# ESNET: EDGE-BASED SEGMENTATION NETWORK FOR REAL-TIME SEMANTIC SEGMENTATION IN TRAFFIC SCENES

*Haoran Lyu*[⋆] *Huiyuan Fu*[⋆] *Xiaojun Hu*[†] *Liang Liu*[⋆]

[⋆] Beijing University of Posts and Telecommunications
[†] DeepAIT Limited

## ABSTRACT

Semantic segmentation is widely used in the industry recently, especially in the field of scene understanding, surveillance and autonomous driving. However, majority of current state-of-the-art algorithms run accompany with high consumption of computation resources. Thus, our work focuses on real-time semantic segmentation which could reduce a large proportion of computation. Traditional methods to speed up segmentation process tend to down sample image. However, down sampling would cause the loss of information. Hence, we propose a real-time edge-based segmentation network (ESNet) that incorporate high-resolution global edge information with low-resolution classification-level semantic information. Our network performs real-time inference on single GPU card on high-resolution Cityscapes dataset.

***Index Terms—*** Real-Time, Semantic Segmentation, Global Edge Information, Classification Level Semantic Information

## 1. INTRODUCTION

Semantic segmentation, as one of the most fundamental tasks in computer vision field, has numerous applications like scene understanding, surveillance and autonomous driving nowadays. After the success of Fully Convolutional Network (FCN) [1], deep convolutional neural networks such as [2–5] make remarkable progress in the field of semantic segmentation. Recent works tend to add more layers or use bigger kernels to enhance the receptive field and context understanding ability of deep neural networks. These works provided better performance in terms of Mean Intersection over Union (mIoU) along with high computation cost and much more parameters.

With the emergence of industrial needs, a great number of computer vision techniques have been adapted to improve the productivity. However, in the field of image segmentation, people find it hard to make it eligible for practical use due to its high cost of computation. Our work dedicates on building a real-time semantic segmentation network architecture with decent accuracy that can be used in practical. Recent works in the field of semantic segmentation can be summarized as two categories:

**Quality-Oriented Semantic Segmentation:** FCN [1], as a pioneer work of semantic segmentation, replaced the last fully-connected layers with convolution layers. SegNet [3] and DeepLab V3+ [7] use the encoder-decoder architecture that can combine high-level semantic information with low-level spatial information. In former works [8–10], spatial relationship is modeled through conditional random fields (CRF) or Markov random fields (MRF). DeepLab [7, 10, 11] used ASPP module which took advantage of dilated convolution to increase the receptive field. PSPNet [12] proposed pyramid pooling (PPM) module that can combine context information from different regions of picture. Lin et al. [13] used multi-path refinement network which combined multiscale image features. These methods achieved effective performance while sacrificing the inference speed.

**Efficiency-Oriented Semantic Segmentation:** State-of-the-art efficient classification networks [14–17] are proposed to speed up the inference phase of the network. Depth-wise separable convolutions are frequently used for these light networks. They took a decent trade off on performance and efficiency. In object detection field, from Fast R-CNN [18] to Faster R-CNN [19], speed also became an important criterion for algorithm design. Afterwards, one-stage detection network like SSD [20] and YOLO [21–23] are widely used in the industry due to their high efficiency. However, the high speed inference in the field of semantic segmentation has not been mature yet. Limited works has been conducted: E-Net [24] and [25] explore lightweight network for semantic segmentation.

Our work is motivated by figure sketching. As first step of sketching, the painter needs to identify the objects and draw the edges of each object onto their canvas. From this perspective, edges contain the information that can distinguish between instances. Edge detection can provide rich low-level texture information and powerful distinguishing ability. In the context of semantic segmentation, edge information can make

Thanks to the National Key R&D Program of China 2017YFB1003000, the Natural Science Foundation of China under Grant no.61872047,61732017, and the NSFC-Guangdong Joint Fund under Grant no.U1501254 for funding.

978-1-5386-6249-6/19/$31.00 ©2019 IEEE

up the lost information caused by down sampling. Therefore, we incorporate the global edge information into segmentation network. Taking efficiency into consideration, we further explore an architecture that light edge detection network extract high-resolution global border information, and at the meantime, heavy segmentation network extract low-resolution classification-level semantic information.

The main contribution of our work can be summarized as following:

- We propose an ESNet framework that incorporates high-resolution of global edge information with low-resolution of classification level semantic information.
- We design an efficient edge detection network (Sec. 2.3) for edge branch.
- We propose a Multilayer Fusion (MLF) module to fuse the multilayer segmentation feature and global edge feature (Sec. 2.4).

