Improving binary semantic scene segmentation for robotics applications

Maria Tzelepi, Nikolaos Tragkas, and Anastasios Tefas

Aristotle University of Thessaloniki
{mtzelepi,nktragkas,tefas}@csd.auth.gr

Abstract. Robotics applications are accompanied by particular computational restrictions, i.e., operation at sufficient speed, on embedded low power GPUs, and also for high-resolution input. Semantic scene segmentation performs an important role in a broad spectrum of robotics applications, e.g., autonomous driving. In this paper, we focus on binary segmentation problems, considering the specific requirements of the robotics applications. To this aim, we utilize the BiseNet model, which achieves significant performance considering the speed-segmentation accuracy trade-off. The target of this work is two-fold. Firstly, we propose a lightweight version of BiseNet model, providing significant speed improvements. Secondly, we explore different losses for enhancing the segmentation accuracy of the proposed lightweight version of BiseNet on binary segmentation problems. The experiments conducted on various high and low power GPUs, utilizing two binary segmentation datasets validated the effectiveness of the proposed method.

Keywords: Semantic segmentation · Binary · Bisenet · Robotics · Low power GPUs.

1 Introduction

Semantic scene segmentation refers to the task of assigning a class label to each pixel of an image, and hence it is also known as pixel-level classification. Semantic scene segmentation is a challenging task involved in numerous robotics applications, such as autonomous driving [1,11,20]. Robotics applications are accompanied by particular computational requirements. That is, the utilized models should be able to effectively operate at sufficient speed, on embedded low power GPUs, while also considering high-resolution input.

Recent advances in Deep Learning (DL), besides other problems [15,16,14], have provided effective models for addressing the general problem of semantic scene segmentation [7]. The seminal approach introduced fully convolutional neural networks [10]. Subsequently, considerable research has been conducted, focusing on improving the segmentation accuracy [2,9], however, without considering the issue of deployment (inference) speed. That is, most of the existing state-of-the-art DL segmentation models are computationally heavy, and hence ill-suited for robotics applications.

Thus, in the recent literature there have been works that also focus on the deployment speed, providing real-time segmentation models, considering mainly high power GPUs [4,12,3,18,6,19]. A comparative study of current semantic segmentation models considering the inherent computational restrictions in the context of robotics applications is provided in [17]. More specifically, extensive experiments have been conducted on different embedded platforms (e.g., AGX Xavier, NVIDIA TX-2), and also for various input resolutions, ranging from lower to higher ones. From the conducted experiments, it is evident that the Bilateral Segmentation Network (BiseNet) [19] model achieves considerable performance considering the segmentation accuracy-speed trade-off. Towards this end, in this work we employ BiseNet model and we address the problem of binary semantic segmentation considering robotics applications.

The target of this work is to explore ways of improving the performance of BiseNet model both in terms of deployment speed and segmentation accuracy, considering binary segmentation problems. To this aim, we first propose a lightweight version of the model. That is, we propose a lightweight network instead of ResNet-18 [8] that is used in the so-called *context path*. Subsequently, we exploit the available losses. Specifically, apart from the widely used cross entropy (softmax) loss, which is also used in the initial version of BiseNet, we apply hinge loss, since as it is shown in the recent literature, it provides improved accuracy considering binary classification problems [15].

The remainder of the manuscript is structured as follows. Section 2 provides the description of the proposed ways of improving binary segmentation, that is the proposed lightweight version of BiseNet and the investigation on loss functions. Next, Section 3 provides the experimental evaluation, and finally conclusions are drawn in Section 4.

2 Proposed Method

The BiseNet model consists of two paths, that is *Spatial Path* and *Context Path*. The spatial path is used in order to preserve the spatial information and generate high resolution features, and the context path with a fast downsampling strategy is used in order to to obtain sufficient receptive field. Furthermore, the model includes two modules, that is a Feature Fusion Module and an Attention Refinement Module, in order to further improve the accuracy with acceptable cost. Finally, apart from the principal cross entropy loss which supervises the output of the BiseNet model, two auxiliary losses are utilized to supervise the output of the context path.

