APSIPA Transactions on Signal and Information Processing, 2023, 12, e45

Original Paper

Lightweight Boundary-Aware Face Alignment with Compressed HourglassNet and Transformer

Wenhui Wang[1,2], Yingxin Li[1,2], Ziqiang Li[1,2] and Jingliang Peng[1,2*]

[1] *Shandong Provincial Key Laboratory of Network Based Intelligent Computing, University of Jinan, Jinan 250022, China*
[2] *School of Information Science and Engineering, University of Jinan, Jinan 250022, China*

ABSTRACT

In this work, we focus on lightweight and accurate face alignment. For that purpose, we propose an algorithm design that promotes a most recently published face alignment method in terms of model size and computing cost while maintaining high accuracy of face alignment. Specifically, we construct a lightweight two-stage neural network. The first stage estimates boundary heatmaps on the facial region, which are then used to guide the facial landmark position prediction in the second stage. For the first stage, we compress an HourglassNet-based structure by reducing the numbers of feature channels and convolutional kernels and optimizing the structure of Hourglass block by ShuffleNet modules. For the second stage, we compress the subnet by utilizing DeLighT, a recently published lightweight version of Transformer. Experimental results on several standard facial landmark detection datasets show that the proposed algorithm achieves sharp advances in model compactness and computing efficiency while keeping a state-of-the-art level of accuracy in facial landmark detection.

*Contributing author: Jingliang Peng, ise_pengjl@ujn.edu.cn. W. Wang and Y. Li contributed equally to this work. This work was supported by Shandong Provincial Natural Science Foundation, China (No. ZR2022MF294).

Received 29 August 2023; Revised 11 September 2023
ISSN 2048-7703; DOI 10.1561/116.00000059

Keywords: Face alignment, Facial landmark detection, Lightweight.

1 Introduction

Facial landmark detection, or named face alignment, can identify the positions and sizes of various key parts of the face, which may provide useful information for subsequent tasks. It is frequently an indispensable task for other computer vision tasks such as face recognition, face pose estimation and face expression recognition.

Traditional facial landmark detection algorithms are mainly based on handcrafted features and require a lot of computation and manual tuning. These methods usually have difficulty dealing with complex scenes. By contrast, deep learning based facial landmark detection algorithms have strong feature extraction capabilities and are able to process highly complex scenes. As such, recent advances in face alignment are usually based on deep learning technology.

On the one hand, more and more deep learning based face alignment models have been proposed, which are increasingly capable of accurate and robust detection of facial landmarks. On the other hand, these models tend to be increasingly complex in terms of model size and inference speed. In order to deploy face-related vision applications on low-end computing platforms, it is often crucial to make lightweight neural network models for the face alignment task. This has motivated our research in this work.

We propose a lightweight neural network model for face alignment in this work. Instead of devising a new model completely from scratch, we choose to optimize an existing state-of-the-art (SOTA) face alignment model [6] that conducts boundary-aware face alignment with enhanced hourglassNet and transformer. It has achieved superior accuracy of face alignment. We call it BAFA model and use it as reference model. Specifically, we follow a two-stage framework with the first stage estimating the boundary heatmaps and the second stage utilizing the estimated boundary heatmaps to guide the prediction of landmark positions. On each stage, we conduct significant compression to the network structure, which sharply reduces model complexity and floating-point computation at comparable prediction accuracies. Major contributions of this work can be summarized as follows.

- **Concise HourglassNet-based subnet for boundary heatmap regression**. For the boundary heatmap regression in the first stage, we optimize the original reference subnet by utilizing ShuffleNet [19] techniques and reducing the volumes of feature channels and convolutional kernels.
- **Concise Transformer-based subnet for landmark coordinate regression**. For the landmark coordinate regression in the second stage, we also make a major improvement to the original reference subnet by using DeLighT [9], a recently published lightweight version of Transformer.