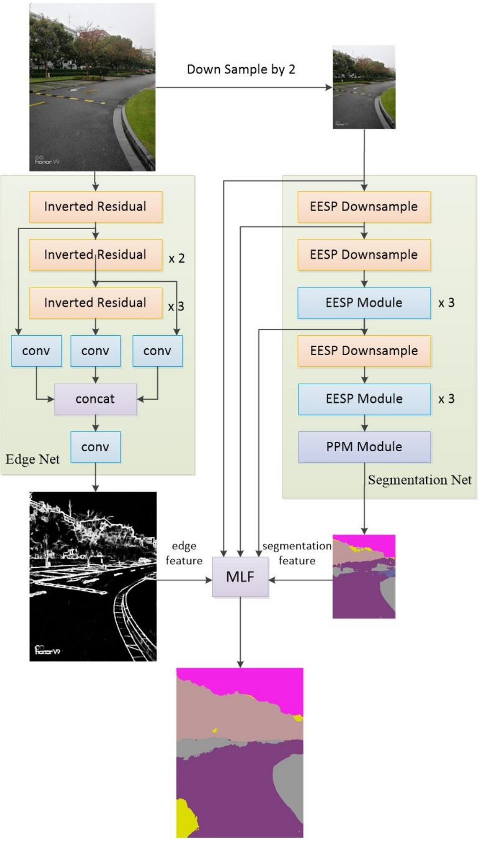

**Fig. 1**: Architecture of ESNet. MLF stand for Multilayer Fusion

## 2. ESNET

In this section, we begin with general pipeline of our method and then describe sub-component separately. We propose an Edge-based Segmentation Network (ESNet) that jointly trains and refines edge detection and segmentation together in a unified network. Efficient backbone networks: MobileNet V2 [16] and ESPNetv2 [26] are used in our work to make further improvement on efficiency for our network.

### 2.1. ESNet Architecture

The overall architecture of ESNet is illustrated in Fig.1. It is composed of two components: Edge net and Segmentation net. Edge net is responsible for extracting the global edge information of an image. Segmentation net is responsible for obtaining a pixel level classification map.

Traditional speed up strategy for segmentation is to down sample the input image and up sample the output result to fit original input. The consequence is that a large number of detail information are dropped during down sampling and it is hard to be recovered by simple up sampling. Thus, encoder-decoder architecture is carried out to recover lost information through deconvolution. However, it also brings the consumption of computation and parameters.

Our method takes advantage of both methods. The idea is to feed high-resolution input into a light weighted edge detection network to extract global inter-class information and feed low-resolution input into a heavy weighted network to obtain detailed pixel level classification information. Thus, we feed original high-resolution input into edge detection network (Sec. 2.2) and down sampled low-resolution input into segmentation network (Sec. 2.3). And incorporate output of these networks with MLF (Multilayer Fusion) module.

### 2.2. Edge Detection Network

In the task of semantic segmentation, the prediction is easily confused by the different categories with similar color or appearances, especially when they are adjacent spatially. Therefore, inter class distinguishable information is needed. With this motivation, we employ edge detection network to reinforce inter class distinguish ability of segmentation. It directly learns semantic boundary with explicit supervision from semantic segmentation annotation. Taking real-time inference into consideration, as the input of edge detection network is high-resolution image, we chose MobileNet V2 [16] as our backbone network. The detailed edge detection network is shown in Fig.2.

Auxiliary layer supervision is employed to obtain a coarse-to-fine edge detection result. Also, inspired by FPN [27], we combine different level supervision output to fuse high-level and low-level information together.

Annotator-robust Loss Function [6] is used to optimize our network:

$$l(X_i;W) = \begin{cases} \alpha \cdot log(1 - P(X_i;W)) & if\ y_i = 0 \\ 0 & if\ 0 < y_i \leq \eta \\ \beta \cdot log(P(X_i;W)) & otherwise \end{cases} \quad (1)$$

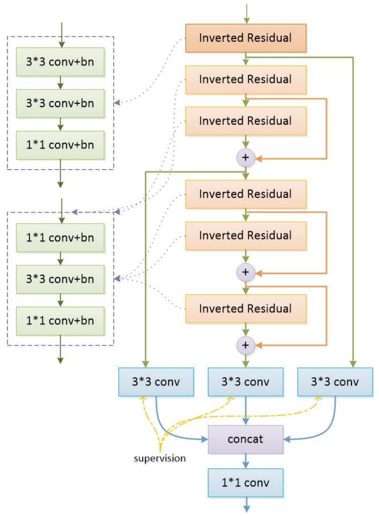

**Fig. 2**: Edge Detection Network for ESNet

In which:

$$\begin{aligned} \alpha &= \lambda \cdot \frac{|Y^+|}{|Y^+ + Y^-|} \\ \beta &= \frac{|Y^-|}{|Y^+ + Y^-|} \end{aligned} \tag{2}$$

