Deep Feature Attention and Aggregation for Real-Time Head Semantic Segmentation

Zhiqin Zhang¹, Bo Zhang²*
School of Computer Science, Wuhan Donghu University, Wuhan 430212, China
*Corresponding author’s e-mail: bob.cheung@ovspark.com

Abstract. This paper proposes a mobile CNN(Convolutional Neural Networks) architecture named FAANet(Feature Attention Aggregation Net), which focused on self-attention and feature aggregation for real time head semantic segmentation using limited computation resource. The proposed network is based on a single lightweight backbone cnn network and aggregates discriminative features through multiple different scale feature maps, and the different feature maps are fused using self-attention sub-block. Using the proposed multi-scale feature communication and propagation, FAANet substantially reduces the size of model, but still achieves big receptive field and strengthen the generalization ability, which achieves a balance between the computation cost and performance. Experiments on head segmentation dataset demonstrate the improved performance of FAANet with 10 times less FLOPs and 3 times faster than the existing state-of-the-art fast semantic segmentation methods and the proposed method can achieve comparable accuracy. Specifically, when computed on one NVIDIA Titan X card, it can achieve 98.3% Mean IOU on our mixed test dataset with only 1.2 GFLOPs and 100 FPS while inferring on a 1024*1024 input image.

1. Introduction
Semantic segmentation is a pixel-wise classification task, and it is a basic and important task in computer vision. A lot of practical applications has been developed in the fields of video effects, robot sensing, video surveillance, autonomous driving and so on. For most such applications, especially for video effects on mobile phone, how to keep real time inference speed and achieve good performance is important.

Some real-time semantic segmentation methods [1-5] have already been proposed and obtained good performances on various benchmarks [6-7]. However, it is time consuming if most of the operations are calculated on the high-resolution feature maps when using the U-shape backbone. Some researches decrease the computation complexity by reducing the input image size, distilling learning or pruning the feature map channels to suppress the computation cost. Though it is possible effective, they always damage the spatial details and can not achieve good segmentation results on the object boundary and small objects. Also, if a network is shallow, the feature discriminative ability will be weakened. In order to improve the segmentation results, it is important to selectively combine the spatial details and context information[4,8]. Nevertheless, it is not the best choice to add more layers on the high-resolution image which will cost more computation resource, and the limited connection between different scale layers will decrease the model learning and generalization ability.

Commonly, pretrained model will be used for semantic segmentation task, such as ResNet, googlenet, inception, DenseNet and so on. Aiming to save inference time, we adopt a lightweight modified model and research for improving the segmentation performance even with small model.
Progressively increasing the feature maps is a traditional method from the several last outputs of a single path architecture, in this kind of design, it is difficult to achieve fine segmentation results because the high-level context is not fused with the spatial level detail features. In order to improve the model learning capacity and expand the receptive field contemporary even in restrictive mobile device, feature reuse is an intuitive thought. So, it is necessary to find a lightweight method as well as incorporating multi-level context into encoded features by using self-attention fusion mechanism named SA-block.

In this paper, we proposed two methods to improve cross-scale feature fusion results. First, for fusing global semantic information and local structure details, we reuse high-level features, meanwhile, different feature maps from different stages was combined for enhanced feature representation ability. Second, we use SA-block selectively fuse different level features. Our whole network structure is visualized in Figure 1.

The proposed FAANet was tested on one standard benchmark dataset Cityscapes and our practical portrait videos dataset which include 500 short video clips. With a 1024×1024 input, FAANet achieves 98.3% Mean IOU with 1.2G FLOPs and speed of 100 FPS on a NVIDIA Titan X card. While implemented on a smaller input size such as 512×512, we achieved the 96.3% Mean IOU with only 0.5G FLOPs which is suitable for mobile application and comparable with most of the state-of-the-art real-time segmentation model.

2. Related Work

2.1 Time Limited Segmentation
Real-time semantic segmentation algorithms are focused on generating the SOTA comparative segmentation mask under limited calculation. ENet achieved good performance but sacrificed miou by reducing the number of downsampling times. Because of the erasing for the last stages of the model, the receptive field is too small and it is difficult to segment larger objects subtly. ICNet uses different scale image inputs and a cascade network was used to decrease computation cost. BiSeNet proposes to prune channels by using spatial path and semantic path to reduce calculation. In the two methods, there is only one branch is used for deep feature extraction, and the other branches are designed to extract different scale details. On the contrary, we obtained distinguished feature by multiple level feature aggregation and self-attention feature fusion.

