SCIENCE CHINA
Information Sciences

• LETTER •

March 2022, Vol. 65 139105:1–139105:2
https://doi.org/10.1007/s11432-020-2971-8

VNet: a versatile network to train real-time semantic segmentation models on a single GPU

Wenxing LI[1,2], Ning LIN[2,3], Mingzhe ZHANG[2], Hang LU[2,3*], Xiaoming CHEN[2,3*] & Xiaowei LI[2,3*]

[1]*College of Computer Science and Technology, Guizhou University, Guiyang* 550025*, China;*
[2]*State Key Laboratory of Computer Architecture, Institute of Computing Technology, Chinese Academy of Sciences, Beijing* 100190*, China;*
[3]*University of Chinese Academy of Sciences, Beijing* 100049*, China*

Received 23 March 2020/Revised 11 June 2020/Accepted 28 June 2020/Published online 5 August 2021

Citation Li W X, Lin N, Zhang M Z, et al. VNet: a versatile network to train real-time semantic segmentation models on a single GPU. Sci China Inf Sci, 2022, 65(3): 139105, https://doi.org/10.1007/s11432-020-2971-8

Dear editor,

Modern semantic segmentation, which has important applications such as medical image analysis, image editing, and video surveillance, has made remarkable progress using deep convolution neural network models. Recently, an efficient real-time semantic segmentation method has received considerable attention, as intelligent edge devices not only have faster inference speed requirements for semantic segmentation models but also cannot rely on the cloud services of data centers. There are two feasible approaches to develop an efficient semantic segmentation model. The first approach is by designing efficient models: designing and developing the models' architecture from scratch (e.g., ENet [1]). The second approach, which is less common but increasingly popular, is network compression: to develop light-weight models (e.g., ICNet [2]) with pruning methods [3] that are widely used in image classification tasks. However, both these approaches are difficult to follow to develop light-weight and fast semantic segmentation models without compromising on the accuracy of the models.

In this study, we investigate a salient question: whether other important factors can achieve better accuracy and why previous studies disregard these factors. In previous studies, the resolution of high-resolution images is usually reduced before training semantic segmentation models on graphic processing units (GPUs). For instance, the 2048 × 1024 resolution Cityscapes dataset [4] is usually randomly cropped or resized to half (1024 × 512) or quarter (512 × 256) of the original resolution. However, our empirical study shows that cropped images lose content information, and resized images destroy image details. Image content information and details are useful for ensuring high accuracy of small semantic segmentation models. Also, owing to the limitations of GPU memory, previous studies are unable to use the original resolution of high-resolution images to train semantic segmentation models. For example, to successfully train the ENet, which has only 0.36 million parameters, on one GPU (12 GB), input images have to be reduced to 1024 × 512 resolution. Although it is possible to train semantic segmentation models with the original resolution of high-resolution images on multiple regular GPUs (one GPU card per image), some larger models, which have higher accuracy and more parameters than ENet, cannot be trained with a single high-resolution image on one GPU card. Therefore, larger GPU memory devices such as NVIDIA Tesla V100 (32 GB) have to be adopted, although they are expensive, which limits researchers from exploring other approaches to develop efficient semantic segmentation models.

This study aims to determine an effective approach to train semantic segmentation models with high-resolution images on one GPU (12 GB) and consequently, develop light-weight, fast-speed, and accurate semantic segmentation models. Hence, we propose a novel versatile network (VNet), which is mainly based on versatile module (V module) and contextual pyramid pooling (CPP) module, as shown in Figure 1. The V module comprises two parts: a reversible function and asymmetric convolution layers. For efficient memory management, the reversible function [5] is adopted so that input feature maps can be computed from the function's output. When we stack more reversible modules, only the final feature map needs to be cached. Thus, in back-propagation, the intermediate feature map can be calculated by the inversion property of reversible modules. The specific calculation formulation for the forward-propagation can be formulated as

$$\begin{aligned} X &= [X_1, X_2], \\ Z &= X_1 + F_1(X_2), \\ Y_1 &= Z, \\ Y_2 &= X_2 + Z, \\ Y &= [Y_1, Y_2], \end{aligned} \tag{1}$$

* Corresponding author (email: luhang@ict.ac.cn, chenxiaoming@ict.ac.cn, lxw@ict.ac.cn)