$Y^+$ and $Y^-$ denote positive sample set and negative sample set respectively. The hyper-parameter $\lambda$ is to balance positive and negative samples. The activation value (CNN feature vector) and ground truth edge probability at pixel $I$ are presented by $X_i$ and $y_i$, respectively. $P(X)$ is the standard sigmoid function, and $W$ denotes all the parameters that will be learned. Overall loss function for edge detection can be formulated as:

$$L(W) = \sum_{i=1}^{|I|} (\sum_{k=1}^{K} l(X_i^{(k)}; W) + l(X_i^{fuse}; W)) \tag{3}$$

where $X^{(k)}$ is the activation value from stage $k$ and $X^{fuse}$ is from fusion layer. $|I|$ is the number of pixels in image $I$, and $K$ is the number of stages.

$l(X_i; W)$ in annotator-robust loss function is designed to eliminate the controversial annotation from different annotators. Thus, different hyper-parameter $\eta$ is set in pretraining phase and joint training phase which is discussed in Sec. 3.1.

### 2.3. Segmentation Network

To guarantee the efficiency, segmentation network receives the down sampled image as input (Fig.1 top). EESP [26] module output a state-of-art performance in terms of efficiency with decent segmentation quality. Thus, we apply EESP module as our backbone in segmentation network. Down sampled input goes through ESPNet V2 backbone and fed into PPM [14] module to aggregate different region of global information. Output of segmentation network, together with inter layer features, are feed into Multilayer fusion module.

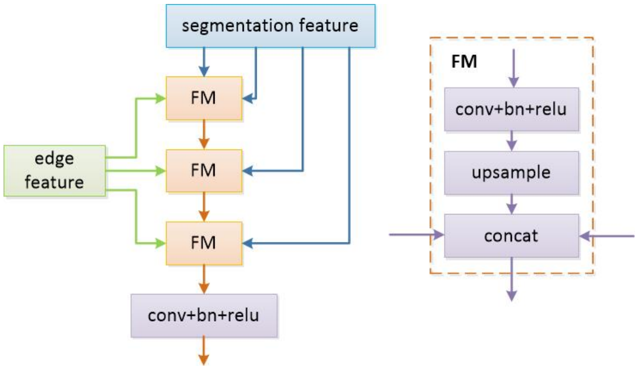

**Fig. 3**: Multilayer Fusion Module. FM(on the right of image) stands for Fusion Module

### 2.4. Multilayer Fusion

As we obtain global border information from edge detection network (Sec. 2.2) and semantic information from segmentation network (Sec. 2.3), a proper fusion of these features is needed. We propose Multilayer Fusion module to incorporate multilayer output feature from segmentation path and global border feature from edge detection path as shown in Fig.3. We extract segmentation feature from different intermediate layer and feed them into Fusion Module (FM). The output feature of edge detection network is down-sampled into different scales to be fused with segmentation feature through FM.

## 3. EXPERIMENTAL RESULTS

We evaluate our approach on widely used public dataset: Cityscapes [28]. It is a large semantic segmentation dataset of urban street scene. The resolution of images in Cityscapes is 2048 * 1024. This dataset contains 33 classes and 19 classes of them are considered for training and evaluation. It consists of 5,000 finely annotated images and 20,000 coarse annotated images. In our work, we use only finely annotated images for training and evaluation. There are 2,979 images for training, 500 images for validation and 1,525 images for testing.

### 3.1. Implementation Details

We conduct our experiments on four Nvidia Tesla P40 GPU card under CUDA 7.5 and CUDNN V5. Our implementation is under the pytorch framework. We pretrain our edge detection network with BSDS500 [29] dataset. In the pretraining phase of edge detection network we apply $\eta$ to eliminate the influence of controversy annotation from different annotators. Segmentation path is initialized with official pretrain model from ESPNet V2 [26]. Then, we jointly train the edge path, segmentation path and MLF with Cityscapes dataset. For Cityscapes, it only provides the segmentation information. Thus, we convert the segmentation annotation into boundary information through the boundary of adjacent different category objects so that we can train end-to-end. In training phase, $\eta$ mentioned in Sec. 2.2 is set to zero because segmentation annotation is explicit for edge information. We optimize our network through ADAM [30] optimizer with batch size 8, betas (0.9, 0.999) and weight decay 0.0005. We use the poly learning rate policy where the learning rate is multiplied by $(1 - \frac{iter}{max\_iter})^{power}$ with power 0.9 and initial learning rate $1e^{-3}$.