In this work, we propose to replace the ResNet-18 model used in the context path, with a more lightweight model. The proposed model, which is based on the VGG model [13], consists of five pairs of convolutional layers followed by batch normalization and Rectified Linear Unit (ReLU) activation. A max-pooling layer follows each convolutional block. The proposed model architecture is illustrated in Fig. 1. Furthermore, we provide Table 1 which summarizes the design of the proposed model. As it will be presented the modified BiseNet model utilizing

the proposed lightweight model in the context path achieves considerable speed improvements.

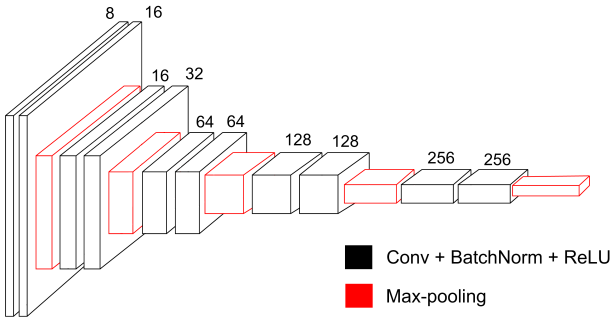

Fig. 1: Proposed lightweight model in the context-path: Red boxes represent the max-pooling layers, while the black boxes represent the convolutional layers, followed by batch normalization ReLU activation. The numbers of output channels of each convolutional layer are also depicted.

Layer	conv3-8 conv3-16	maxpool	conv3-16 conv3-32	maxpool	conv3-64 conv3-64	maxpool	conv3-128 conv3-128	maxpool	conv3-256 conv3-256	maxpool

Table 1: Layers of the proposed fully-convolutional lightweight context-path. *conv3-x* abbreviates a convolutional layer with kernel 3×3 and x output channels. Batch normalization is applied to each convolutional layer, while ReLU is used as activation function. The maxpool layers downsample by a factor of 2 feature maps.

Subsequently, since lightweight models usually have inferior performance as compared to their heavyweight counterparts, we explore ways to improve their performance. To achieve this goal, we focus on the loss functions for training the segmentation model. More specifically, even though cross entropy loss is a widely used loss function in DL, in a recent work [15] it has been demonstrated that, considering binary classification problems, hinge loss can achieve improved classification performance. Motivated by the aforementioned observation, in this work, we extend this investigation on binary segmentation problems. Thus, we utilize hinge loss so as to supervise the output of the whole model. Hinge loss per pixel is defined as:

$$\ell_h = \sum_{j=1}^{N_c} max(0, 1 - \zeta\{c = j\} y_j^{last}) \tag{1}$$

where $c \in [1, \cdots, N_c]$ indicates the correct class among the N_c classes, y_j^{last} indicates the score with respect to the j-th class, and

$$\zeta\{condition\} = \begin{cases} 1 & \text{, if condition} \\ -1 & \text{, otherwise} \end{cases}$$

In our case, $N_c = 2$, since we deal with binary segmentation problems.

3 Experiments

In this work, we first evaluate the deployment speed of the proposed modified BiseNet model, since a principal target of this work is to provide a faster model for binary semantic segmentation. We evaluate the deployment speed in term of Frames Per Second (FPS), on various high power and low power GPUs, as well as for various input sizes. Subsequently, we evaluate the performance of the modified model utilizing hinge loss against cross entropy loss as principal supervised loss, utilizing mean Intersection Over Union (mIOU) as evaluation metric.

3.1 Datasets

In this work, we utilize the CityScapes [5] dataset, exploiting only the Human and Vehicle classes, in order to build the two binary segmentation datasets, i.e., *Human Vs Non-Human* and *Vehicle Vs Non-Vehicle*, respectively. The first one consists of 11,900 train images and 2,000 test images, while the second one consists of 2,975 train images and 500 test images.

3.2 Implementation Details

All the experiments conducted using the Pytorch framework. Mini-batch gradient descent is used for the networks training, where an update is performed for every mini-batch of 8 samples. Momentum is set to 0.9, while the learning rate policy of the initial work was followed. All the models are trained on an NVIDIA 2080 Ti, and the deployment speed was tested on various low power GPUs.

