applied sciences

MDPI

Article

EHANet: An Effective Hierarchical Aggregation Network for Face Parsing

Ling Luo *,†, **Dingyu Xue** and **Xinglong Feng** †

College of Information Science and Engineering, Northeastern University, Shenyang 110819, China; xuedingyu@mail.neu.edu.cn (D.X.); fengxinglong@vip.163.com (X.F.)

* Correspondence: lingluo@stumail.neu.edu.cn

† These authors contributed equally to this work.

Received: 7 April 2020; Accepted: 28 April 2020; Published: 30 April 2020

Abstract: In recent years, benefiting from deep convolutional neural networks (DCNNs), face parsing has developed rapidly. However, it still has the following problems: (1) Existing state-of-the-art frameworks usually do not satisfy real-time while pursuing performance; (2) similar appearances cause incorrect pixel label assignments, especially in the boundary; (3) to promote multi-scale prediction, deep features and shallow features are used for fusion without considering the semantic gap between them. To overcome these drawbacks, we propose an effective and efficient hierarchical aggregation network called EHANet for fast and accurate face parsing. More specifically, we first propose a stage contextual attention mechanism (SCAM), which uses higher-level contextual information to re-encode the channel according to its importance. Secondly, a semantic gap compensation block (SGCB) is presented to ensure the effective aggregation of hierarchical information. Thirdly, the advantages of weighted boundary-aware loss effectively make up for the ambiguity of boundary semantics. Without any bells and whistles, combined with a lightweight backbone, we achieve outstanding results on both CelebAMask-HQ (78.19% mIoU) and Helen datasets (90.7% F1-score). Furthermore, our model can achieve 55 FPS on a single GTX 1080Ti card with 640 × 640 input and further reach over 300 FPS with a resolution of 256 × 256, which is suitable for real-world applications.

Keywords: semantic segmentation; face parsing; semantic gap compensation block; stage contextual attention mechanism; weighted boundary-aware loss

1. Introduction

Face parsing, also known as face fine-grained segmentation, has attracted much attention due to its remarkable behavior, such as face beauty [1], face image synthesis [2], and expression transfer [3]. Face parsing aims to assign different semantic labels to each pixel of a facial image (e.g., nose, eye, hair, brow), as shown in Figure 1. In the past few decades, much effort has been devoted to building robust face-parsing models under the controlled scenarios. Although these methods have achieved promising results, they are usually severely degraded under uncontrolled scenarios, which limits their scope of application.

Recently, the performance of segmentation has been greatly improved by the involvement of deep convolutional neural networks (DCNNs), especially the well-known FCN-based [4–7] frameworks. However, most of the classical semantic segmentation structures rely on cumbersome backbones (e.g., VGG16 [4] occupies approximately 500 MB of memory, and it takes about 100 ms to perform a forward inference even on a powerful GPU), which is not conducive to the deployment of low-end embedded devices. For large-scale model deployments, low-latency and high-efficiency are often contradictory. How to balance the relationship between them is a problem to be considered in the task of facial parsing.

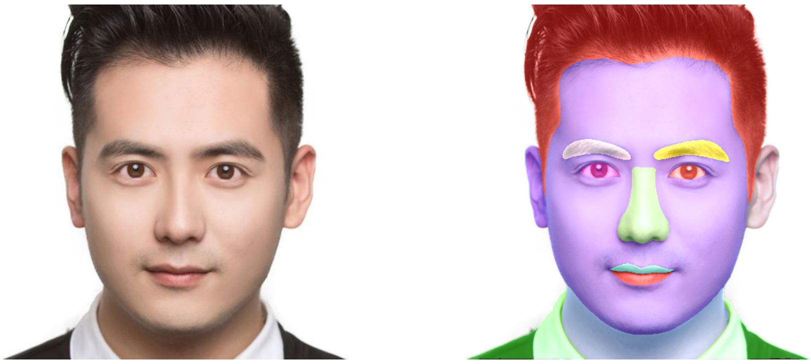

(**a**)Input RGB image (**b**)Rendered image

Figure 1. An example of face parsing.

Compared with general semantic segmentation tasks, there are three main challenges in face parsing. Firstly, since the facial area of a person is symmetrical, it is challenging to distinguish the left and right eyes because they have similar representations and textures. Secondly, boundary ambiguity (e.g., the region between hair and dark hats) often interferes with the visual system of annotators, which also confuses the learned model. Thirdly, in general, features from shallow layers encode more detailed information, while features from deep layers encode more semantic information to distinguish between different categories. To enhance the model's feature representation ability and extract multi-scale features, *concatenation* or *add* is usually used for aggregation between different feature blocks. However, the semantic gap between these blocks at different stages is rarely considered, which can have a negative effect on performance.

To address the aforementioned problems, including reducing the delay of the network, the categories being difficult to distinguish, the boundaries being ambiguous, and the semantic gaps between different layers, we propose a compact and effective hierarchical aggregation network named EHANet which can compensate the magnitude of the receptive field between layers at different stages and effectively reduce the computational complexity of the model. More specifically, we first present a so-called stage contextual attention mechanism (SCAM) that weights feature map channels at different stages to model the correlation between feature map channels. Secondly, we introduce a semantic gap compensation block (SGCB) to compensate for the semantic gap between different feature blocks. Furthermore, in view of the ambiguity of the boundary, we propose a weighted boundary-aware loss, which effectively improves the boundary semantic discrimination ability. Finally, through the use of an FPN-like [8] structure and a lightweight ResNet18 [9] backbone, a network integrating efficiency and performance is obtained.

The proposed method is evaluated on two public datasets, CelebAMask-HQ [10] and Helen [11]. Extensive experimental results show that the performance of our proposed network is comparable to the state-of-the-art models, while creating a lesser resource overhead. Specifically, we obtain 78.19% mIoU (mean intersection of union) with 71 FPS and 76.69% mIoU with 89 FPS on the CelebAMask-HQ `test` set while achieving a 90.7% F1-score on the Helen `test` set on a single 1080Ti GPU.

In summary, our contributions are four-fold as follows:

- An end-to-end powerful and efficient network is proposed, called EHANet, which can make up for the semantic gap between different hierarchies and improve the overall discriminative ability of the model.
- We propose a stage contextual attention mechanism, which re-encodes feature map channels to effectively utilize the correlations between different feature map channels.

- A weighted boundary-aware supervision is designed to enhance the network's ability to distinguish between different categories in the boundary area.
- We verify the effectiveness of the method on two benchmark datasets, and the results prove the superiority of our algorithm.

2. Related Work

Face Parsing. Early works mainly focused on exemplars and graphical models. Kae et al. [12] combined the conditional random field and the Boltzmann machine to model both local and global structure in face segmentation. Liu et al. [13] integrated the CNN into graphical models for structured prediction problems. Smith et al. [11] proposed an exemplar-based segmentation algorithm that exploits landmarks and SIFT features to transfer partial masks from aligned exemplars to the test images. With the popularity of DCNNs, a lot of CNN-based work has emerged. Liu et al. [14] proposed a pixel-level face parsing network by combining shallow CNN and spatial variant RNN. Guo et al. [15] designed an encoder–decoder network for face parsing with the help of a novel adaptive prior mechanism. Considering that a traditional crop-and-resize pipeline may ignore the contextual area outside the regions of interest (such as hair), Lin et al. [16] proposed a "RoI tanh-warping" image processing method, which achieved a state-of-the-art result.