2.2 Multiple Level Feature Aggregation
In traditional approaches, the encoder-decoder structure was used to solve pixel-to-pixel prediction. Relative to deep network, how to aggregate different scale features between blocks deserves more attention. RefineNet proposed a complex refine module in each magnifying stage before concatenating multi-scale features which is too time-consuming. DenseNet uses the dense connection to aggregate different feature for easy training.

3. Our Method
We start with our observation and research of calculation cost when applying traditional semantic segmentation strategy in our head segmentation task on mobile device. This motivates our multiple scale feature aggregation strategy based on self-attention to combine detail and spatial information in different level feature maps of the network to achieve comparable accuracy even in tiny model size. The whole architecture of Deep Feature Attention and Aggregation Network (FAANet) is illustrated in Figure 1. We focus on making the selectively fusion of different scale features in networks, it is important to aggregate different receptive field features for semantic segmentation task. Our aggregation strategy is composed of sub-network aggregate and using attention sub-block selectively fuse different level features.
3.1 How To Aggregate Feature At Tiny Backbone

The structure is composed with a coarse to fine encoder-decoder network which can refine different scale feature. Features aggregation implements mixture of different scale features at the network level. In general, we implement our architecture as a stack of feature refining sub-network by selectively feeding the output of the previous sub-network to the next. A sub-network process can be defined as \( y = \Phi(x) \), the output of sub-network \( \Phi_n \) is the input of sub-network \( \Phi_{n+1} \), so the whole network aggregate thought can be formulated as: \( Y = \Phi_n(\Phi_{n-1}(...\Phi_1(X))) \). Sub-network aggregation allows these high-level features to be reprocessed on account of low-level spatial feature, it is beneficial for further evaluating and reassessing higher order spatial relationships.

We also use multiple stage feature aggregation strategy, for a single sub-network \( \Phi_n(x) \), a stage process can be defined as \( \Phi_n \). The stage in the previous sub-network is \( \Phi_{i-1} \), \( i \) means the index of the stage. Sub-stage aggregation method can be formulated as:

\[
x_n^i = \begin{cases} 
  x_n^{i-1} + \phi_n^i(x_n^{i-1}) & \text{if } n = 1, \\
  x_n^{i-1} + \phi_n^i([x_n^{i-1}, x_{n-1}]) & \text{otherwise,}
\end{cases}
\]

(1)

While, \( x_{n-1}^{i} \) is coming from:

\[
x_{n-1}^i = x_{n-1}^{i-1} + \phi_{n-1}^i(x_{n-1}^{i-1})
\]

(2)

In our strategy, high-resolution features block will flow into the low-resolution block. Our formulation learns a new mapping of \( n \)th feature maps and fuses \((n-1)\)th features and receptive field. So, the full connection of feature Information flow can greatly improve the feature discrimination. Details was showed in Figure 1.

3.2 Self-Attention Block

First, the object context estimation will be calculated, \( p \) is the object context for each pixel and is defined as the probability if a set of pixels belong to the same category as \( p \). The object context map shows the probability if the pixel \( p \) and each other pixel belong to the same object. The computation of object context map is given as follows,

\[
\psi_{pi} = \frac{1}{Z_p} \exp(f_q(x_p)^T f_k(x_i)),
\]

(3)

where \( x_p \) and \( x_i \) are the representation vectors of the pixels \( p \) and \( i \), \( Z_p \) is the normalization number:

\[
Z_p = \sum_{i=1}^{N} \exp(f_q(x_p)^T f_k(x_i))
\]

Then, \( f_q(\cdot) \) and \( f_k(\cdot) \) are the query transform function and the key transform function which can be formed by convolution, matrix multiply and reshape operation. Second, object context aggregation is calculated, object context representation of the pixel \( p \) can be constructed by aggregating the representations of the pixels as follows,
where $\phi(\cdot)$ is the value transform function following the self-attention[9].