3.3 Experimental Results

In the first set of experiments we evaluate the deployment speed of the proposed modified lightweight BiseNet model against the initial version which uses the ResNet-18 model. We have conducted experiments on a high power NVIDIA 2080 Ti, a high power NVIDIA 2070, a low power NVIDIA Jetson TX-2, and a low power NVIDIA AGX Xavier. Furthermore, we use various input dimensions ranging from 400×400 to 1024×1024. The experimental results are illustrated in Tables 2-5. Best results are printed in bold. As it is shown, the proposed model runs significantly faster as compared to the initial model, in any considered case. It can also be observed increased discrepancy for lower input sizes.

Moreover, it should be emphasized that the proposed lightweight version of BiseNet accomplishes faster inference speed compared to the original BiseNet model (using ResNet-18), without considerably sacrificing the segmentation accuracy. For example, on the Vehicle dataset, where the proposed lightweight version of BiseNet achieves mIOU 94.82% using the cross entropy loss, the original BiseNet achieves mIOU 95.86%. That is, we sacrifice the segmentation accuracy by roughly 1%, gaining significant speed ups (e.g., on NVIDIA AGX Xavier for input 600 $\times$ 600, BiseNet runs at 26.78 FPS, while the modified lightweight BiseNet runs at 40.12 FPS).

Table 2: Evaluation of speed in terms of FPS utilizing the proposed lightweight model in the context path, against the ResNet-18 on an NVIDIA 2080 Ti.

Input Size	**BiseNet - ResNet18**	**BiseNet - Proposed**
1024 $\times$ 1024	66.51	**75.23**
1280 $\times$ 720	70.69	**84.06**
800 $\times$ 800	98	**119.76**
600 $\times$ 600	163.91	**215.61**
640 $\times$ 360	216.37	**305.89**
400 $\times$ 400	269.58	**353.59**

Table 3: Evaluation of speed in terms of FPS utilizing the proposed lightweight model in the context path, against the ResNet-18 on an NVIDIA 2070.

Input Size	**BiseNet - ResNet18**	**BiseNet - Proposed**
1024 $\times$ 1024	50.40	**59.15**
1280 $\times$ 720	56.17	**66.31**
800 $\times$ 800	77.46	**93.85**
600 $\times$ 600	125.04	**166.26**
640 $\times$ 360	184.59	**251.26**
400 $\times$ 400	237.61	**315.24**

Subsequently, the experimental results for evaluating the hinge loss against cross entropy loss, utilizing the proposed lightweight BiseNet model, on the two binary segmentation datasets are presented in Table 6. Best results are printed in bold. As it is demonstrated, hinge loss accomplishes superior performance as compared to the cross entropy loss, considering binary segmentation problems.

Furthermore, we note that better segmentation performance is achieved using hinge loss on the original BiseNet model (using ResNet-18), too. For example, on the Vehicle dataset the original BiseNet achieves mIOU 95.86% with cross entropy loss, while hinge loss achieves mIOU 96.40%.

Table 4: Evaluation of speed in terms of FPS utilizing the proposed lightweight model in the context path, against the ResNet-18 on an NVIDIA Jetson TX2.

Input Size	**BiseNet - ResNet18**	**BiseNet - Proposed**
1024×1024	3.98	**5.02**
1280×720	4.32	**5.58**
800×800	5.71	**7.69**
600×600	9.7	**13.36**
640×360	14.71	**21.05**
400×400	18.82	**26.42**

Table 5: Evaluation of speed in terms of FPS utilizing the proposed lightweight model in the context path, against the ResNet-18 on an NVIDIA AGX Xavier.

Input Size	**BiseNet - ResNet18**	**BiseNet - Proposed**
1024×1024	11.58	**15.32**
1280×720	12.23	**16.96**
800×800	16.38	**23.40**
600×600	26.78	**40.12**
640×360	40.13	**60.66**
400×400	52.42	**77.83**

Table 6: Evaluation on segmentation performance in terms of mIOU (%) utilizing the modified lightweight BiseNet model, for evaluating hinge loss against cross entropy loss.