- **Superior efficiency and accuracy of the holistic model**. With the two concise subnets integrated, the proposed model reduces the size and computation of the original reference SOTA model by around 50% with nearly negligible degradation in accuracy of face alignment in our experiments.

2 Related Works

2.1 Traditional Machine Learning Based Algorithms

In traditional facial landmark detection methods, handcrafted feature descriptors such as SIFT [7] and HOG [1] are commonly used. Among these methods, the pose normalization algorithm (PNU) [2] is a representative algorithm that normalizes facial images to a unified pose, selects landmarks manually, and learns landmark positions using support vector regression (SVR) [10]. Compared to other traditional algorithms, the PNU algorithm can better handle problems such as changes in facial pose, occlusion, and expressions. However, it requires a lot of manual intervention and design, and entails a high computational cost.

2.2 Deep Learning Based Algorithms

In recent years, many facial landmark detection methods have been proposed based on deep learning technology. Among these, coordinate regression and heatmap regression have become two mainstream approaches to facial landmark detection.

Coordinate regression methods utilize neural networks to learn the mapping from an input image to facial landmarks coordinates. This method directly obtains precise coordinates of facial landmarks, but it requires more training samples and complex network structures. Some effective methods include: LAB [17] introduces a boundary-aware face alignment algorithm that estimates boundary heatmaps and utilizes boundary information to accurately predict facial landmarks. Wing Loss [3] introduces a novel loss function for robust facial landmark localization using convolutional neural networks. ODN [20] places a specific emphasis on encoding features in occluded regions and merging facial geometric features with semantic features. SLPT [18] introduces a sparse local patch transformer to achieve robust face alignment. BAFA [6] proposes a boundary-aware face alignment model that firstly predicts boundary heatmaps and then uses them to guide the landmark coordinates prediction.

Heatmap regression methods utilizes the prediction of heatmaps surrounding facial landmarks to achieve landmark localization. This method is relatively simple and does not rely on direct coordinate regression, making it

capable of handling issues such as occlusion and blurriness. However, due to the discrete nature of heatmap predictions, there may be cases of inaccurate localization, which may require post-processing or interpolation. Some effective methods include: AWing [15] enhances the Wing loss function by making its value approach zero for small errors, resulting in improved accuracy and robustness. HRNetV2 [14] enhances the network's perception of details and local features by multi-branch modules and feature fusion across resolutions. LUVLi [5] provides a method for predicting the visibility of each landmark and the algorithm's confidence, which helps us better understand and analyze the landmarks in the image. HIH [4] utilizes two types of heatmaps, namely the original heatmap and the quantization-robust heatmap, to collaboratively counteract the impact of quantization on the results. MMDN [13] explores the high-order feature correlations to enhance the robustness of detection. PIPNet [4] predicts landmark heatmaps and offset values that are used to finally derive the landmarks positions.

The deep learning based models proposed so far tend to be complex in terms of size and computation. Therefore, it is crucial to compress face alignment models while maintaining a good accuracy, such that they may be suited for applications on low-end platforms.

3 Method

We build our lightweight boundary-aware face alignment (LW-BAFA) model with reference to BAFA [6], a SOTA face alignment model that has been most recently published. As BAFA, our proposed lightweight model is composed of two stages. The first stage estimates boundary heatmaps and the second stage predicts landmarks positions under the guidance of the estimated boundary heatmaps. In both stages, careful designs are made to make them lightweight. The structural diagram is shown in Figure 1, where the first (resp. second) row corresponds to the first (resp. second) stage. Details about the two stages (or subnets) are provided in the following subsections.