### 3.2. Speed analysis

As one of the most important factors in our work, we experiment the speed of our network under different configurations. We measure FLOPs for input size 224*224 on NVIDIA Titan Xp GPU. We compare the influence of s (scale factor) under different number of IR(Inverted Residual). Scale factor is defined in ESPNet V2 [26]. It stands for the kernel number ratio against base setting which influences the number of kernels defined in EESP [26] module. Number of IR is the number of Inverted Residual layers defined in Sec. 2.2. As illustrated in Table 1, scale factor has limited effect on FLOPs in ESNet. As high-resolution and low-resolution image goes through edge branch and segmentation branch simultaneously, it would provide better efficiency if both branch shares similar FLOPs. Thus, we evaluate the computation consumption of edge branch of our ESNet.

**Table 1**: FLOPs of ESNet under different network configurations with input resolution 224*224, s: scale factor, IR: number of Inverted Residual

| Network | s = 0.5 | s = 1 | s = 1.5 |
|---|---|---|---|
| ESNet(IR=6) | 134.14M | 158.25M | 193.39M |
| ESNet(IR=3) | 95.18M | 119.29M | 154.43M |
| ESNet(IR=1) | 36.95M | 61.06M | 96.20M |

**Table 2**: FLOPs of Edge branch under different number of IRs with input resolution 224*224

| | IR = 1 | IR = 3 | IR = 6 |
|---|---|---|---|
| Edge | 23.80M | 82.03M | 120.99M |

As shown in Table 2, IR module occupies most of the FLOPs as it accepts larger input than segmentation branch. Thus, to achieve better efficiency, we adopt number of IR=1 so that we can make a balance between segmentation branch and edge detection branch.

**Table 3**: Speed and FLOP comparison with state-of-the-art networks

| Network | FLOPs | Speed (FPS) |
|---|---|---|
| PSPNet [12] | 82.78 G | 5 |
| FCN-8s [1] | 62.71 G | 15 |
| DeepLab-v2 [10] | 37.51 G | 6 |
| SegNet [3] | 31 G | 17 |
| ESNet(Ours) | 193 M<br>154 M<br>96 M | 34<br>37<br>56 |

We also compare our ESNet speed (FPS) and FLOPs with current state-of-the-art networks. ESNet achieve real-time inference with less FLOPs with large input size 1024*512.

### 3.3. Performance analysis

**Table 4**: Speed and performance comparison with state-of-the-art traffic scene segmentation networks under input resolution 1024*512

| Network | Time(ms) | Frame(FPS) | mIoU(%) |
|---|---|---|---|
| Multi-boost [32] | 250 | 4 | 59.2 |
| SCNN [31] | 161 | 6 | 68.2 |
| ESNet(Ours) | 18 | 34 | 63.2 |
| | 29 | 53 | 60.1 |

SCNN [31] provides powerful SCNN basic module that learns spatial relationship for structure output. A single SCNN module would consume 42ms under GeForce GTX TITAN Black card. We test SCNN with resnet backbone on single Titan Xp GPU card. Compare to SCNN, ESNet achieve 767% and 566% speed up with 8.1% and 5% mIoU trade off. Our ESNet achieves real-time inference and provides competitive performance in terms of mIoU.

## 4. CONCLUSION

We propose an edge-based real-time segmentation network (ESNet). It incorporates high-resolution of global edge information with low-resolution of classification level semantic information. The major contributions are: ESNet framework, efficient edge extraction network and Multilayer Fusion (MLF) module. We conduct experiments on Cityscapes dataset and achieve real-time inference with decent performance.

## 5. REFERENCES

[1] J. Long, E. Shelhamer, and T. Darrell. Fully convolutional networks for semantic segmentation. In IEEE Conference on Computer Vision and Pattern Recognition, 2015.

[2] Chen, L., Papandreou, G., Kokkinos, I., Murphy, K., Yuille, A.L.: Semantic image segmentation with deep convolutional nets and fully connected crfs. ICLR. 2015.

[3] V. Badrinarayanan, A. Kendall, and R. Cipolla. Segnet: A deep convolutional encoder-decoder architecture for image segmentation. IEEE Transactions on Pattern Analysis and Machine Intelligence, 39(12):2481C2495, 2017.

[4] Zhao, Hengshuang, et al. Pyramid scene parsing network. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR). 2017.