Lightweight Networks. Han et al. [17] first proposed a lightweight model called SqueezeNet, which employs several 1 × 1 convolutions and parallel convolutions. While the performance is equivalent to AlexNet, the calculation amount of the model parameters is reduced by eight times. MobileNet [18,19] introduces deep separable convolutions to achieve low-latency results without a significant drop in accuracy. From the perspective of optimizing the network structure, ShuffleNet [20,21] fuses the operations of group convolution and channel shuffling to ensure the information flow and dimensionality reduction between channels. ShuffleNet v2 innovatively establishes four guidelines, which are very helpful for the design of lightweight architectures. Octave convolution [22] reduces the size of low-frequency features through feature sharing in adjacent locations, thereby reducing feature redundancy and memory consumption. Some recent solutions [23–25] mainly refer to the above modules and channel pruning [26] tricks to reduce model runtime.

Contextual Information. A great deal of research has focused on exploiting contextual information to enhance the representation capabilities of segmentation. Global pooling is widely used in various backbones to obtain the contextual information for global representation. Dilated convolution [27] expands the receptive field by introducing an expansion rate, which is commonly used in semantic segmentation tasks. DFANet [23] aggregates discriminative features through sub-network and sub-stage cascade, respectively. PSPNet [28] uses multi-scale pyramid pooling to obtain features at different scales. ACFNet [29] harvests the contextual information from a categorical perspective. Recently, ExFuse [30] has been proposed to improve the low-level context through additional supervision of the encoder.

Attention Mechanism. Computer vision draws on the attention mechanism of natural language processing, and produces many wonderful results. SeNet [31] relies on context to automatically obtain the importance of each feature channel through self-learning. CBAM [32] combines spatial and channel attention mechanisms. Compared to SeNet, which only focuses on channels, CBAM can achieve better results. Based on CBAM's dual-path attention, DANet [33] directly uses non-local autocorrelation matrices for operations, avoiding tedious operations. CCNet [34] proposes a novel vertical and horizontal attention module that can be used to capture contextual information from remote dependencies in a more efficient way.

Boundary Supervision. Many studies have confirmed that boundary supervision can further sharpen and refine the edge contour prediction. CE2P [35] improves edge segmentation in a multi-task learning manner by introducing additional boundary supervision in human parsing task. ETNet [36] introduces boundary fine-grained restrictions in the encoder to guide feature extraction during medical segmentation. Contrary to the above, MSFNet [37] uses features extracted from the backbone to implement the boundary supervision with classes.

3. Methodology

The framework of our proposed method is illustrated in Figure 2. Specifically, it consists of the following four parts. The first part is the stage contextual attention mechanism (SCAM), which consists of vanilla convolution and the channel attention mechanism. The second part is the semantic gap compensation block (SGCB), which is motivated by the dilated convolution to increase the receptive field and bridge the semantic gaps at different scales. The third part is the boundary-aware (BA) module, which resorts to auxiliary edge supervision to strengthen the model's ability to recognize boundaries. In the last part, we dissect the overall architecture of EHANet and give the definition of the loss function.

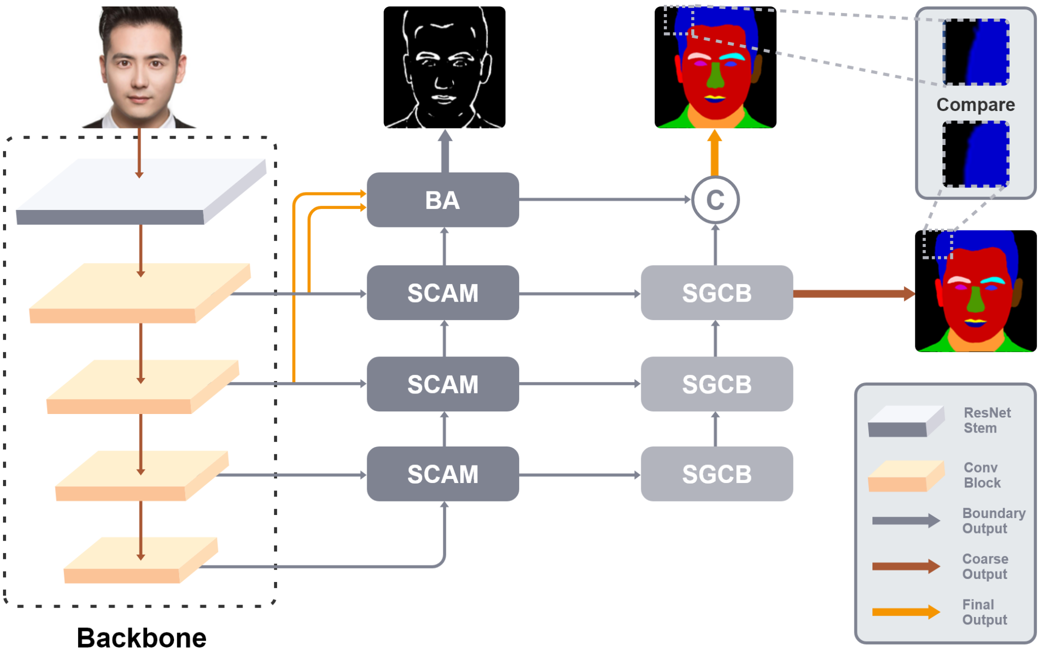


Figure 2. Diagram of our effective hierarchical aggregation network (best viewed in color). "C" stands for channel-wise concatenation. "SCAM" denotes stage contextual attention mechanism. "SGCB" denotes semantic gap compensation block. "BA" denotes boundary-aware module.

3.1. Stage Contextual Attention Mechanism

Faced with ambiguous semantic categories, model outputs are often misclassified or produce inconsistent parsing results. Depending on the interdependence between feature map channels, we can emphasize interdependent feature maps to improve the feature representation of a particular category. Partly inspired by the work of SeNet [31], we introduce a stage attention based on channel-wise attention. Unlike SeNet only using the current context, we use higher-level features with richer semantics to provide contextual guidance for lower-level ones. Based on the above observations, we extract the high-level context features of stage-$(n+1)$ to facilitate learning of the low-level features of stage-n, which is called stage contextual attention mechanism (SCAM). It should be noted that "stage" represents different convolutional blocks in Figure 2.

As illustrated in Figure 3, given a high-level input feature map $f_H^{in} \in \mathbb{R}^{C\times H\times W}$, we first feed it into a global pooling and two 1×1 shrink and expand convolutions (Add *relu* and *sigmoid* layers sequentially) to generate a global feature map f_G, where $f_G \in \mathbb{R}^{C\times 1\times 1}$. Then, feature map $f_L^{in} \in \mathbb{R}^{C\times h\times w}$ is obtained from the low-level input feature map followed by another 1×1 convolution.