3.1 Boundary Heatmap Estimation

As in the BAFA architecture, the first stage (or boundary heatmap estimation subnet) in LW-BAFA is composed of a preprocessing block, four Hourglass blocks with SDFusion (shallow and deep feature fusion) blocks in between and a convolution layer, as shown in Figure 1. BAFA [6] mentioned that the SDFusion module is to connect adjacent hourglass modules and generate attention maps to enhance the fusion of shallow convolutional outputs and features obtained from the pyramid pooling module. The module utilizes two 1×1 convolutions and one residual module for feature fusion. Additionally,

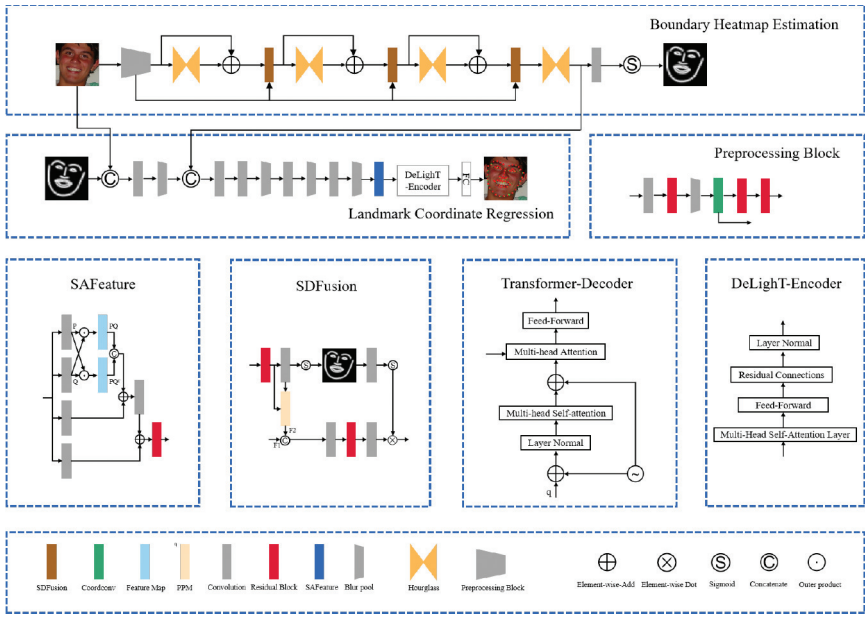


Figure 1: Architectural overview of our proposed lightweight boundary-aware face alignment model. The first row represents the subnet responsible for estimating the boundary heatmap, while the second row represents the subnet responsible for predicting landmark coordinates. The remaining portion illustrates the fine structures within the neural network. Further details about the model can be found in Section 3.

each SDFusion module can predict boundary heatmaps, which can be used not only for intermediate supervision but also for generating attention maps. Different from BAFA, the first stage in our proposed LW-BAFA is compressed by reducing convolution filters and channels and modifying Hourglass blocks by ShuffleNet modules, as detailed below.

As shown in Figure 1, the preprocessing block contains three residual blocks. We compress each residual block by reducing the numbers of filters in the first and second convolution layers. Specifically, the residual block in BAFA is shown on the left part of Figure 2 and, after the simplification, the residual block in LW-BAFA is shown on the right.

Furthermore, we modify the Hourglass blocks in BAFA by drawing inspiration from ShuffleNetV2 [8]. In ShuffleNetV2, the residual module adopts depthwise separable convolution that significantly reduces the parameter count and computational cost of the model. Moreover, ShuffleNetV2 introduces channel shuffling operation to enhance the model's representational capacity in the channel dimension and improve its nonlinear modeling ability. Other advantages of the ShuffleNetV2 residual module include reduced correlation between channels and adaptiveness to inputs of different resolutions. Therefore,

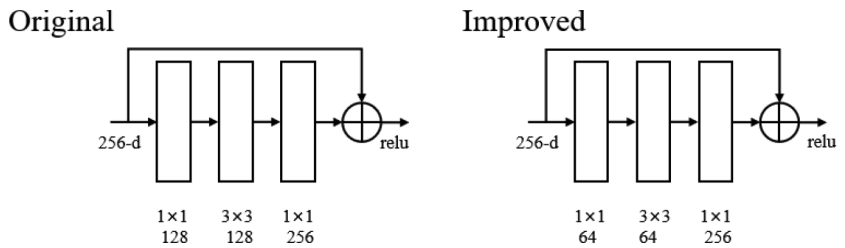


Figure 2: Comparison between the original residual module and the improved residual module.