[5] L.-C. Chen, G. Papandreou, F. Schroff, and H. Adam. Rethinking atrous convolution for semantic image segmentation. arXiv, 2017.

[6] Liu, Yun, et al. Richer convolutional features for edge detection. Computer Vision and Pattern Recognition (CVPR), 2017 IEEE Conference on. IEEE, 2017.

[7] Chen, Liang-Chieh, et al. Encoder-decoder with atrous separable convolution for semantic image segmentation. arXiv preprint arXiv:1802.02611. 2018.

[8] Liu, Ziwei, et al. Semantic image segmentation via deep parsing network. Proceedings of the IEEE International Conference on Computer Vision. 2015.

[9] Zheng, Shuai, et al. ”Conditional random fields as recurrent neural networks.” Proceedings of the IEEE international conference on computer vision. 2015.

[10] Chen, Liang-Chieh, et al. Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. IEEE transactions on pattern analysis and machine intelligence 40.4 (2018): 834-848.

[11] Chen, Liang-Chieh, et al. Rethinking atrous convolution for semantic image segmentation. arXiv preprint arXiv:1706.05587. 2017.

[12] Zhao, Hengshuang, et al. Pyramid scene parsing network. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR). 2017.

[13] Lin, G., et al. Refinenet: Multi-path refinement networks for high-resolution semantic segmentation. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR). 2017.

[14] A. G. Howard, M. Zhu, B. Chen, D. Kalenichenko, W. Wang, T. Weyand, M. Andreetto, and H. Adam. Mobilenets: Efficient convolutional neural networks for mobile vision applications. arXiv preprint arXiv:1704.04861, 2017.

[15] Ma, Ningning, et al. Shufflenet v2: Practical guidelines for efficient cnn architecture design. Proceedings of the European Conference on Computer Vision (ECCV). 2018.

[16] M. Sandler, A. Howard, et al. Mobilenetv2: Inverted residuals and linear bottlenecks. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR). 2018.

[17] Chollet, Francois. Xception: Deep Learning with Depthwise Separable Convolutions. 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). IEEE, 2017.

[18] Girshick, R.: Fast R-CNN. In: ICCV. 2015.

[19] Ren, Shaoqing, et al. ”Faster r-cnn: Towards real-time object detection with region proposal networks.” Advances in neural information processing systems. 2015.

[20] Liu, Wei, et al. Ssd: Single shot multibox detector. European conference on computer vision. Springer, Cham, 2016.

[21] Redmon, J., Divvala, S.K., Girshick, R.B., Farhadi, A.: You only look once: Unified, real-time object detection. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR). 2016.

[22] Redmon, J., Farhadi, A.: YOLO9000: better, faster, stronger. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR). 2017.

[23] Redmon, Joseph, and Ali Farhadi. Yolov3: An incremental improvement. arXiv preprint arXiv:1804.02767 (2018).

[24] Paszke, Adam, et al. Enet: A deep neural network architecture for real-time semantic segmentation. arXiv preprint arXiv:1606.02147 (2016).

[25] Romera, E., Alvarez, J.M., Bergasa, L.M., Arroyo, R.: Efficient convnet for realtime semantic segmentation. Intelligent Vehicles Symposium (IV). 2017.

[26] Mehta, Sachin, et al. ESPNetv2: A Light-weight, Power Efficient, and General Purpose Convolutional Neural Network. arXiv preprint arXiv:1811.11431. 2018.

[27] Lin, Tsung-Yi, et al. Feature pyramid networks for object detection. CVPR. Vol. 1. No. 2. 2017.

[28] M. Cordts, M. Omran, S. Ramos, T. Rehfeld, M. Enzweiler, R. Benenson, U. Franke, S. Roth, and B. Schiele. The cityscapes dataset for semantic urban scene understanding. In IEEE Conference on Computer Vision and Pattern Recognition, 2016.

[29] P. Arbelaez, M. Maire, C. Fowlkes, and J. Malik. Contour detection and hierarchical image segmentation. IEEE TPAMI, 33(5):898C916, 2011.

[30] Kinga, D., and J. Ba Adam. A method for stochastic optimization. International Conference on Learning Representations (ICLR). Vol. 5. 2015.

[31] Pan, Xingang, et al. Spatial as deep: Spatial cnn for traffic scene understanding. Thirty-Second AAAI Conference on Artificial Intelligence. 2018.

[32] Costea, Arthur D., and Sergiu Nedevschi. Traffic scene segmentation based on boosting over multimodal low, intermediate and high order multi-range channel features. Intelligent Vehicles Symposium (IV), 2017 IEEE.