Lightweight Real-time Makeup Try-on in Mobile Browsers with Tiny CNN Models for Facial Tracking

TianXing Li *†
ModiFace Inc.
tianxingli24@gmail.com

Zhi Yu *
ModiFace Inc.
vicky@modiface.com

Edmund Phung
ModiFace Inc.
edmund@modiface.com

Brendan Duke
ModiFace Inc.
brendan@modiface.com

Irina Kezele
ModiFace Inc.
irina@modiface.com

Parham Aarabi
ModiFace Inc.
parham@modiface.com

Abstract

Recent works on convolutional neural networks (CNNs) for facial alignment have demonstrated unprecedented accuracy on a variety of large, publicly available datasets. However, the developed models are often both cumbersome and computationally expensive, and are not adapted to applications on resource restricted devices. In this work, we look into developing and training compact facial alignment models that feature fast inference speed and small deployment size, making them suitable for applications on the aforementioned category of devices. Our main contribution lies in designing such small models while maintaining high accuracy of facial alignment. The models we propose make use of light CNN architectures adapted to the facial alignment problem for accurate two-stage prediction of facial landmark coordinates from low-resolution output heatmaps. We further combine the developed facial tracker with a rendering method, and build a real-time makeup try-on demo that runs client-side in smartphone Web browsers. We prepared a demo link to our Web demo that can be tested in Chrome and Firefox on Android, or in Safari on iOS: `https://s3.amazonaws.com/makeup-paper-demo/index.html`[1] [2]

1. Introduction

Facial alignment is a key component of virtual makeup try-on methods. These methods require very high accuracy in facial alignment, as any error in makeup placement would result in a poor user experience. The alignment also needs to be robust to variations in lighting, pose, face shape, and skin tone. It is particularly important that the landmarks are precisely aligned for frontal and close to frontal face poses, as those are common in virtual try-on applications.

In the context of real-time applications on resource constrained platforms such as mobile and Web, we need to address the requirements for both low computational demands and small model size. The latter is of particular importance for client-side Web applications where long loading times are not ideal.

The state-of-the-art facial alignment architectures [10][19][22] have not been developed for the purpose of real-time inference on resource restricted devices. However, our applications primarily target such devices. To strike a better balance for real-time applications on those platforms, the ideal architecture should minimize load and execution time while preserving alignment accuracy.

Our proposed architecture does the following to meet these requirements: its first stage makes coarse initial predictions, from which crops of shared convolutional features are taken; these Regions of Interest (RoIs) are then processed by the second stage to produce more spatially refined predictions. Besides the coarse-to-fine alignment approach, we also balance the number of layers and the resolution of feature maps, and achieve fine-level alignment with high computational efficiency, while maintaining a small model size. The resulting architecture is suitable for real-time Web applications in mobile browsers.

2. Related Work

2.1. Facial Landmark Alignment

The facial landmark alignment problem has a long history with classical computer vision solutions. For instance, the fast ensemble tree based [8] algorithm achieves reasonable accuracy and is widely used for real-time face tracking [9]. However, the model size required to achieve such ac-

[*] These two authors contributed equally.

[†] The work performed during internship at ModiFace Inc.

[1] The link works in listed browsers on corresponding devices, and should start with “https://”.

[2] More results and demo videos are in project page: `http://research.modiface.com/makeup-try-on-cvprw2019/`

curacy is prohibitively large.

Current state-of-the-art accuracy for facial landmark alignment are achieved by convolutional neural network based methods. To maximize accuracy on challenging datasets [1][11][17], they use large neural networks that are not real-time, and have model sizes of tens to hundreds of MB [22][13] that would entail unreasonable load times for Web applications.

2.2. Efficient CNN Architectures

To bring the performance of convolutional neural networks to mobile vision applications, numerous architectures with efficient building blocks such as MobileNetV2 [18], SqueezeNet [7] and ShuffleNet [24] have recently been released. These networks aim to maximize performance (e.g., classification accuracy) for a given computational budget, which consists of the number of required learnable parameters (the model size) and multiply-adds.

We will focus our discussion on the state-of-the-art MobileNetV2, whose inverted residual blocks are used in our own design. Their choice of depthwise convolutions over regular convolutions drastically reduces the number of multiply-adds and learnable parameters, at a slight cost in performance [18]. Furthermore, the inverted design, which is based upon the principle that network expressiveness can be separated from capacity, allows for a large reduction in the number of cross-channel computations within the network [18].

Finally, the residual design similar to ResNet [6] eases issues with gradient propagation in deeper networks.

