective. Int J Comput Vis 111(1):98–136
11. Gkioxari G, Hariharan B, Girshick R, Malik J (2014) R-CNNs for pose estimation and action detection. arXiv preprint. arXiv:1406.5212
12. Gross R, Matthews I, Cohn J, Kanade T, Baker S (2010) Multi-pie. Image Vis Comput 28(5):807–813
13. Happy S, Routray A (2015) Automatic facial expression recognition using features of salient facial patches. IEEE Trans Affect Comput 6(1):1–12
14. He K, Zhang X, Ren S, Sun J (2016) Deep residual learning for image recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 770–778
15. Ioffe S, Szegedy C (2015) Batch normalization: accelerating deep network training by reducing internal covariate shift. In: Proceedings of the 32nd international conference on machine learning, pp 448–456
16. Jia Y, Shelhamer E, Donahue J, Karayev S, Long J, Girshick R, Guadarrama S, Darrell T (2014) Caffe: Convolutional architecture for fast feature embedding. In: Proceedings of the 22nd ACM international conference on multimedia. ACM, pp 675–678
17. Korshunova I, Shi W, Dambre J, Theis L (2016) Fast face-swap using convolutional neural networks. arXiv:1611.09577
18. Köstinger M, Wohlhart P, Roth PM, Bischof H (2011) Annotated facial landmarks in the wild: a large-scale, real-world database for facial landmark localization. In: IEEE international conference on computer vision workshops (ICCV workshops), pp 2144–2151. https://doi.org/10.1109/ICCVW.2011.6130513
19. Liang L, Xiao R, Wen F, Sun J (2008) Face alignment via component-based discriminative search. In: European conference on computer vision. Springer International Publishing, pp 72–85
20. Liu S, Ou X, Qian R, Wang W, Cao X (2016) Makeup like a superstar: deep localized makeup transfer network. In: 25th international joint conference on artificial intelligence, IJCAI2016
21. Long J, Shelhamer E, Darrell T (2015) Fully convolutional networks for semantic segmentation. In: IEEE conference on computer vision and pattern recognition (CVPR), pp 3431–3440
22. Luo P, Wang X, Tang X (2012) Hierarchical face parsing via deep learning. In: IEEE conference on computer vision and pattern recognition, pp 2480–2487
23. Masi I, Trần AT, Hassner T, Leksut JT, Medioni G (2016) Do we really need to collect millions of faces for effective face recognition? In: European conference on computer vision. Springer, pp 579–596
24. Matthews I, Baker S (2004) Active appearance models revisited. Int J Comput Vis 60(2):135–164
25. Mosaddegh S, Simon L, Jurie F (2014) Photorealistic face de-identification by aggregating donors' face components. In: Asian conference on computer vision. Springer, pp 159–174
26. Oikawa MA, Dias Z, de Rezende Rocha A, Goldenstein S (2016) Manifold learning and spectral clustering for image phylogeny forests. IEEE Trans Inf Forensics Secur 11(1):5–18
27. Pinheiro PO, Lin TY, Collobert R, Dollár P (2016) Learning to refine object segments. In: European conference on computer vision. Springer, pp 75–91
28. Ranjan R, Patel VM, Chellappa R (2016) Hyperface: a deep multi-task learning framework for face detection, landmark localization, pose estimation, and gender recognition. arXiv:1603.01249
29. Reinhard E, Adhikhmin M, Gooch B, Shirley P (2001) Color transfer between images. IEEE Comput Graph Appl 21(5):34–41
30. Saito S, Li T, Li H (2016) Real-time facial segmentation and performance capture from rgb input. In: European conference on computer vision. Springer International Publishing, pp 244–261
31. Shao Z, Ding S, Zhao Y, Zhang Q, Ma L (2016) Learning deep representation from coarse to fine for face alignment. In: IEEE international conference on multimedia and expo
32. Sheng K, Dong W, Kong Y, Mei X, Li J, Wang C, Huang F, Hu BG (2015) Evaluating the quality of face alignment without ground truth. Comput Graphics Forum 34(7):213–223
33. Smith BM, Zhang L, Brandt J, Lin Z, Yang J (2013) Exemplar-based face parsing. In: IEEE conference on computer vision and pattern recognition (CVPR), pp 3484–3491
34. Sun Y, Wang X, Tang X (2013) Deep convolutional network cascade for facial point detection. In: IEEE conference on cmputer vision and pattern recognition (CVPR), pp 3476–3483