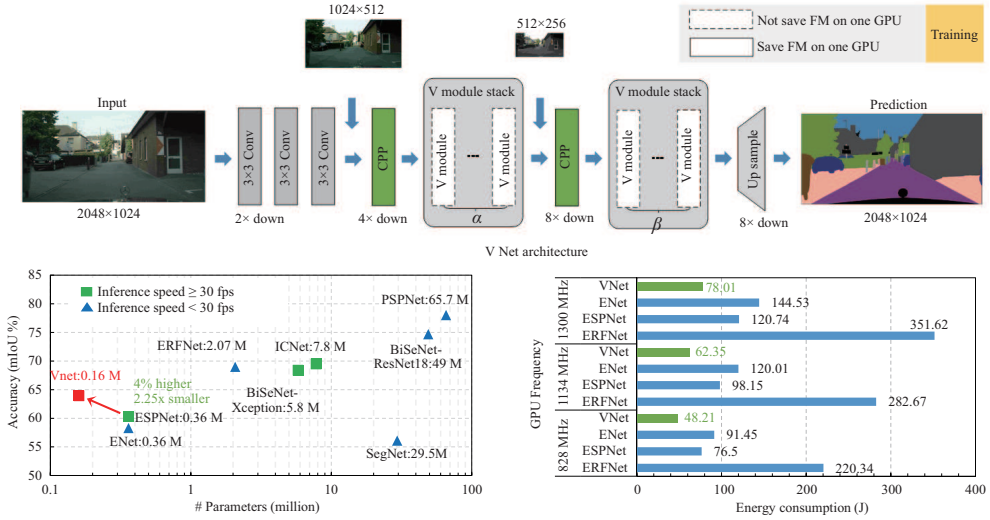

Figure 1 (Color online) VNet architecture and the corresponding experiments on an NVIDIA TITAN Xp and an NVIDIA Jetson Tx2.

where X is the input feature map, which is concatenated on the channel dimension of feature map X_1 and X_2, and Y is the output feature map. Through this expression, the input feature map X does not need to be stored, only the output Y requires to be cached. The corresponding backward of reversible function is

$$\begin{aligned} Y &= [Y_1, Y_2], \\ Z &= Y_1, \\ X_2 &= Y_2 - Z, \\ X_1 &= Z - F_1(X_2), \\ X &= [X_1, X_2]. \end{aligned} \tag{2}$$

Furthermore, to reduce the number of training parameters, the asymmetric convolution [6] has been introduced so that an $n \times n$ convolution kernel can be factorized to an $n \times 1$ convolution followed by a $1 \times n$ convolution kernel, and this factorization works well on medium layers. Herein, we replace the 3×3 convolution kernel with a 3×1 convolution followed by a 1×3 convolution kernel. Thus, we can reduce the number of parameters and the computational cost of VNet by 33% when the number of input filters and number of output filters are identical.

Contextual information is essential in semantic segmentation. The CPP module is inspired by the pyramid pooling module (PPM) [7], which acquires features from four different global pooling branches. Compared with PSPNet, the CPP module is designed to reduce the resolution of feature maps, which concatenates the parallel outputs of point-wise convolutions with stride – 2 and the sum of PPMs. The point-wise convolution is a 1×1 convolution kernel designed to utilize the channel level contextual information, which decreases the number of parameters and reduces the computation cost of VNet. We use the CPP to aggregate more spatial level contextual information to improve the accuracy.

Experiment. VNet can be trained using multiple high-resolution images on one GPU (12 GB) and can be effectively deployed on resource-constrained edge devices. In particular, VNet can process 2048×1024 high-resolution images at a rate of 55.5 frames per second (fps) and 15.5 fps on an NVIDIA TITAN Xp and an NVIDIA Jetson Tx2 with only 0.16 million parameters. The accuracy of VNet is 4% higher than ESPNet, and this is largely due to the training of VNet with high-resolution images. Also, the number of parameters of VNet is 2.25× smaller than that of ESPNet (as illustrated in Figure 1). Moreover, we obtain the energy consumption of VNet on the Jetson Tx2. Owing to the fast inference speed and lower-power consumption of VNet, it has the lowest energy consumption compared to other models. For instance, when the GPU frequency is set to 1300 MHz, the GPU energy consumption of VNet is 78.01 J, which is 1.85×, 1.54×, and 4.5× smaller than of ENet, ESPNet, and ERFNet [8], respectively. Consequently, our method is more efficient for battery-powered edge devices.

Acknowledgements This work was supported by National Key R&D Program of China (Grant No. 2018YFA0701500), Strategic Priority Research Program of CAS (Grant No. XDB44000000), Beijing Academy of Artificial Intelligence (BAAI), National Natural Science Foundation of China (Grant No. 61532017), and CARCH Innovation Project (Grant No. CARCH4506).

References

1 Paszke A, Chaurasia A, Kim S, et al. Enet: a deep neural network architecture for real-time semantic segmentation. 2016. ArXiv:1606.02147
2 Zhao H S, Qi X J, Shen X Y, et al. ICNet for real-time semantic segmentation on high-resolution images. In: Proceedings of European Conference on Computer Vision, 2018. 405–420
3 Li H, Kadav A, Durdanovic I, et al. Pruning filters for efficient ConvNets. 2016. ArXiv:1608.08710
4 Cordts M, Omran M, Ramos S, et al. The cityscapes dataset for semantic urban scene understanding. In: Proceedings of Computer Vision and Pattern Recognition, 2016. 3213–3223
5 Gomez A N, Ren M, Urtasun R, et al. The reversible residual network: backpropagation without storing activations. In: Proceedings of Advances in Neural Information Processing Systems, 2017. 2214–2224
6 Chen L C, Papandreou G, Schroff F, et al. Rethinking atrous convolution for semantic image segmentation. 2017. ArXiv:1706.05587
7 Zhao H S, Shi J P, Qi X J, et al. Pyramid scene parsing network. In: Proceedings of Computer Vision and Pattern Recognition, 2017. 2881–2890
8 Romera E, Alvarez J M, Bergasa L M, et al. ERFNet: efficient residual factorized ConvNet for real-time semantic segmentation. IEEE Trans Intell Transp Syst, 2018, 19: 263–272