In particular, 1 × 1 convolution ensures the consistency of the channel while reducing the number of calculations needed. After that we perform an element-wise matrix multiplication between f_L^{in} and f_G, and then *Add* with f_H^{in} through bilinear upsampling as $U(\cdot)$ to get the output feature map $f_L^{out} \in \mathbb{R}^{C\times h\times w}$. Overall, it can be written as:

$$f_L^{out} = f_L^{in} \odot f_G + U(f_H^{in}) \tag{1}$$

where $\odot$ denotes element-wise multiplication.

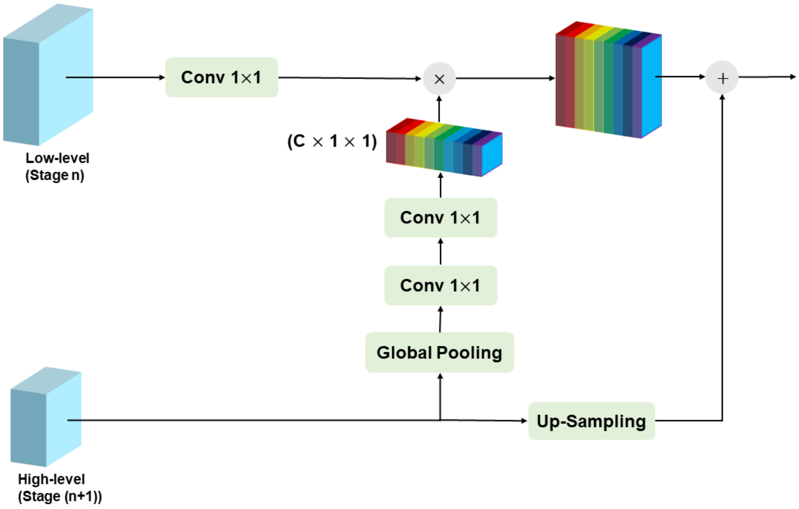
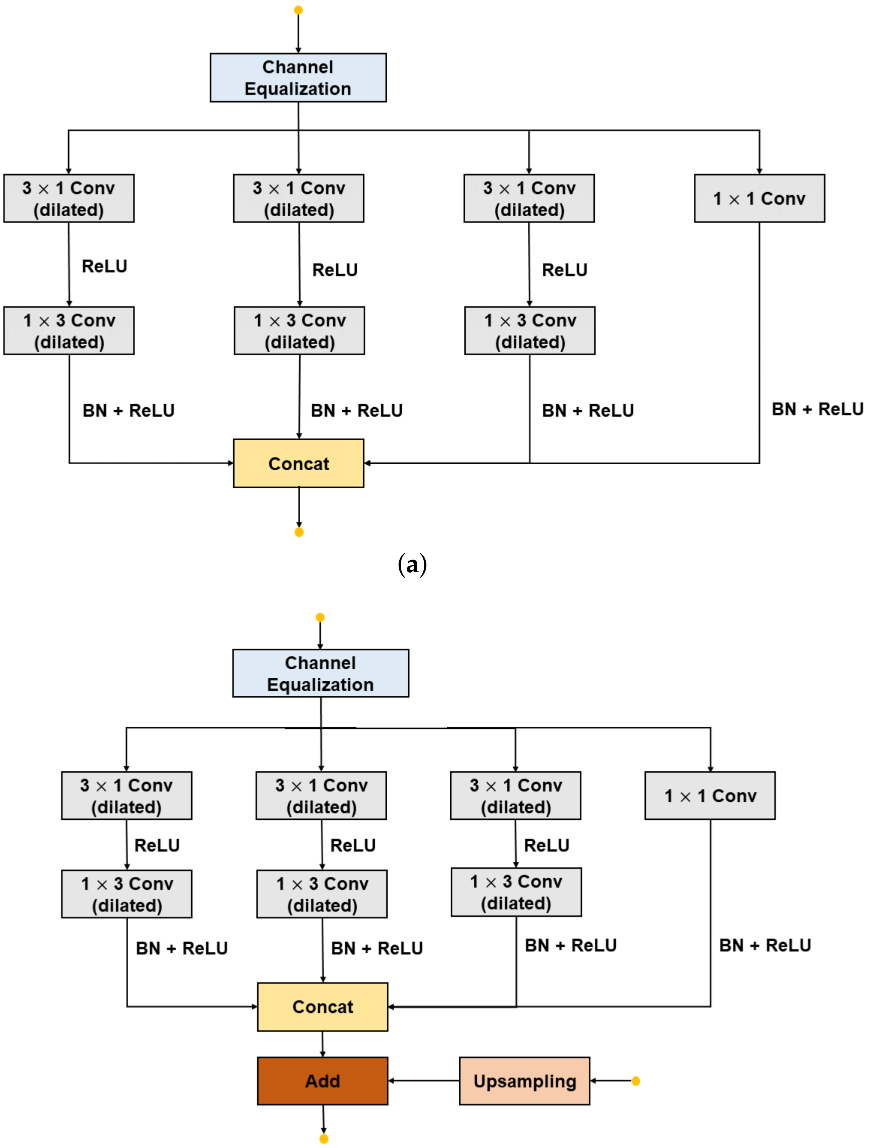


Figure 3. Stage contextual attention mechanism. "Conv" denotes convolutional operation.

3.2. Semantic Gap Compensation Block

In the networks with encoder-decoder architecture, skip connections are often employed to fuse shallow and deep layers. ExFuse [30] points out that simple fusion of deep and shallow layers of the network may lead to semantic gaps, which should be alleviated in order to obtain robust multi-scale features. Motivated by this work, we construct a semantic gap compensation block (SGCB) to compensate semantic gaps in a hierarchical aggregation manner. Our insight is that shallow layers utilize receptive field enhancement to bridge the semantic gap with deep layers. In detail, it contains two points: (1) increasing the receptive field can capture richer context to enhance the shallow representation ability; (2) by adjusting different rates, the receptive fields in the deep and shallow layers are generally consistent.

ERFNet [38] indicates that 1D factorized convolution greatly reduces the number of parameters (only 33% of the conventional convolution), while the accuracy is close to 2D convolution. Motivated by ERFNet and Inception series [6,7], we design an efficient SGAB module (Figure 4a) that captures hierarchical features of different scales in parallel. ShuffleNet v2 [21] points out that the FLOPs (float-point operations) are the smallest when the dimensions of the input and output channels are the same. Following this principle, we use *Channel Equalization* module (channel dimensions of different branches are equally divided) and *Concatenation* module to guarantee the consistency of the input and output channels. Take the first branch as an example, which consists of $\{3\times 1,\ 1\times 3\}$ convolution pairs, where each convolution is equipped with a dilation rate r_i that determines the span of the convolution interval. Setting different r_i values (i.e., $(1,\ 2,\ 4)$) for different branches can acquire multi-scale context. For stage-level feature fusion (Figure 4b), *Add* is used to perform element-wise summation of feature maps at different stages after semantic compensation.



(**a**)

(**b**)

Figure 4. Semantic gap compensation block. Among them, (**a**) is for the current layer, and (**b**) is employed for the fusion between different stages. Notation: "Conv" is convolutional operation. "BN" denotes the batch normalization. "Dilated" means dilated convolution.