Dataset	**Cross Entropy Loss**	**Hinge Loss**
Human Vs Non-Human	87.82	**89.23**
Vehicle Vs Non-Vehicle	94.82	**95.03**

Finally, some qualitative results are presented utilizing the proposed modified model trained for human segmentation and vehicle segmentation in Fig. 2.

4 Conclusions

In this paper, we dealt with binary segmentation problems, utilizing the BiseNet model, which achieves significant performance considering the speed-segmentation accuracy trade-off. First, we proposed a lightweight version of BiseNet model, in order to improve the deployment speed. Subsequently, we explored different losses in order to enhance the segmentation accuracy of the proposed lightweight version of BiseNet on binary segmentation problems. The experiments conducted on various high and low power GPUs, utilizing two binary segmentation datasets, validated the effectiveness of proposed lightweight version of BiseNet in terms of deployments speed, as well as that hinge loss provides improved performance considering binary segmentation problems.

Acknowledgment

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 871449 (OpenDR). This publication reflects the authors' views only. The European Commission is not responsible for any use that may be made of the information it contains.

References

1. Alonso, I., Riazuelo, L., Murillo, A.C.: Mininet: An efficient semantic segmentation convnet for real-time robotic applications. IEEE Transactions on Robotics (2020)
2. Badrinarayanan, V., Kendall, A., Cipolla, R.: Segnet: A deep convolutional encoder-decoder architecture for image segmentation. IEEE transactions on pattern analysis and machine intelligence **39**(12), 2481–2495 (2017)
3. Chao, P., Kao, C.Y., Ruan, Y.S., Huang, C.H., Lin, Y.L.: Hardnet: A low memory traffic network. In: Proceedings of the IEEE International Conference on Computer Vision. pp. 3552–3561 (2019)
4. Chen, W., Gong, X., Liu, X., Zhang, Q., Li, Y., Wang, Z.: Fasterseg: Searching for faster real-time semantic segmentation. arXiv preprint arXiv:1912.10917 (2019)
5. Cordts, M., Omran, M., Ramos, S., Rehfeld, T., Enzweiler, M., Benenson, R., Franke, U., Roth, S., Schiele, B.: The cityscapes dataset for semantic urban scene understanding. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 3213–3223 (2016)
6. Emara, T., Abd El Munim, H.E., Abbas, H.M.: Liteseg: A novel lightweight convnet for semantic segmentation. In: 2019 Digital Image Computing: Techniques and Applications (DICTA). pp. 1–7. IEEE (2019)
7. Garcia-Garcia, A., Orts-Escolano, S., Oprea, S., Villena-Martinez, V., Garcia-Rodriguez, J.: A review on deep learning techniques applied to semantic segmentation. arXiv preprint arXiv:1704.06857 (2017)