we are motivated to replace the residual units in a traditional Hourglass block with the lighter-weight residual units from ShuffleNetV2, as shown in Figure 3. Note that we choose to make the replacement on only the decoder side but not throughout the whole Hourglass block. This is an empirical decision.

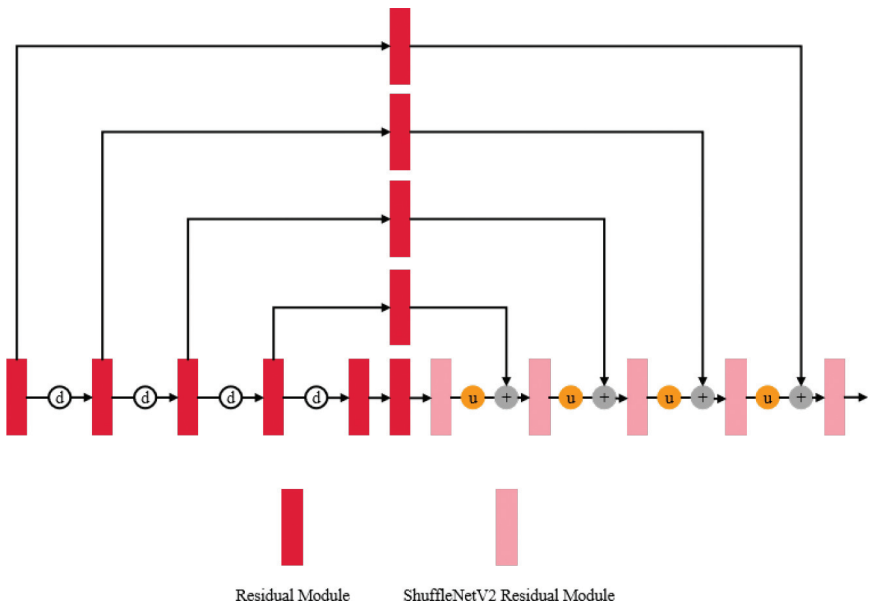


Figure 3: Structural overview of the improved Hourglass block.

3.2 Landmark Coordinate Prediction

The second stage (or landmark coordinate prediction subnet) in LW-BAFA is composed of convolutional and blur pooling layers, a SAFeature (self-attention-based feature re-extraction) module and a DeLighT encoder [9], as shown in

Figure 1. BAFA [6] mentioned that the SAFusion module consists of multiple branches and utilizes matrix outer product to generate new feature maps, which are then fused using self-attention mechanism. The purpose of this module is to extract and integrate key information among different inputs, thereby aiding in the final prediction results. Compared with BAFA, the major contribution by LW-BAFA in the second stage is the utilization of DeLighT encoder instead of vanilla Transformer decoder.

Although Transformer has achieved great successes in addressing various vision tasks, its multi-head attention mechanism [12, 16] and many layers of feed-forward networks make the model have a large number of parameters, heavy computational burden and high storage requirements.

Our work involves replacing the Transformer decoder in BAFA with the DeLighT encoder, which can make the model more lightweight. Specifically, the DeLighT encoder reduces the number of heads in the attention mechanism, uses fewer self-attention mechanisms and feed-forward network layers, and replaces the ordinary convolution operation with separable convolution. These changes make the model more compact, lightweight, and still able to maintain high prediction performance. Therefore, by replacing the Transformer decoder with the DeLighT encoder, we can reduce the computational burden and storage requirements of the model without significant impairment to its prediction accuracy. Note that this replacement strategy was fixed through empirical study. We explored alternative strategies like replacing the encoder instead of the decoder in the Transformer, or using the decoder instead of the encoder from the DeLighT. Among all the replacement strategies we explored, replacing the Transformer decoder with the DeLighT encoder yielded the best results, and we ultimately chose it. In order to accommodate the DeLighT encoder in the network pipeline, necessary interface changes and feature reshapings are made.