3.3. Boundary-Aware Module

Similarly to [35], in order to strengthen the boundary information, we introduce a boundary-aware (BA) module. To our knowledge, shallow features contribute to boundary perception. Based on this, we leverage deep separable convolutions to efficiently extract boundary features and project them to the boundary perception space.

Due to weak semantics, boundary pixels are often confused with neighboring pixels belonging to different categories, which results in unsatisfactory segmentation results. To alleviate this problem, we propose two strategies. One strategy is to specify a boundary map for boundary supervision to highlight the influence of boundary pixels. Another strategy is to combine the BA branch and coarse-segmentation branch as input to get rich semantics while retaining boundary details.

3.4. EHANet for Segmentation

The loss functions can be described as:

$$L_s = -\frac{1}{N}\sum_{n \in N}\sum_{i,j} p_n^s(i,j) \cdot \log(\hat{p_n^s}(i,j)) \tag{2}$$

$$L_b = \begin{cases} -\sum\limits_{i,j} \beta \cdot \log(\hat{p^b}(i,j)) & p^b(i,j) = 1, \\ -\sum\limits_{i,j} (1-\beta) \cdot \log(1-\hat{p^b}(i,j)) & p^b(i,j) = 0 \end{cases} \tag{3}$$

$$L_c = -\frac{1}{N}\sum_{n \in N}\sum_{i,j} \cdot p_n^c(i,j) \cdot \log(\hat{p_n^c}(i,j)) \tag{4}$$

where L_s, L_b, and L_c refer to the losses of the coarse-segmentation, boundary-aware, and combined-segmentation branches, respectively. $\hat{p}$ and p represent the N-channel confidence map and the N-channel ground truth, respectively. (i, j) denotes the 2D coordinates of the pixel. Based on the binary cross-entropy, boundary-aware loss allocates an appropriate weight ratio β to alleviate the imbalance of foreground and background categories. According to strategy 1, contrary to CE2P [35], we introduce a coefficient θ to consolidate the influence of the boundary. As shown in Equations (5) and (6), a positive α is set to enforce the weight of boundary pixels, which is called weighted boundary-aware loss.

$$L_w = -\frac{1}{N}\sum_{n \in N}\sum_{i,j} \theta \cdot p_n^c(i,j) \cdot \log(\hat{p_n^c}(i,j)) \tag{5}$$

$$\theta = \begin{cases} 1+\alpha & p^b(i,j) = 1, \\ 1 & p^b(i,j) = 0 \end{cases} \tag{6}$$

$$L_{total} = \lambda_s \cdot L_s + \lambda_b \cdot L_b + \lambda_w \cdot L_w \tag{7}$$

L_{total} denotes the total loss function. λ_s, λ_b, and λ_w are utilized to balance the losses during training. The parameters Θ of the EHANet are optimized by minimizing Equation (7), which are formulated as $\min_\Theta L = L_{total}$. During the training process, λ_s, λ_b, and λ_c are empirically set to 1, 1, and 2 respectively. Moreover, α is experimentally set to 50. For a more detailed discussion of α, refer to Section 5.3.1.

To strike a balance between speed and performance, we built a lightweight network based on a truncated ResNet18 [9] backbone, which was pretrained on ImageNet. The ResNet [9] backbone is composed of a stem block and four bottlenecks. Among them, the stem block contains a 7 × 7 convolutional layer and a max pooling layer, each of which reduces the dimensions to 1/2. Except for the first of the four bottlenecks, the steps of the remaining bottlenecks are unified at 2. In addition, each bottleneck consists of two 3×3 convolutions and one skip connection. It is worth noting that the default paradigm is *Conv* + *BN* + *Relu*, except when feature fusion is required. Therefore, the total down-sampling rate of the network is 32.

Considering that ResNet Stem lacks context, the stage contextual attention mechanism (SCAM) is only laterally connected with stage-1, stage-2, and stage-3, just like the structure of a feature pyramid network (FPN) [8]. Subsequently, an semantic gap compensation block (SGCB) module is added after each SCAM. Moreover, both the reduction factor and the expansion factor in SCAM module are 8. The dilation rates r_i in the three SGCBs are (1, 2, 4), (3, 6, 9), and (7, 9, 13). Regardless of SCAM or SGCB, the number of output channels is set to 256. For the auxiliary boundary-aware (BA) module, we introduce two parallel paths from stage-1 and stage-2 to extract detailed features, and then concatenate them after the up-sampling operation. More specifically, the parallel branches all go through a 3×3 deep separable convolution with stride 1 and a 1×1 conventional convolution to refine the features. On the one hand, this module is used as auxiliary supervision. On the other hand, after 3 × 3 convolution, it performs fine-segmentation along with the coarse-segmentation sub-network. Benefiting from this framework, the fusion segmentation model not only contains rich semantics, but also refines the boundaries.

4. Experiment Setup

4.1. Datasets

CelebAMask-HQ. CelebAMask-HQ dataset [10] is a large-scale facial semantic understanding dataset with a resolution of 512 × 512. It consists of 30,000 manually fine labeled data involving 19 classes. We refer to CelebA-HQ [39] and divide it into 24,183/2993/2824 images for training, validation, and testing.

Helen. Helen dataset [11] is a challenging facial parsing dataset that contains 11 semantic classes; i.e., hair, eyes, lips, *etc*. The training, validation, and testing sets consist of 2330, 100, and 300 images respectively.

4.2. Implementation Details

Our experiments are conducted using Pytorch v1.4.0 framework with CUDA and CuDNN backends. The *baselr* is set to 0.001 for CelebAMask-HQ and 0.007 for Helen. Standard mini-batch gradient descent is employed as the optimizer with the momentum of 0.9, weight decay of 1×10^{-5} and batch size of 16. We adopt the widely equipped *poly* training strategy where the *baselr* is multiplied by $(1 - \frac{iter}{total\_iter})^{0.9}$ after each iteration. To avoid over-fitting, common data augmentations are used, including random horizontal flip, random color jittering, random scaling in the range of [0.5, 2], and random crop image patches. We train the model for 150 epochs on CelebAMask-HQ dataset, and 200 epochs on Helen. Additionally, the experimental environment is equipped with a 3.60 GHz CPU and a Nvidia GTX 1080Ti graphics card. Our code is available on GitHub (https://github.com/JACKYLUO1991/FaceParsing).

4.3. Evaluation Metrics

We adopt the most commonly used Pixel Acc. (pixel accuracy) [4] and mIoU (mean intersection of union) [4] for CelebAMask-HQ and F1-score [11] for Helen to evaluate the model's performance. The mathematical expression of Pixel Acc. and mIoU can be written as:

$$Pixel\ Acc. = \frac{\sum_{i=0}^{k} p_{ii}}{\sum_{i=0}^{k} \sum_{j=0}^{k} p_{ij}} \tag{8}$$

$$mIoU = \frac{1}{k+1} \sum_{i=0}^{k} \frac{p_{ii}}{\sum_{j=0}^{k} p_{ij} + \sum_{j=0}^{k} p_{ji} - p_{ii}} \tag{9}$$

where $k+1$ is the number of classes (including background); p_{ij} indicates the number of pixels that belong to category i but have been misjudged as category j.