2.3. Heatmap

Fully convolutional neural network architectures based on heatmap regression [2][20][3][14] have been widely used on human pose estimation tasks. The use of heatmaps provides a high degree of accuracy, along with an intuitive means of seeing the network's understanding and confidence of landmark regression. This technique has also been used in recent facial alignment algorithms such as the Stacked Hourglass architecture [22]. However, the Stacked Hourglass approach [22] uses high resolution heatmaps, which require a large amount of computation in the decoding layers. There is room for optimization here, as the heatmaps only have non-negligible values in a very concentrated and small portion of the overall image. This observation motivates us to use regional processing, which allows for the network to focus its processing in relevant areas. (i.e., the approximate Rregion of Interest).

2.4. Mask RCNN

There are a series of frameworks which are flexible and robust to object detection and semantic segmentation like Fast R-CNN [4], Faster R-CNN [15] and Fully Convolutional Network [12]. Faster R-CNN proposes a multi-branch design to perform bounding box regression and classification in parallel. Mask-RCNN [5] is an extension of Faster-RCNN, and adds a new branch for predicting segmentation masks based on each Region of Interest. Of particular interest is Mask-RCNN's use of RoI align [5], which allows for significant savings in computation time by taking crops from shared convolutional features. By doing this, it avoids recomputing features for overlapping Regions of Interest.

3. Proposed Method

There are two main contributions of our model:

1. We use RoI align [5] for each individual landmark to save potentially overlapping computation, allow the network to avoid non-relevant regions, and force the network to learn to produce useful shared features.
2. Our two-stage localization architecture along with auxiliary coordinate regression loss allow us to work with extremely small and computationally cheap heatmaps at both stages.

3.1. Model Structure

The model has two-stages and is trained end-to-end, as illustrated in Figure 1.

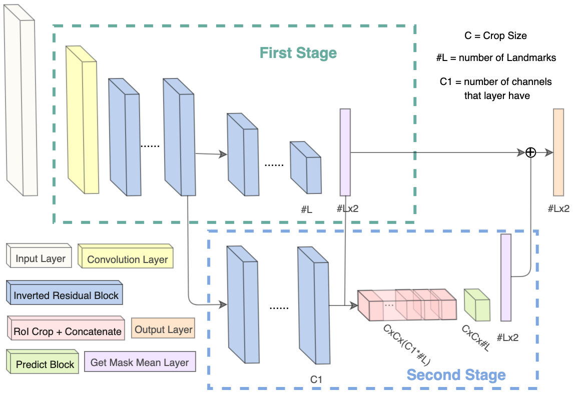

Figure 1. Our two-stages Network Architecture.

The first stage is formed by a list of Inverted Residual Blocks [18], which predict $H \times H$ heatmaps, one for each facial landmark. Interpreting the normalized activations over the heatmaps as a probability distribution, we compute the expected values of these heatmaps to obtain the x, y coordinates. This will be described in more detail in the following subsection.

The second stage has several shared convolutional layers, which branch off from part of the first stage. Using the coarse predictions from the previous stage, we apply RoI align [5] to the final shared convolutional features. Each of the cropped features are input to one final convolutional layer, which has separate weights for each individual landmark. Our predict block implements this efficiently with the use of group convolutions [24]. The final output is a heatmap for each landmark. The coordinates obtained from these heatmaps indicate the required offset from the initial coarse prediction, i.e., if the heatmap at this stage is perfectly centered, then there is effectively no refinement applied to the first stage prediction.

3.2. Coordinate Regression from Heatmaps

For our ground truth heatmaps, we use a Gaussian distribution with a mode corresponding to the ground truth coordinates' positions. Letting x, y denote the coordinates of any pixel in the feature map, we can compute its value using a $2D$ Gaussian distribution with the corresponding landmark coordinate as center.

The regressed x_{pred}, y_{pred} is then the expected value of the pixel locations according to the distribution computed from the heatmap. Let i, j denote the coordinates in the heatmap, and ρ_{ij} denote the corresponding probability value:

$$\left[x_{pred}, y_{pred}\right] = \sum_{i=1}^{W}\sum_{j=1}^{H} \rho_{ij}\left[x_i, y_j\right] \qquad (1)$$

3.3. Loss Function

We apply a pixelwise sigmoid cross entropy [23] to learn the heatmaps, which we denote as L_h. Additionally, in order to alleviate issues with the heatmaps being cut off for landmarks near boundaries, we add on an L_2 distance loss with a loss weight λ: $L_{total} = L_h + \lambda \cdot L_2$.

$$L_h = \frac{1}{N}\sum_{n=1}^{N}\sum_{l=1}^{L}\sum_{i=1}^{W}\sum_{j=1}^{H} \qquad (2)$$

$$[\rho_{ij}^{l} log \hat{\rho_{ij}^{l}} + (1-\rho_{ij}^{l}) log(1-\hat{\rho_{ij}^{l}})] \cdot w_{ij}^{l}$$

$$w_{ij}^{l} = \frac{((i^n - \hat{i_l^n})^2 + (j^n - \hat{j_l^n})^2) \cdot 2}{W^2 + H^2} \qquad (3)$$

Where ρ_{ij}^{l} is the prediction value of the heatmap in the lth channel at pixel location (i, j) of n's sample, while $\hat{\rho_{ij}^{l}}$ is the corresponding ground truth. w_{ij}^{l} is the weight at that location, which is calculated from Equation 3. $(\hat{i_l^n}, \hat{j_l^n})$ is the ground truth coordinate of the n's sample's lth landmark.