3.3 Loss Function

The complete loss function, $Loss$, is composed of two components, L_{lm} and L_{bh}, representing the losses incurred by the landmark coordinates prediction and the boundary heatmap estimation, respectively. It is specifically defined by

$$L_{lm} = \frac{1}{N_{lm}} \sum_{i=1}^{N_{lm}} ||p_i - \hat{p}_i||^2. \tag{1}$$

$$L_{bh} = \frac{1}{N_{bh}} \sum_{i=1}^{N_{bh}} \omega_i ||H_i - \hat{H}_i||^2. \tag{2}$$

$$Loss = L_{lm} + \beta L_{bh}. \tag{3}$$

In (1), N_{lm} denotes the number of facial landmarks, p_i and $\hat{p}_i$ denote the predicted coordinates and the ground truth, respectively. In (2), N_{bh} denotes

the number of predicted boundary heatmaps (equal to the number of hourglass blocks), H_i and $\hat{H}_i$ are the i-th predicted boundary heatmap and the i-th ground truth, respectively, ω_i is the weight. In (3), β is a hyperparameter to regulate the two types of losses, which is set to 0.0001 by default.

4 Experiments

4.1 Implementation Details

Each input image undergoes cropping and resizing to a size of 256 × 256. Each boundary heatmap is sized to 64 × 64. In order to enhance the training data, random translation (±10%), rotation (±30°), horizontal flipping (50%), illumination adjustments (±20%), blurring (10%) and occlusion are performed. During training, we employ an Adam optimizer with an initial learning rate of 1×10^{-4} and β_1 and β_2 values of 0.5 and 0.9, respectively. The network is trained on one GPU (NVIDIA 3090 24GB) for 150 epochs, and the learning rate is reduced to 1/10 of the previous value for twice at the 90th and the 120th epochs. The batch size is 16 and, in the loss function, the weights $\omega_{i=1,2,3,4}$ are 0.25, 0.5, 0.75 and 1.0, respectively.

4.2 Metrics and Datasets

We use Normalized Mean Error (NME), Failure Rate (FR) and Area under the Curve (AUC) to measure the prediction accuracy. NME is defined as

$$\text{NME}(P, \hat{P}) = \frac{1}{N} \sum_{i=1}^{N_{lm}} \frac{||p_i - \hat{p}_i||^2}{d} \times 100\%. \tag{4}$$

where P and $\hat{P}$ denote the predicted and annotated coordinates of landmarks, respectively, p_i and $\hat{p}_i$ indicate the coordinates of the i-th landmark in P and $\hat{P}$, respectively, N is the number of the facial landmarks, and d is the reference distance (*i.e.* the inter-ocular distance) to normalize the error. FR refers to the percentage of the failed images whose NMEs are above a certain threshold in the test set. AUC is calculated based on the cumulative error distribution curve, and a larger AUC value means more images well estimated.

In addition, we use Parameters (Param.), Giga Floating-Point Operations (GFLOPs) and Frames per Second (FPS) to measure the model complexity. Param. refers to the number of optimizable weight parameters and measures the memory requirements of a model, while GFLOPs and FPS measure the computational efficiency of a model.

The 300 W [11] dataset contains 3,148 images for training and 689 images for testing. Following the widely used evaluation setting, the test sets usually

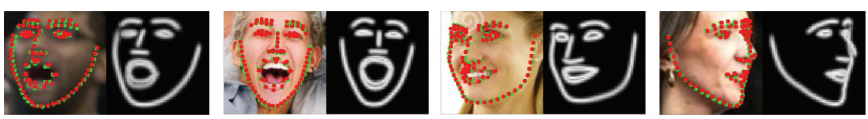
consist of he common set (554 images), the challenging set (135 images) and the full set (the total 689 images). Each image in 300W is annotated with 68 facial landmarks.