Precision and recall can be defined for each class and F1-score is the harmonic mean of them, with an expression of:

$$F1 = 2 \times \frac{precision \cdot recall}{precision + recall} \tag{10}$$

5. Results and Discussion

Unless otherwise stated, all experiments adopt ResNet18 backbone as a benchmark. In the following sections, we first evaluate performance with state-of-the-art methods on `test` sets of

CelebAMask-HQ and Helen. Then, we further conduct ablation studies on CelebAMask-HQ `validation` set to confirm the effectiveness of our method.

5.1. Results on CelebAMask-HQ

In this subsection, we compare our algorithm with several recently published methods, including DFANet [23], DANet [33], DABNet [24], CE2P [35], and UNet [5] on the CelebAMask-HQ `test` set. For fair comparison, we re-implement the above frameworks under the same hardware configuration without using extra training data or multi-scale testing.

As depicted in Table 1 and Figure 5, our solution outperforms others by a large margin. Surprisingly, our approach surpassed CE2P (the winner of LIP Challenge 2018) (http://sysu-hcp.net/lip/index.php) by 0.32% mIoU (78.19% vs. 77.87%) while halving the running time. Restricted by the cumbersome ResNet101, CE2P is not suitable for real-time segmentation. On the contrary, benefiting from the lightweight ResNet18, our network achieved comparable results with CE2P while ensuring low latency. By modifying the backbone network to ResNet34, the accuracy of EHANet is much better than CE2P, which reflects that CE2P has a large number of redundant parameters. Moreover, it can be observed that compared with the fastest method DABNet, the accuracy of our method is more than 10% higher than the former. To further evaluate the effect, detailed per-category comparisons are also reported in Table 1, where our method achieves the highest IoU on 10 out of 19 categories. Overall, the improvements over the state-of-the-art methods confirm the effectiveness of our EHANet for face parsing.

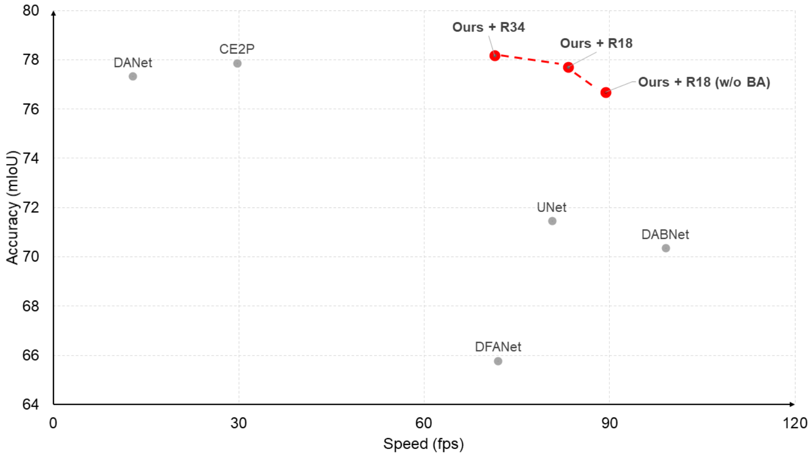


Figure 5. Runtime and mIoU (mean intersection of union) on CelebAMask-HQ. "R" denotes ResNet backbone (i.e., R18 is equivalent to ResNet18). Our method achieved remarkable results in both accuracy and efficiency.

Lightweight models often lack context, resulting in poor segmentation for small objects. I.e., neither DFANet nor UNet detected a "necklace." In contrast, our network can well alleviate this problem through the channel-wise attention mechanism in order to better encode the semantic categories. The qualitative results on CelebAMask-HQ `test` set are presented in Figure 6. From the dotted rectangle, we can observe that our results are smoother and more natural.

Table 1. Category-wise comparison (IoU) on the CelebAMask-HQ `test` set. The best performance for each individual class is marked with bold-face number. "BA" denotes boundary-aware branch, "†" denotes EHANet with ResNet34 backbone, "FPS" denotes frames per second. Resolution is unified to 512 × 512.

Methods	Background	Skin	Nose	Eye-Glass	Left-Eye	Right-Eye	Left-Brow	Right-Brow	Left-Ear	Right-Ear	Mouth
DFANet [23]	88.12	89.69	83.96	69.29	69.19	69.12	66.04	66.05	66.52	65.85	70.23
DABNet [24]	90.36	91.27	86.04	72.62	73.69	74.38	69.78	69.25	72.28	70.83	78.41
UNet [5]	89.36	92.25	87.83	77.64	79.45	79.75	74.17	73.98	77.28	76.00	83.47
DANet [33]	**93.04**	93.15	88.92	84.04	80.65	80.84	75.63	75.37	**78.93**	**78.29**	85.45
CE2P [35]	92.78	93.17	88.54	84.44	**81.95**	**82.03**	75.55	75.51	78.35	77.72	85.65
EHANet(Ours)	91.80	92.90	88.57	84.00	81.58	81.92	75.02	74.93	78.25	77.90	85.55
EHANet(Ours) + BA	92.76	93.16	88.73	84.62	81.70	81.96	75.54	75.48	78.58	77.55	85.62
EHANet(Ours) + BA †	92.98	**93.26**	**89.02**	**84.91**	81.85	82.02	**75.66**	**75.52**	78.76	78.00	**85.69**

Methods	Upper-Lip	Lower-Lip	Hair	Hat	Earring	Necklace	Neck	Cloth	FPS	mIoU (%)	Pixel Acc. (%)
DFANet [23]	66.17	70.26	86.25	61.02	25.68	0.00	76.38	60.21	72	65.79	91.85
DABNet [24]	73.91	78.37	88.52	63.56	36.23	0.01	78.74	67.27	**99**	70.36	93.24
UNet [5]	79.32	82.00	88.16	32.62	42.58	0.00	79.92	61.88	81	71.46	93.00
DANet [33]	80.41	83.52	**91.31**	75.87	**52.78**	9.46	83.55	**78.07**	13	77.33	95.09
CE2P [35]	80.89	83.57	91.18	76.13	52.47	19.73	83.53	76.30	30	77.87	95.00
EHANet(Ours)	80.98	83.25	90.31	74.26	48.11	13.16	81.38	73.30	89	76.69	94.51
EHANet(Ours) + BA	80.96	**83.60**	91.08	76.30	51.82	17.61	83.37	76.10	83	77.71	94.96
EHANet(Ours) + BA †	**81.04**	83.47	91.26	**77.05**	51.84	**20.19**	**84.15**	77.93	71	**78.19**	**95.28**