3.4. Data

Our dataset consists of a union of Helen [11], LFPW [1] and iBUG [17] datasets and additional images that we collected in-house, with 3681 images in total. The image annotation follows Helen annotation format [11], but with corrected original annotations and with an addition of a couple of new eyebrow landmarks. The contour annotation is sparse. This dataset has a total of 62 inner points and 3 contour points as shown in Figure 2. The pose distribution is approximately -30°to 30°in yaw angle, which agrees well with our application scenario. For the purpose of comparison with state-of-the-art methods, we further train and test our model design on the 300W dataset [16].

4. Results and Ablation Study

The reported error is the commonly used mean squared error normalized by the inter-pupil distance. Table 1 shows the full model error (of the two-stage model trained with the heatmap loss, combined with L_2 loss) in the last row, and the results when only L_2 loss (first row), or only heatmap loss (second row), or only one CNN stage (third row) are used. This ablation study clearly shows that all three tested components: heatmap loss, L_2 loss, and extending the architecture with a second processing stage, bear individual importance and improves the model accuracy.

The tests on 300W dataset and comparisons with Look at Boundary (LAB) [21], a state-of-the-art method for accurate facial alignment, resulted in the following (Table 2): our full-size model of 6.6MB achieves similar accuracy as LAB method on 300W Common Subset, while exhibiting some drop in accuracy on the 300W Full dataset, which can be explained by our architectural design being adapted more towards landmark detection on unoccluded and mostly frontal faces. In addition, "half α" ($\alpha = 0.5$), which halves the number of channels per layer, preserves the accuracy when tested on the Common Subset. Finally, we show that a further reduction in model size and input image resolution does not significantly degrade the accuracy, as shown in Table 3, while attaining much faster computational speeds and overall a smaller model size.

Method	Inner Error	Contour Error
Without heatmap loss	3.53	9.01
Without L_2 loss	4.45	12.3
Without second stage	3.50	9.32
Our full model	**3.21**	**9.00**

Table 1. Results on our dataset with 65 points.

Method	Common Subset	Challenging Subset	Fullset
LAB (4-stack)[21]	4.20	7.41	4.92
LAB (8-stack)[21]	**3.42**	**6.98**	**4.12**
Our model ($\alpha = 1$)	**3.42**	8.51	4.42
Our model ($\alpha = 0.5$)	3.43	14.7	5.64

Table 2. Results on 300W dataset.

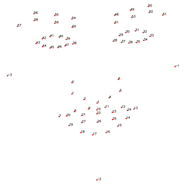

Figure 2. The annotation of 65 landmarks.

Total params	158,091
Total MAdd	173.69M
Total Flops	90.75M
Model Size	607KB
Inference Time (iPhone XR)	20ms

Table 3. Our demo model, results in Table 1 last row.

5. Rendering

Following landmark prediction by our CNN model, we define facial parts (e.g., lips) and render makeup on them. Figure 3 depicts the full pipeline, included rendering modules.

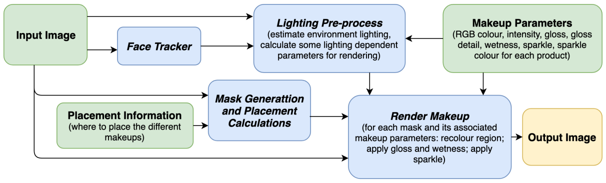
Input Image
Face Tracker
Lighting Pre-process
(estimate environment lighting, calculate some lighting dependent parameters for rendering)
Makeup Parameters
(RGB colour, intensity, gloss, gloss detail, wetness, sparkle, sparkle colour for each product)
Mask Generattion and Placement Calculations
Placement Information
(where to place the different makeups)
Render Makeup
(for each mask and its associated makeup parameters: recolour region; apply gloss and wetness; apply sparkle)
Output Image

Figure 3. The Rendering Pipeline.

6. Acknowledge

We thank Brittany Wanka, Rena Lu and Sarah Koehler for assistance with cleaning and annotating data. We would also like thank Irene Jiang for comments and discussions.