The WFLW dataset [17] contains 7,500 images for training and 2,500 images for testing with 98 landmarks and rich attribute labels. It also has six different test subsets with attribute labels, such as occlusion, make-up and illumination.

4.3 Results and Analysis

Figure 4 depicts some results of selected test images from the 300W and WFLW datasets. For each test image, the ground-truth and predicted landmark locations are marked with red and green colors in the left image, respectively, and the boundary heatmap estimated by the proposed method is shown in the right image. From these examples, it can be observed that our proposed model can adapt well to various challenging situations. In most of the cases shown here, the predicted results are close to the ground truth.

(a) 300 W



(b) WFLW

Figure 4: Selective results on test images from the 300W and the WFLW datasets. For each test image, the ground-truth and predicted landmark locations are respectively represented by red and green points in the left image, while the boundary heatmap estimated by the first network stage is displayed in the right image.

Evaluation of accuracy on 300W. We conducted a comparative analysis between our proposed method and various advanced techniques using three different sets of test, including the common, challenging, and full sets. The outcomes of the evaluation are presented in Table 1. Among them, BAFA [6] and AWing [15] achieve the best results. Although our proposed model is slightly inferior to them, it is still comparable with PIPNet [4] and superior to all the rest models.

Evaluation of accuracy on WFLW. We conducted a comparison of different methods on the Testset as well as several subsets, including large pose, expression, illumination, make-up, occlusion, and blur (we respectively

Table 1: Comparing with state-of-the-art methods on 300W. Key: [Best, Second Best, Third Best].

	Method	Common	Challenging	Full
NME(%) ↓	AnchorFace	3.12	6.19	3.72
	SRN	3.08	5.86	3.64
	SRN+HG	3.03	5.38	3.49
	LAB	2.98	5.19	3.49
	HRNetV2	2.87	5.15	3.32
	ACHR	2.83	7.04	4.23
	PIPNet	2.78	4.89	3.19
	AWing	2.72	4.52	3.07
	BAFA	2.71	4.70	3.10
	Ours	2.78	4.98	3.21

Table 2: Comparison with state-of-the-art methods on WFLW (Testset). Key: [Best, Second Best, Third Best].

Method	NME(%)↓	FR(%)↓	AUC ↑
LAB	5.27	7.56	0.5323
Wing	5.11	6.00	0.5504
MMDN	4.87	-	-
LUVLi	4.37	3.12	0.5777
AWing	4.36	2.84	0.5719
AnchorFace	4.32	2.96	0.5769
PIPNet	4.31	-	-
HIH_C	4.18	2.96	0.5970
BAFA	4.16	2.32	0.5927
Ours	4.31	2.68	0.5817

substitute lp, exp, ill, m-u, occ and blu to represent them in Table 3). The comparison results are presented in Table 2 and Table 3. From these tables, it can be observed that BAFA [6] and HIH [4] consistently achieve the best results in most cases. Although our proposed model is slightly inferior to them, it is still comparable with PIPNet [4] and superior to all the rest models.

Evaluation of efficiencies on WFLW. Besides the accuracy, we also compare the efficiencies of the SOTA models by the metrics of parameter count, GFLOPs, and FPS on the WFLW dataset. Since we need to run all these models on our computing platform for this comparison, we are only able to compare with a subset of the benchmark algorithms that have source codes

Table 3: Comparison with state-of-the-art methods on WFLW (Subset). Key: [Best, Second Best, Third Best].