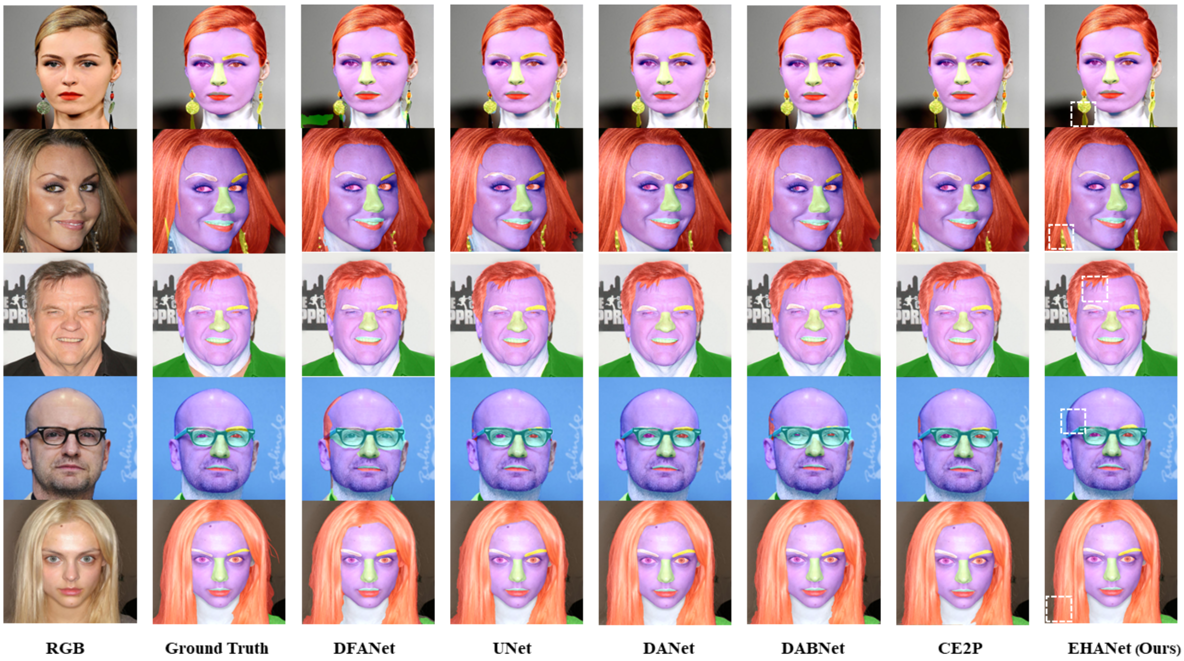


Figure 6. Qualitative comparisons on CelebAMask-HQ. Zoom in for more details in electronic version.

5.2. Results on Helen

To validate the generalization ability of our method, we further report the comparison between our model and existing face parsing methods on the Helen `test` set. For a fair comparison with previous work, we refer to the preprocessing step in [15]. Furthermore, F1-score is calculated by combining eyebrow, eye, mouth, and nose categories.

It can be observed from Table 2 that our model outperforms all other models except the method proposed by Lin et al. [16]. Lin's method introduced the novel RoI tanh-warping, which has the disadvantage of taking up too many resources and having high-resolution input requirements. On the other hand, it is worth noting that when model complexity is relatively low, EHANet still boosts the performance by 0.2% (90.7% vs. 90.5%) compared to VGG-based RED-Net [15], which indicates that EHANet is more compact and efficient.

Table 2. Results on the Helen `test` dataset. * Denotes additional data processing.

Methods	Year	Overall F1-Score (%)
Smith et al. [11]	2013	80.4
Liu et al. [13]	2015	84.7
Liu et al. [14]	2017	88.6
Guo et al. [15]	2018	90.5
Lin et al. * [16]	2019	**92.4**
Ours	2020	90.7

5.3. Ablation Study

In this subsection, we perform ablation experiments to illustrate the effectiveness of each component of our EHANet. To quickly verify the experimental results, the default image resolution is 256×256.

5.3.1. Ablation Studies on Weighted Boundary-Aware Loss

To evaluate the effectiveness of our proposed weighted boundary-aware (WBA) loss, we construct ablation experiments on it. WBA loss utilizes strong supervisory information to enhance the ability to distinguish between adjacent categories. Equation (5) gives the supervision loss of the WBA, where θ indicates the impact of boundary on the overall loss, and the magnitude of θ is determined by α. The mIoU scores are reported in Table 3. When $\theta \equiv 1$, L_w (=L_c) becomes the vanilla cross-entropy loss, which treats different categories equally, making it difficult to distinguish between "hard samples". The mIoU value at this time is the lowest, only 72.34%. As α gradually increases, mIoU shows a ridge trend (72.71% → 73.00% → 72.83%) and reaches the highest value at 50. The reasons can be briefly described as follows: (1) when α is less than 30, the model does not fully learn the boundary knowledge; (2) when α is greater than 70, the model focuses on boundary, while neglecting the learning of the main task. Therefore, proper adjustment of this hyper-parameter has a promoting effect on the results. All of the above confirm the effectiveness of our boundary supervisory strategy for robust feature learning.

Table 3. Performance comparison with and without weighted boundary-aware loss.

WBA Loss	α	mIoU (%)
✗	-	72.34
✓	30	72.71
✓	50	**73.00**
✓	70	72.83

5.3.2. Ablation Studies on Each Component

We use the feature pyramid network (FPN) [8] equipped with ReNet18 backbone as the baseline, which restores the input size by bilinear up-sampling on the P2 layer, resulting in a mIoU of 71.02%. In the experiment, the effect was observed by gradually replacing the lateral connection layer and convolution fusion layer in the FPN with the stage contextual attention mechanism module and semantic gap compensation block, and adding other components. We report the quantitative comparison results in Table 4. On the one hand, the stage contextual attention mechanism module can provide rich context, and the semantic gap compensation block reduces the gap between semantics. Adding these two modules improves the performance by 0.37% and 0.49%, respectively. On the other hand, the boundary-aware module explicitly introduces boundary features into the latent space of features, and strengthens the ability to represent boundary details, which greatly improves the performance of our model by 0.54%. The square dashed box in Figure 2 provides an intuitive comparison of whether to add a boundary-aware branch. When we further introducing weighted boundary-aware loss, we obtain the optimal result, a mIoU of 73%. The latter two illustrate the effect of boundary processing on improving performance. Figure 7 shows the qualitative comparison results, where our model gets more consistent results for objects of the same category and keeps more detailed information, benefiting from our proposed components.

Table 4. Ablation experiments of EHANet on CelebAMask-HQ `validation` set. "SCAM" denotes stage contextual attention mechanism module, "SGCB" denotes semantic gap compensation block, "BA" denotes boundary-aware branch, "WBA loss" denotes weighted boundary-aware loss.

