2018 21st International Conference on Intelligent Transportation Systems (ITSC)
Maui, Hawaii, USA, November 4-7, 2018

Fusion Scheme for Semantic and Instance-level Segmentation

Arthur Daniel Costea*, Andra Petrovai*, Sergiu Nedevschi

Abstract— **A powerful scene understanding can be achieved by combining the tasks of semantic segmentation and instance level recognition. Considering that these tasks are complementary, we propose a multi-objective fusion scheme which leverages the capabilities of each task: pixel level semantic segmentation performs well in background classification and delimiting foreground objects from background, while instance level segmentation excels in recognizing and classifying objects as a whole. We use a fully convolutional residual network together with a feature pyramid network in order to achieve both semantic segmentation and Mask R-CNN based instance level recognition. We introduce a novel heuristic fusion approach for panoptic segmentation. The instance and semantic segmentation output of the network is fused into a panoptic segmentation. This is achieved using object sub-category class and instance propagation guidance by object category class from semantic segmentation. The proposed solution achieves significant improvements in semantic object segmentation and object mask boundaries refinement at low computational costs.**

I. INTRODUCTION

Semantic segmentation and instance recognition enable a thorough understanding of the environment at image pixel level. Semantic segmentation identifies the semantic class of each pixel, while instance segmentation provides an object-level representation by assigning instance labels to each object pixel. Extensive research is carried out for solving both tasks using deep convolutional neural networks. Most solutions are built on dilated ResNet [14] and Fully Convolutional Neural Networks (FCN) [29][3][44]. In the case of instance segmentation, significant improvements have been achieved by the Mask R-CNN framework [13] where a Feature Pyramid Network [27] provides a multi-scale feature representation for object detection and instance segmentation.

Semantic segmentation performs particularly well in the case of background classes but struggles in recognizing object subcategories or large-scale objects. Due to the fact that classification is achieved at pixel level, an object may receive multiple labels. In the case of Mask R-CNN, objects are detected and classified as a whole and the class is propagated to every pixel of the instance mask, hence an object is assigned a unique semantic label. However, the instance mask is computed at a lower resolution (28×28) resulting in a coarser boundary for large-scale objects.

In order to alleviate the downsides we employ a unified architecture consisting of a shared backbone and individual network heads for each tasks, and propose a fusion scheme. Kirillov et al. introduce in [19] a novel task and data format, the panoptic segmentation. It unifies semantic segmentation and instance segmentation by requiring both semantic class and instance ID for each individual pixel. The authors in [19] also introduce a baseline heuristic approach for generating the panoptic segmentation from semantic and instance segmentation. The fused output starts from non-overlapping instance segments, generated by a NMS-like procedure, and combines it with semantic segmentation by resolving any overlap between foreground and background in favor of the foreground class (from instance segmentation). The output fusion approach described in this paper is a novel heuristic approach for generating panoptic segmentation from semantic and instance segmentation.

The main contributions of this paper are:

- improved semantic segmentation network head;
- novel output fusion heuristic approach for panoptic segmentation.

II. RELATED WORK

State-of-the-art semantic segmentation methods use Fully Convolutional Neural Networks (FCN) [29] for dense pixel predictions. Classification networks usually learn features at 5 different scales, with the final layer having a 32x lower resolution than the input. Consecutive striding is harmful for the semantic segmentation task since the final segmentation map is obtained by upsampling the last layer of the CNN. In order to recover the resolution loss, several architectures have been proposed.

Atrous convolutions have been extensively used for controlling the resolution output [2][3]. The authors modified the original ResNet [14] architecture by adopting dilated (atrous) convolutions with various rates in the last or last two residual blocks. Atrous convolutions enlarge the field of view of filters by capturing multi-scale context without decimating the resolution. Bilinear upsampling is applied on top of the last layer to obtain the final segmentation map. An alternative to atrous convolutions represents scale-adaptive convolutions [43] which are capable of learning dilation rates.

Spatial Pyramid Network models are usually built on top of a dilated FCN and add a Spatial Pyramid Module. In PSPNet [44], this module encodes global information by applying various size average pooling kernels at the last layer. DeepLabV2 [2] introduces Atrous Spatial Pyramid Pooling (ASPP) which performs parallel atrous convolutions with different rates.

Encoder-decoder networks usually use a deeper and narrower CNN for feature extraction and a more complex decoder replaces bilinear interpolation. ENet [30] model

*Both authors contributed equally to this work.

Image Processing and Pattern Recognition Group, Computer Science Department, Technical University of Cluj-Napoca, Romania
{arthur.costea, andra.petrovai, sergiu.nedevschi} @cs.utcluj.ro

978-1-7281-0323-5/18/$31.00 ©2018 IEEE

learns the upsampling of low resolution features with deconvolution layers. The network runs in real time at the cost of reduced performance. ERFNet [33] obtaines better results by employing a residual network with factorized convolutions. SegNet [1] has a symmetric encoder-decoder architecture and introduces the unpooling layer for upsampling. The U-net model [34] uses shortcut connections from the encoder to decoder to help recover object details and spatial information.