Method	lp	exp	ill	m-u	occ	blu
LAB	10.24	5.51	5.23	5.15	6.79	62.3
Wing	8.75	5.36	4.93	5.41	6.37	5.81
MMDN	8.15	4.99	4.61	4.72	6.17	5.72
LUVLi	-	-	-	-	-	4.79
AWing	7.38	4.58	4.32	4.27	5.19	4.96
PIPNet	7.51	4.44	4.19	4.02	5.36	5.02
HIH_C	7.20	4.19	4.45	3.97	5.00	4.81
BAFA	7.20	4.46	4.07	4.10	4.87	4.66
Ours	7.50	4.67	4.21	4.39	5.10	4.81

Table 4: Comparison with state-of-the-art methods on Parameter count, GFLOPs and FPS on WFLW. Key: [Best, Second Best, Third Best].

Method	Param.(M)	GFLOPs	FPS(GPU)
PIPNet	45.7	10.5	51.2
AWing	25.1	26.7	32.7
MMDN	16.2	86.46	15.7
SLPT	13.19	6.12	16.1
HRNet	9.7	4.8	7.6
BAFA	19.96	22.23	32.1
Ours	9.48	11.4	34.1

released. Specifically, we are only able to compare with PIPNet [4], AWing [15], MMDN [13], SLPT [18], HRNet [14] and BAFA [6] of all the benchmark algorithms reported in Table 1, Table 2 and Table 3. The comparison results are shown in Table 4. Our model has the smallest number of parameters. In particular, its parameter count is about 1/5 that of PIPNet, less than 1/2 that of BAFA and less than 1/2 that of AWing. In terms of GFLOPs, it is inferior to HRNet abd SLPT but comparable to PIPNet and superior to the rest. In terms of FPS, it ranks second only to PIPNet.

As shown in Table 4, compared with BAFA, Ours promotes the FPS but the increase of FPS does not match the sharp decrease of GFLOPs. This can be explained as follows. On the one hand, the proposed model compression is done mainly by reducing convolution kernels, reducing convolution to depthwise separable convolution and reducing heads of attention in the transformer. On the other hand, convolution and multi-head attention computing have been

highly parallelized by current systems and environments for deep learning and, therefore, kernel number, kernel size and head count may not impact the inference time too much with decent GPU configurations. But, in any way, reduction of GFLOPs is important since it saves energy for computing and facilitates running inferences on low-configuration platforms.

Considering all the accuracy and efficiency statistics in Table 1, Table 2, Table 3 and Table 4, we conclude that our proposed model achieves superior efficiencies, especially so in terms of parameter count, while maintaining the SOTA level of accuracy.

5 Conclusion

In this paper, we have proposed a lightweight boundary-aware face alignment algorithm. It promotes a SOTA two-stage reference face alignment model. In the first stage for boundary heatmap estimation, it applies ShuffleNet techniques and reduces feature channel volumes and convolutional kernels to simply the subnet. In the second stage, it then introduces a concise Transformer-based subnet for landmark coordinate regression. This improvement is achieved by utilizing DeLighT, a lightweight version of Transformer. By integrating these concise subnets, our proposed model achieves superior efficiency compared to the original reference model with nearly negligible degradation in accuracy. Compared with other SOTA face alignment models, our proposed model also achieves superior performance taking both efficiency and accuracy into account.

In the future, we plan to further lighten the network structure as well as make efficient model design adaptive to various hardware platforms and operating systems.

References

[1] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, Vol. 1, 2005, 886–893 vol. 1, DOI: 10.1109/CVPR.2005.177.

[2] Ding, Changxing, Tao, and Dacheng, "Pose-invariant face recognition with homography-based normalization," *Pattern Recognition: The Journal of the Pattern Recognition Society*, 66, 2017, 144–52.

[3] Z.-H. Feng, J. Kittler, M. Awais, P. Huber, and X.-J. Wu, "Wing loss for robust facial landmark localisation with convolutional neural networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, 2235–45.