Methods	mIoU (%)
baseline	71.02
baseline + SCAM	71.39
baseline + SCAM + SGCB	71.88
baseline + SCAM + SGCB + BA	72.34
baseline + SCAM + SGCB + BA + WBA loss (ours)	**73.00**

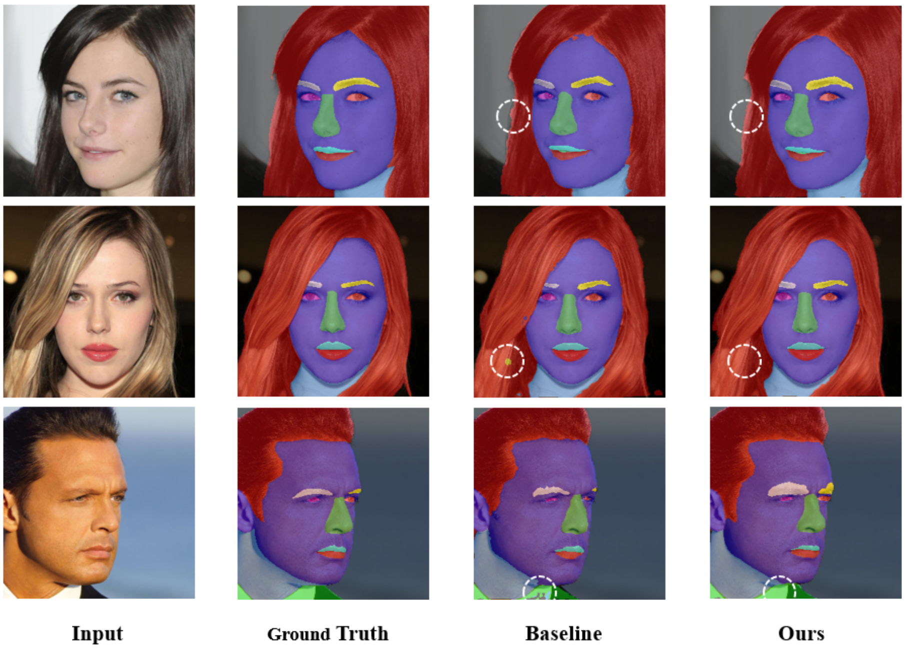


Figure 7. Face parsing results comparison against FPN-ResNet18 [8] baseline, where significantly improved regions are marked with white dashed boxes. To highlight the details, we merge the input and output color maps according to 1:1. Our method performs better on both boundary details and fine-grained segmentation.

5.4. Efficiency and Accuracy

We carry on the experiments to demonstrate the potential of our segmentation architecture in terms of accuracy and efficiency trade-off. Since our goal is to design an efficient and universal framework, the impact of different input resolutions and different backbones on model performance needs to be given quantitatively. As shown in Table 5, we note that deeper network tends to perform better (ResNet18 → 73.00%, ResNet101 → 75.35%). However, the weakness is that the amount of floating-point calculations increases exponentially (3.1 G vs. 11.5 G). From another perspective, the increased input brings rich spatial details. Although it can improve the accuracy of the model, it brings high latency. Moreover, as the image dimension gradually rises from 256 to 640, the performance gain becomes inconspicuous (e.g., an increase of 2.36% from 256 to 384 but only an increase of 0.53% from 512 to 640), indicating that the model tends to be saturated.

Regardless of different backbones or different resolutions, our EHANet consistently brings consistent positive gains in terms of mIoU, which suggests the scalability of our proposed method. In particular, the fastest setting of our method runs at a speed of 313 FPS at mIoU 73.00%. The whole comparative experiment provides a reference for how to choose the appropriate model in real-world environment.

Table 5. Performance comparison of models at different resolutions and different backbones. $M = 10^6$, $G = 10^9$. "-" represents the same statistics as above.

Methods	Input Size	FLOPs	FPS	#Params	mIoU (%)
EHANet + ResNet18	256 × 256	**3.1**G	**313**	**11.8**M	73.00
EHANet + ResNet18	384 × 384	7.1G	143	-	75.36
EHANet + ResNet18	512 × 512	12.5G	83	-	76.51
EHANet + ResNet18	640 × 640	19.6G	55	-	**77.04**
EHANet + ResNet34	256 × 256	5.6G	294	21.9M	74.77
EHANet + ResNet50	256 × 256	6.7G	204	25.0M	75.21
EHANet + ResNet101	256 × 256	11.5G	167	43.9M	75.35

6. Conclusions

In this paper, we present an effective hierarchical aggregation network named EHANet for real-time face parsing. Firstly, to capture long-term dependencies to enhance the discrimination of different categories, we propose a stage contextual attention mechanism module. Next, we introduce a semantic gap compensation block to bridge the semantic gap caused by feature fusion at different stages. Finally, we use weighted boundary-aware loss to force the model to distinguish between adjacent categories. Our method has achieved remarkable results on both the CelebAMask-HQ and Helen datasets, proving the robustness of our network. Subsequent ablation experiments further confirmed the effectiveness of the proposed method. The advantage of low latency makes our method further applicable to mobile deployment.

Our future work includes exploring the effect of weakly supervised signals on segmentation performance.

Author Contributions: Conceptualization, L.L. and D.X.; methodology, L.L. and D.X.; software, L.L.; validation, L.L. and X.F.; formal analysis, L.L.; investigation, L.L. and X.F.; resources, D.X.; data curation, L.L.; writing—original draft preparation, L.L.; writing—review and editing, D.X.; visualization, L.L.; supervision, D.X.; project administration, D.X. All authors have read and agreed to the published version of the manuscript.

Acknowledgments: This work was done when Ling Luo was an intern at Meidaojia Research, Beijing, P.R. China. L.L thanks their support for this work.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Ou, X.; Liu, S.; Cao, X.; Ling, H. Beauty emakeup: A deep makeup transfer system. In Proceedings of the 24th ACM International Conference on Multimedia, Amsterdam, The Netherlands, 15–19 October 2016; pp. 701–702.
2. Wang, T.C.; Liu, M.Y.; Zhu, J.Y.; Tao, A.; Kautz, J.; Catanzaro, B. High-resolution image synthesis and semantic manipulation with conditional gans. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 19–21 June 2018; pp. 8798–8807.
3. Zhang, D.; Lin, L.; Chen, T.; Wu, X.; Tan, W.; Izquierdo, E. Content-adaptive sketch portrait generation by decompositional representation learning. *IEEE Trans. Image Process.* **2016**, *26*, 328–339. [CrossRef] [PubMed]
4. Long, J.; Shelhamer, E.; Darrell, T. Fully convolutional networks for semantic segmentation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Boston, MA, USA, 7–12 June 2015; pp. 3431–3440.
5. Ronneberger, O.; Fischer, P.; Brox, T. U-net: Convolutional networks for biomedical image segmentation. In Proceedings of the International Conference on Medical Image Computing and Computer-Assisted Intervention, Munich, Germany, 5–9 October 2015; pp. 234–241.
6. Szegedy, C.; Ioffe, S.; Vanhoucke, V.; Alemi, A. A. Inception-v4, inception-resnet and the impact of residual connections on learning. In Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence, San Francisco, CA, USA, 4–9 February 2017; pp. 4278–4284.
7. Szegedy, C.; Liu, W.; Jia, Y.; Sermanet, P.; Reed, S.; Anguelov, D.; Erhan. D.; Vanhoucke. V.; Rabinovich, A. Going deeper with convolutions. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Boston, MA, USA, 7–12 June 2015; pp. 1–9.
8. Lin, T. Y.; Dollár, P.; Girshick, R.; He, K.; Hariharan, B.; Belongie, S. Feature pyramid networks for object detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 2117–2125.
9. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep residual learning for image recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 26 June–1 July 2016; pp. 770–778.
10. Lee, C. H.; Liu, Z.; Wu, L.; Luo, P. MaskGAN: Towards diverse and interactive facial image manipulation. *arXiv* **2019**, arXiv:1907.11922.
11. Smith, B.M.; Zhang, L.; Brandt, J.; Lin, Z.; Yang, J. Exemplar-based face parsing. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Oregon, Portland, 25–27 June 2013; pp. 3484–3491.