Another approach to capture multi-scale information is to resize the input samples at different resolutions and use a shared feature extractor [9]. The resulted feature maps at different scales are aggregated with concatenation [26] or attention models [4].

A perception system usually performs multiple tasks such as semantic segmentation, object detection, instance segmentation and others. Having separate models for each task implies a very high computational cost and memory footprint, which in most cases is infeasible. A solution to this problem represents multi-task learning by having a shared CNN model that learns to optimize the tasks simultaneously. Since scene understanding can be achieved by perceiving both semantic and structure information, combining complementary tasks in a unified framework is beneficial for each particular task. In [17], the authors design a CNN that jointly learns semantic segmentation, instance segmentation and depth regression and propose a new multi-task loss which efficiently weights the loss of each task. UberNet [20] trains in an end-to-end manner a CNN that addresses several classification and regression tasks. [7] predicts depth, surface normals and semantic labels using a single multi-scale architecture. MultiNet [37] enables real time applications such as autonomous driving with a very efficient network, solving vehicle detection and road segmentation. A top performing framework for object detection and instance segmentation is Mask R-CNN [13], which extends Faster R-CNN [32] with an instance segmentation branch. The unified architecture employs Feature Pyramid Network [27] to learn multi-scale representations.

III. UNIFIED NETWORK ARCHITECTURE

In this work, we develop a unified network architecture based on CNN which simultaneously performs object detection, instance segmentation and semantic segmentation. Our goal is to design a model that can be trained end-to-end with a single optimization step.

State-of-the-art semantic segmentation networks based on ResNet usually employ atrous convolutions in the last two residual blocks such that the final feature responses are 8x or 16x smaller than the input resolution. Classification networks usually use a 32x downsampling factor, but the authors in [3] [5] have shown that semantic segmentation results are greatly affected by the signal decimation and adopting a smaller downsampling factor such as 8x leads to better performance at the cost of higher memory usage. The winning entry of the COCO Stuff Challenge 2017 [28] ”ResNeXt-FPN” [18] proposed by team FAIR has shown that a backbone network used for detection and classification such as ResNeXt [41] with a 32x downsampling factor can be successfully used to achieve state-of-the-art results in semantic segmentation. We use as baseline ”ResNet-FPN”, the ResNet variant of ”ResNeXt-FPN” solution for object detection, instance segmentation and semantic segmentation, and propose an improved semantic segmentation head.

A. Baseline model

The ”ResNet-FPN” baseline architecture is an extension of the Mask R-CNN framework for object detection and classification, and instance segmentation [13]. It is based on a Feature Pyramid Network (FPN) [27] defined over a ResNet [14][41] architecture. Mask R-CNN extends Faster R-CNN [32], an object detection and classification network. The Faster R-CNN detector consists of two stages: a Region Proposal Network (RPN) and RoIPool. The RPN proposes object candidates and the second stage extracts features using these candidates and applies bounding box classification and regression. Both stages use a shared feature representation based on ResNet and Feature Pyramid Network. The 4-level Feature Pyramid is built at the last layer of ResNet in a top-down manner by upsampling feature maps from higher pyramid levels and merging them via lateral connections with corresponding feature representations from the ResNet network. The process propagates coarser but semantically stronger features to the more finer feature maps, therefore each level of the pyramid will consist of more complex, richer features. The RoIPooling stage extracts features from different level of the pyramid according to scale. Mask R-CNN adds a mask prediction branch on top of Faster R-CNN, which outputs a binary mask for each RoI.

B. Segmentation head

For the segmentation task our model shares the same feature representation based on ResNet-FPN used by instance segmentation and object detection and classification as seen in Figure 1. Since the elements in a scene appear at various sizes depending on their distance to the camera, we find very important that our models learns multi-scale features. We employ multiple mechanism for addressing multiple scales: the Feature Pyramid Network, atrous convolutions and multiple image scales.

The baseline output of the FPN from Mask R-CNN [13] consists of 256 feature maps at four scales: $1/4$, $1/8$, $1/16$ and $1/32$. At each of the four scales we add an individual segmentation head in order to capture multi-resolution features. Since segmentation represents a pixel-level classification task we employ FCN on top of FPN leyers. Dilated (atrous) convolutions are important tools for extracting context and long-range information. Since the top level features of the pyramid at $1/32$ and $1/16$ provide better localization and stronger semantics we further incorporate multi-scale information to the model by adopting an Atrous Spatial Pyramid (ASP). Therefore, we apply an ASP [3] for the segmentation heads at $1/32$ and $1/16$ by using one 1×1 convolution and three 3×3 dilated convolutions having dilation rates of 6, 12 and 18. Each convolution is followed