[4] H. Jin, S. Liao, and L. Shao, "Pixel-in-pixel net: Towards efficient facial landmark detection in the wild," *International Journal of Computer Vision*, 129(12), 2021, 3174–94.

[5] A. Kumar, T. K. Marks, W. Mou, Y. Wang, M. Jones, A. Cherian, T. Koike-Akino, X. Liu, and C. Feng, "Luvli face alignment: Estimating landmarks' location, uncertainty, and visibility likelihood," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, 8236–46.

[6] Y. Li, D. Niu, J. Peng, *et al.*, "Boundary-Aware Face Alignment with Enhanced HourglassNet and Transformer," *APSIPA Transactions on Signal and Information Processing*, 12(1), 2023.

[7] D. Lowe, "Object recognition from local scale-invariant features," in *Proceedings of the Seventh IEEE International Conference on Computer Vision*, Vol. 2, 1999, 1150–1157 vol.2, DOI: 10.1109/ICCV.1999.790410.

[8] N. Ma, X. Zhang, H. Zheng, and J. Sun, "ShuffleNet V2: Practical Guidelines for Efficient CNN Architecture Design," *CoRR*, abs/1807.11164, 2018, arXiv: 1807.11164, http://arxiv.org/abs/1807.11164.

[9] S. Mehta, M. Ghazvininejad, S. Iyer, L. Zettlemoyer, and H. Hajishirzi, "DeLighT: Very Deep and Light-weight Transformer," *CoRR*, abs/2008.00623, 2020, arXiv: 2008.00623, https://arxiv.org/abs/2008.00623.

[10] E. Osuna, R. Freund, and F. Girosit, "Training support vector machines: an application to face detection," in *Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 1997, 130–6, DOI: 10.1109/CVPR.1997.609310.

[11] C. Sagonas, G. Tzimiropoulos, S. Zafeiriou, and M. Pantic, "300 faces in-the-wild challenge: The first facial landmark localization challenge," in *Proceedings of the IEEE international conference on computer vision workshops*, 2013, 397–403.

[12] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," *Advances in neural information processing systems*, 30, 2017.

[13] J. Wan, Z. Lai, J. Li, J. Zhou, and C. Gao, "Robust facial landmark detection by multiorder multiconstraint deep networks," *IEEE Transactions on Neural Networks and Learning Systems*, 33(5), 2021, 2181–94.

[14] J. Wang, K. Sun, T. Cheng, B. Jiang, C. Deng, Y. Zhao, D. Liu, Y. Mu, M. Tan, X. Wang, *et al.*, "Deep high-resolution representation learning for visual recognition," *IEEE transactions on pattern analysis and machine intelligence*, 43(10), 2020, 3349–64.

[15] X. Wang, L. Bo, and L. Fuxin, "Adaptive wing loss for robust face alignment via heatmap regression," in *Proceedings of the IEEE/CVF international conference on computer vision*, 2019, 6971–81.

[16] S. Woo, J. Park, J.-Y. Lee, and I. S. Kweon, "Cbam: Convolutional block attention module," in *Proceedings of the European conference on computer vision (ECCV)*, 2018, 3–19.

[17] W. Wu, C. Qian, S. Yang, Q. Wang, Y. Cai, and Q. Zhou, "Look at boundary: A boundary-aware face alignment algorithm," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, 2129–38.

[18] J. Xia, W. Qu, W. Huang, J. Zhang, X. Wang, and M. Xu, "Sparse Local Patch Transformer for Robust Face Alignment and Landmarks Inherent Relation Learning," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, 4052–61.

[19] X. Zhang, X. Zhou, M. Lin, and J. Sun, "ShuffleNet: An Extremely Efficient Convolutional Neural Network for Mobile Devices," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018.

[20] M. Zhu, D. Shi, M. Zheng, and M. Sadiq, "Robust facial landmark detection via occlusion-adaptive deep networks," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, 3486–96.