12. Kae, A.; Sohn, K.; Lee, H.; Learned-Miller, E. Augmenting CRFs with Boltzmann machine shape priors for image labeling. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Oregon, Portland, 25–27 June 2013; pp. 2019–2026.
13. Liu, S.; Yang, J.; Huang, C.; Yang, M.H. Multi-objective convolutional learning for face labeling. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Boston, MA, USA, 7–12 June 2015; pp. 3451–3459.
14. Liu, S.; Shi, J.; Liang, J.; Yang, M. H. Face parsing via recurrent propagation. *arXiv* **2017**, arXiv:1708.01936.
15. Guo, T.; Kim, Y.; Zhang, H.; Qian, D.; Yoo, B.; Xu, J.; Zou. D.; Han. J.; Choi, C. Residual Encoder Decoder Network and Adaptive Prior for Face Parsing. In Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence, New Orleans, LA, USA, 2–7 February 2018; pp. 6861–6869.
16. Lin, J.; Yang, H.; Chen, D.; Zeng, M.; Wen, F.; Yuan, L. Face Parsing with RoI Tanh-Warping. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 16–20 June 2019; pp. 5654–5663.
17. Iandola, F.N.; Han, S.; Moskewicz, M.W.; Ashraf, K.; Dally, W.; Keutzer, K. SqueezeNet: AlexNet-level accuracy with 50x fewer parameters and <0.5 MB model size. *arXiv* **2016**, arXiv:1602.07360.
18. Howard, A.G.; Zhu, M.; Chen, B.; Kalenichenko, D.; Wang, W.; Weyand, T.; Andreetto. M.; Adam, H. MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications. *arXiv* **2017**, arXiv:1704.04861.
19. Sandler, M.; Howard, A.; Zhu, M.; Zhmoginov, A.; Chen, L. Mobilenetv2: Inverted residuals and linear bottlenecks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, MA, USA, 19–21 June 2018; pp. 4510–4520.
20. Zhang, X.; Zhou, X.; Lin, M.; Sun, J. Shufflenet: An extremely efficient convolutional neural network for mobile devices. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, MA, USA, 19–21 June 2018; pp. 6848–6856.
21. Ma, N.; Zhang, X.; Zheng, H. T.; Sun, J. Shufflenet v2: Practical guidelines for efficient cnn architecture design. In Proceedings of the European Conference on Computer Vision (ECCV), Munich, Germany, 8–14 September 2018; pp. 116–131.
22. Chen, Y.; Fan, H.; Xu, B.; Yan, Z.; Kalantidis, Y.; Rohrbach, M.; Yan. S.; Feng, J. Drop an octave: Reducing spatial redundancy in convolutional neural networks with octave convolution. In Proceedings of the IEEE International Conference on Computer Vision, Long Beach, CA, USA, 16–20 June 2019; pp. 3435–3444.
23. Li, H.; Xiong, P.; Fan, H.; Sun, J. Dfanet: Deep feature aggregation for real-time semantic segmentation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 16–20 June 2019; pp. 9522–9531.
24. Li, G.; Yun, I.; Kim, J.; Kim, J. DABNet: Depth-wise Asymmetric Bottleneck for Real-time Semantic Segmentation. *arXiv* **2019**, arXiv:1907.11357.
25. Chollet, F. Xception: Deep learning with depthwise separable convolutions. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 1251–1258.
26. He, Y.; Zhang, X.; Sun, J. Channel pruning for accelerating very deep neural networks. In Proceedings of the IEEE International Conference on Computer Vision, Venice, Italy, 22–19 October 2017; pp. 1389–1397.
27. Yu, F.; Koltun, V. Multi-scale context aggregation by dilated convolutions. *arXiv* **2015**, arXiv:1511.07122.
28. Zhao, H.; Shi, J.; Qi, X.; Wang, X.; Jia, J. Pyramid scene parsing network. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 2881–2890.
29. Zhang, F.; Chen, Y.; Li, Z.; Hong, Z.; Liu, J.; Ma, F.; Han. J.; Ding, E. ACFNet: Attentional class feature network for semantic segmentation. In Proceedings of the IEEE International Conference on Computer Vision, Seoul, Korea, 27 October–2 November 2019; pp. 6798–6807.
30. Zhang, Z.; Zhang, X.; Peng, C.; Xue, X.; Sun, J. Exfuse: Enhancing feature fusion for semantic segmentation. In Proceedings of the European Conference on Computer Vision (ECCV), Munich, Germany, 8–14 September 2018; pp. 269–284.
31. Hu, J.; Shen, L.; Sun, G. Squeeze-and-excitation networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, MA, USA, 19–21 June 2018; pp. 7132–7141.
32. Woo, S.; Park, J.; Lee, J. Y.; So Kweon, I. Cbam: Convolutional block attention module. In Proceedings of the European Conference on Computer Vision (ECCV), Munich, Germany, 8–14 September 2018; pp. 3–19.

33. Fu, J.; Liu, J.; Tian, H.; Li, Y.; Bao, Y.; Fang, Z.; Lu, H. Dual attention network for scene segmentation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 16–20 June 2019; pp. 3146–3154.
34. Huang, Z.; Wang, X.; Huang, L.; Huang, C.; Wei, Y.; Liu, W. Ccnet: Criss-cross attention for semantic segmentation. In Proceedings of the IEEE International Conference on Computer Vision, Seoul, Korea, 27 October–2 November 2019; pp. 603–612.
35. Ruan, T.; Liu, T.; Huang, Z.; Wei, Y.; Wei, S.; Zhao, Y. Devil in the details: Towards accurate single and multiple human parsing. In Proceedings of the AAAI Conference on Artificial Intelligence, Honolulu, HI, USA, 27 January–1 February 2019; pp. 4814–4821.
36. Zhang, Z.; Fu, H.; Dai, H.; Shen, J.; Pang, Y.; Shao, L. ET-Net: A Generic Edge-aTtention Guidance Network for Medical Image Segmentation. In Proceedings of the International Conference on Medical Image Computing and Computer-Assisted Intervention, Shenzhen, China, 13–17 October 2019; pp. 442–450.
37. Si, H.; Zhang, Z.; Lv, F.; Yu, G.; Lu, F. Real-Time Semantic Segmentation via Multiply Spatial Fusion Network. *arXiv* **2019**, arXiv:1911.07217.
38. Romera, E.; Alvarez, J. M.; Bergasa, L. M.; Arroyo, R. Erfnet: Efficient residual factorized convnet for real-time semantic segmentation. *IEEE Trans. Intell. Transp. Syst.* **2017**, *19*, 263–272. [CrossRef]
39. Karras, T.; Laine, S.; Aila, T. A style-based generator architecture for generative adversarial networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 16–20 June 2019; pp. 4401–4410.