Our self-attention block was shown in Figure.2.

4.Experiments
we evaluate the proposed method on two challenging benchmarks: Cityscapes and our head test dataset, and it is a big challenge for real-time semantic segmentation. We first compare the accuracy and speed results on Cityscapes with the existing fast segmentation algorithms. After that, a practical experiment on head dataset was made to prove that the architecture is fit for real mobile application. We used the following training strategy: mini-batch stochastic gradient descent (SGD) with batch size 64, momentum 0.8 and weight decay $1e^{-6}$. For fair comparison, we used the traditional cross-entropy loss at each pixel. Data augmentation includes mean subtraction, resizing, cropping and random horizontal flip for training. Some segmentation results are shown in Figure.3.

In Table 1, Different structure was evaluated on Cityscapes val dataset.'Scale' means scaling ratio of input image. the results are shown, as can be observed, the miou metrics is kept comparable, while the inference speed of the proposed method is superior to state-of-the-art methods. The baseline of the proposed method achieves 98.3% Mean IOU on the our test dataset with only 1.2 GFLOPs and a speed of 100 FPS on one NVIDIA Titan X card while inferring on a 1024*1024 resolution image.

| Model     | Scale | FLOPS  | mIOU(%) |
|-----------|-------|--------|---------|
| ResNet-50 | 0.25  | 9.3G   | 64.5    |
| Xception A| 1.0   | 1.6G   | 59.2    |
| Xception B| 1.0   | 0.83G  | 55.4    |
| Our       | 1.0   | 1.2G   | 66.6    |

Figure 2. self-attention block (SA-Block)

Figure 3. Segmentation results mask of the proposed FAANet
5. Conclusion

In this paper, we propose multiple level aggregation and self-attention fusion strategy to tackle real-time semantic segmentation on high resolution image. Aggregation strategy connects different receptive field convolution layers to refine different level features from coarse to fine, which can vastly reduce calculation time. Quantitative experimental results on Cityscapes and our own head dataset are detailed to show the effectiveness of our strategy.

Acknowledgement

This paper was co-supported by the Foundation for Young Scholars in Wuhan Donghu University under grant 2018dhzk006. Thanks for the support of Wuhan Donghu University.

References

[1] Vijay Badrinarayanan, Alex Kendall, and Roberto Cipolla. Segnet: A deep convolutional encoder-decoder architecture for image segmentation. IEEE transactions on pattern analysis and machine intelligence, 39(12):2481–2495, 2017.

[2] Michael Treml, Jose Arjona-Medina, Thomas Unterthiner, Rupesh Durgesh, Felix Friedmann, Peter Schuberth, Andreas Mayr, Martin Heusel, Markus Hofmarcher, Michael Widrich, et al. Speeding up semantic segmentation for autonomous driving. In MLITS, NIPS Workshop, 2016.

[3] Zifeng Wu, Chunhua Shen, and Anton van den Hengel. Real-time semantic image segmentation via spatial sparsity. arXiv preprint arXiv:1712.00213, 2017.

[4] Changqian Yu, Jingbo Wang, Chao Peng, Changxin Gao, Gang Yu, and Nong Sang. Bisenet: Bilateral segmentation network for real-time semantic segmentation. arXiv preprint arXiv:1808.00897, 2018.

[5] Hang Zhang, Kristin Dana, Jianping Shi, Zhongyue Zhang, Xiaogang Wang, Ambrish Tyagi, and Amit Agrawal. Context encoding for semantic segmentation. arXiv preprint arXiv:1803.08904, 2018.

[6] Bolei Zhou, Hang Zhao, Xavier Puig, Sanja Fidler, Adela Barriuso, and Antonio Torralba. Semantic understanding of scenes through the ade20k dataset. arXiv preprint arXiv:1608.05442, 2016.

[7] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dolla’r, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In European conference on computer vision, pages 740–755. Springer, 2014.

[8] Hengshuang Zhao, Xiaojuan Qi, Xiaoyong Shen, Jianping Shi, and Jiaya Jia. Icnet for real-time semantic segmentation on high-resolution images. arXiv preprint arXiv:1704.08545, 2017.

[9] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin. Attention is all you need. In NIPS, pages 5998–6008, 2017.