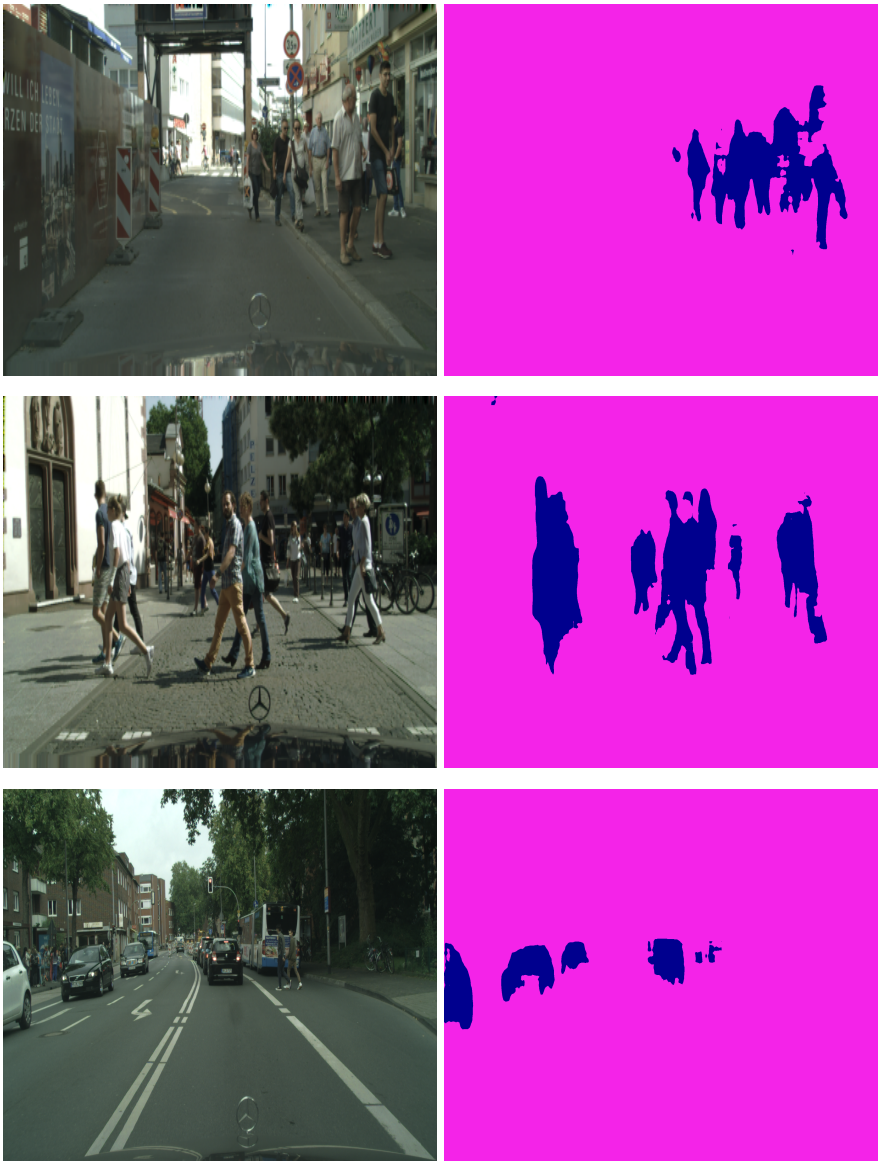

Fig. 2: Predictions of the modified lightweight BiSeNet model trained on Human Vs Non-Human and Vehicle Vs Non-Vehicle datasets.

8. 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. pp. 770–778 (2016)
9. Lateef, F., Ruichek, Y.: Survey on semantic segmentation using deep learning techniques. Neurocomputing **338**, 321–348 (2019)
10. Long, J., Shelhamer, E., Darrell, T.: Fully convolutional networks for semantic segmentation. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 3431–3440 (2015)
11. Milioto, A., Lottes, P., Stachniss, C.: Real-time semantic segmentation of crop and weed for precision agriculture robots leveraging background knowledge in cnns. In: 2018 IEEE international conference on robotics and automation (ICRA). pp. 2229–2235. IEEE (2018)
12. Poudel, R.P., Liwicki, S., Cipolla, R.: Fast-scnn: Fast semantic segmentation network. arXiv preprint arXiv:1902.04502 (2019)
13. Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556 (2014)
14. Tzelepi, M., Passalis, N., Tefas, A.: Probabilistic online self-distillation. Neurocomputing (2022)
15. Tzelepi, M., Tefas, A.: Improving the performance of lightweight cnns for binary classification using quadratic mutual information regularization. Pattern Recognition p. 107407 (2020)
16. Tzelepi, M., Tefas, A.: Efficient training of lightweight neural networks using online self-acquired knowledge distillation. In: 2021 IEEE International Conference on Multimedia and Expo (ICME). pp. 1–6. IEEE (2021)
17. Tzelepi, M., Tefas, A.: Semantic scene segmentation for robotics applications (2021)
18. Yu, C., Gao, C., Wang, J., Yu, G., Shen, C., Sang, N.: Bisenet v2: Bilateral network with guided aggregation for real-time semantic segmentation. arXiv preprint arXiv:2004.02147 (2020)
19. Yu, C., Wang, J., Peng, C., Gao, C., Yu, G., Sang, N.: Bisenet: Bilateral segmentation network for real-time semantic segmentation. In: Proceedings of the European conference on computer vision (ECCV). pp. 325–341 (2018)
20. Zhang, Y., Chen, H., He, Y., Ye, M., Cai, X., Zhang, D.: Road segmentation for all-day outdoor robot navigation. Neurocomputing **314**, 316–325 (2018)