35. Van de Sande KE, Uijlings JR, Gevers T, Smeulders AW (2011) Segmentation as selective search for object recognition. In: IEEE international conference on computer vision (ICCV). IEEE, pp 1879–1886
36. Xiong X, De la Torre F (2013) Supervised descent method and its applications to face alignment. In: IEEE conference on computer vision and pattern recognition (CVPR), pp 532–539
37. Yang Y, Hospedales TM (2014) A unified perspective on multi-domain and multi-task learning. arXiv:1412.7489
38. Zhang J, Shan S, Kan M, Chen X (2014) Coarse-to-fine auto-encoder networks (CFAN) for real-time face alignment. In: European conference on computer vision. Springer International Publishing, pp 1–16
39. Zhang K, Zhang Z, Li Z, Qiao Y (2016) Joint face detection and alignment using multitask cascaded convolutional networks. IEEE Signal Process Lett 23(10):1499–1503
40. Zhang Z, Luo P, Loy CC, Tang X (2014) Facial landmark detection by deep multi-task learning. In: European conference on computer vision. Springer, pp 94–108
41. Zhou J, Chen J, Ye J (2011) Clustered multi-task learning via alternating structure optimization. In: Advances in neural information processing systems, pp 702–710
42. Zhu X, Ramanan D (2012) Face detection, pose estimation, and landmark localization in the wild. In: IEEE conference on computer vision and pattern recognition (CVPR). IEEE Computer Society, Washington, DC, pp 2879–2886

Yucheng Zhao received the BSc degree in Automation from Shanghai Jiaotong University in 2015. He is currently working toward the MEng degree in National Laboratory of Pattern Recognition, Institute of Automation, Chinese Academy of Sciences. His research interests include face analysis, deep learning.



Fan Tang received the BSc degree in computer science from North China Electric Power University in 2013. He is currently working toward the PhD degree in National Laboratory of Pattern Recognition, Institute of Automation, Chinese Academy of Sciences. His research interests include computer graphics, computer vision and machine learning.

Weiming Dong is a Professor in the Sino-European Lab in Computer Science, Automation and Applied Mathematics (LIAMA) and National Laboratory of Pattern Recognition (NLPR) at Institute of Automation, Chinese Academy of Sciences. He received his BSc and MSc degrees in Computer Science in 2001 and 2004, both from Tsinghua University, China. He received his PhD in Computer Science from the University of Lorraine, France, in 2007. His research interests include image synthesis and image recognition. Weiming Dong is a member of the ACM and IEEE.

Feiyue Huang is the director of Social Network Platform Department, Tencent. He received his BSc and PhD degrees in Computer Science in 2001 and 2008, both from Tsinghua University, China. His research interests include image understanding and face recognition.

Xiaopeng Zhang received his MSc degree in Mathematics from Northwest University in 1987, and the PhD degree in Computer Science from Institute of Software, Chinese Academy of Sciences (CAS), in 1999. He is a Professor in the Sino-European Lab in Computer Science, Automation and Applied Mathematics (LIAMA) and (National Laboratory of Pattern Recognition) at Institute of Automation, Chinese Academy of Sciences (CASIA). His main research interests are computer graphics and pattern recognition. He received the National Scientific and Technological Progress Prize (Second Class) in 2004. Xiaopeng Zhang is a member of ACM and IEEE.