References

[1] Peter N. Belhumeur, David W. Jacobs, David J. Kriegman, and Neeraj Kumar. Localizing parts of faces using a consensus of exemplars. In *The 24th IEEE Conference on Computer Vision and Pattern Recognition*, 2011. 2, 3

[2] Adrian Bulat and Georgios Tzimiropoulos. Human pose estimation via convolutional part heatmap regression. In *European Conference on Computer Vision(ECCV)*, 2016. 2

[3] Yu Chen, Chunhua Shen, Hao Chen, Xiu-Shen Wei, Lingqiao Liu, and Jian Yang. Adversarial learning of structure-aware fully convolutional networks for landmark localization. *IEEE transactions on pattern analysis and machine intelligence*, 2019. 2

[4] Ross Girshick. Fast r-cnn. In *the IEEE international conference on computer vision*, 2015. 2

[5] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask r-cnn. In *the IEEE international conference on computer vision*, 2017. 2

[6] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *the IEEE conference on computer vision and pattern recognition workshops*, 2016. 2

[7] Forrest N Iandola, Song Han, Matthew W Moskewicz, Khalid Ashraf, William J Dally, and Kurt Keutzer. Squeezenet: Alexnet-level accuracy with 50x fewer parameters and¡ 0.5 mb model size. 2016. 2

[8] Vahid Kazemi and Josephine Sullivan. One millisecond face alignment with an ensemble of regression trees. In *the IEEE conference on computer vision and pattern recognition workshops*, 2014. 1

[9] Davis E King. Dlib-ml: A machine learning toolkit. *Journal of Machine Learning Research*, 2009. 1

[10] Marek Kowalski, Jacek Naruniec, and Tomasz Trzcinski. Deep alignment network: A convolutional neural network for robust face alignment. In *the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 2017. 1

[11] Vuong Le, Jonathan Brandt, Zhe Lin, Lubomir Bourdev, and Thomas S Huang. Interactive facial feature localization. In *European conference on computer vision*, 2012. 2, 3

[12] Jonathan Long, Evan Shelhamer, and Trevor Darrell. Fully convolutional networks for semantic segmentation. In *the IEEE conference on computer vision and pattern recognition*, 2015. 2

[13] Alejandro Newell, Kaiyu Yang, and Jia Deng. Stacked hourglass networks for human pose estimation. In *European Conference on Computer Vision*, 2016. 2

[14] Leonid Pishchulin, Eldar Insafutdinov, Siyu Tang, Bjoern Andres, Mykhaylo Andriluka, Peter V Gehler, and Bernt Schiele. Deepcut: Joint subset partition and labeling for multi person pose estimation. In *the IEEE Conference on Computer Vision and Pattern Recognition*, 2016. 2

[15] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. In *Advances in neural information processing systems*, 2015. 2

[16] Christos Sagonas, Georgios Tzimiropoulos, Stefanos Zafeiriou, and Maja Pantic. 300 faces in-the-wild challenge: The first facial landmark localization challenge. In *the IEEE International Conference on Computer Vision Workshops*, 2013. 3

[17] Christos Sagonas, Georgios Tzimiropoulos, Stefanos Zafeiriou, and Maja Pantic. A semi-automatic methodology for facial landmark annotation. In *the IEEE conference on computer vision and pattern recognition workshops*, 2013. 2, 3

[18] Mark Sandler, Andrew Howard, Menglong Zhu, Andrey Zhmoginov, and Liang-Chieh Chen. Mobilenetv2: Inverted residuals and linear bottlenecks. In *the IEEE Conference on Computer Vision and Pattern Recognition*, 2018. 2

[19] Yi Sun, Xiaogang Wang, and Xiaoou Tang. Deep convolutional network cascade for facial point detection. In *the IEEE conference on computer vision and pattern recognition*, 2013. 1

[20] Shih-En Wei, Varun Ramakrishna, Takeo Kanade, and Yaser Sheikh. Convolutional pose machines. In *the IEEE conference on computer vision and pattern recognition workshops*, 2016. 2

[21] Wayne Wu, Chen Qian, Shuo Yang, Quan Wang, Yici Cai, and Qiang Zhou. Look at boundary: A boundary-aware face alignment algorithm. In *the IEEE Conference on Computer Vision and Pattern Recognition*, 2018. 3

[22] Kevan Yuen and Mohan M Trivedi. An occluded stacked hourglass approach to facial landmark localization and occlusion estimation. *IEEE Transactions on Intelligent Vehicles*, 2017. 1, 2

[23] Ning Zhang, Evan Shelhamer, Yang Gao, and Trevor Darrell. Fine-grained pose prediction, normalization, and recognition. *arXiv preprint arXiv:1511.07063*, 2015. 3

[24] Xiangyu Zhang, Xinyu Zhou, Mengxiao Lin, and Jian Sun. Shufflenet: An extremely efficient convolutional neural network for mobile devices. In *the IEEE Conference on Computer Vision and Pattern Recognition*, 2018. 2