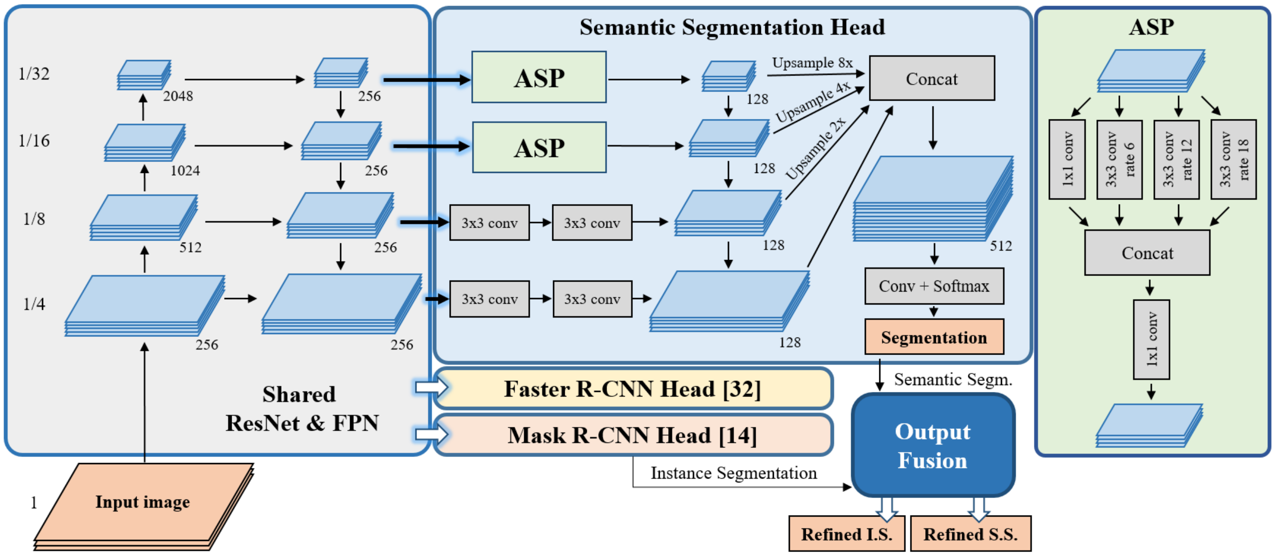

Fig. 1: A shared ResNet-FPN network is used for 3 tasks. The Faster-RCNN head performs object detection, Mask-RCNN head performs instance segmentation. Our semantic segmentation head is based on Atrous Spatial Pyramid (ASP)

by a Group Normalization layer [39] which we found more effective than Batch Normalization [15]. Batch normalization is a very important component in CNN that accelerates convergences by normalizing feature maps along the batch dimension. A problem arises when using small batch size and inaccurate batch statistics are computed, resulting in an increase in model error. In this work, we have a small batch size of 2 images due to GPU memory constraints. Therefore, we prefer the alternative to Batch Normalization, that is Group Normalization which is invariant to batch size, since normalization is done along groups of channels. Each Group Normalization layer is followed by a non-linear ReLU activation. The resulting feature maps are concatenated and passed through 128 1×1 filters. We note that we do not use pooling operations as in [3], which would constrain the model to a fixed input size, but instead we opt for multi-scale inputs. From the $1/8$ and $1/4$ levels we extract features of finer scales using two 3×3 convolutions as in [18]. At each of the four scales, the segmentation heads generate 128 features maps. For a further refinement, the outputs are fused in a pyramidal manner using a refinement pyramid (RP). Starting with the highest scale layer, the feature maps are upsampled two times and are added to the output of the following layer. This way, the features from each layer learn only the residues with respect to the higher-level layers. Next, the fused outputs are upsampled and concatenated into 512 feature maps at $1/4$. Finally, a 1×1 convolution is used to generate the class predictions.

To obtain the final segmentation predictions we use a per-pixel softmax and minimize the multinomial cross-entropy loss. During training, we optimize a multi-task loss defined as: $L = L_{box} + L_{cls} + L_{mask} + L_{segm}$

C. Training and optimization

In this subsection, we provide details about the training protocol. We train the model end-to-end with a single optimization step. We initialize our model with the pretrained Mask-RCNN [13] weights on the Microsoft COCO dataset [28] for the tasks of instance segmentation and object detection and classification. We employ stochastic gradient descent (SGD) with momentum 0.9 and a "poly" learning rate policy starting from $5e-3$. Our network converges in $32k$ iterations. As data augmentation, we adopt horizontal flipping and multiple image scales (from 0.8 to 1) at training time. Experiments were carried out on a system with 2 Nvidia 1080Ti GPUs with 2×11 GB memory. Due to memory limitation, batch size was set to 2 images (1/GPU). The original Mask-RCNN network uses Batch Normalization, but since we have a very small batch size, we freeze the Batch Normalization statistics and learn only the affine parameters γ and β. Group Normalization is employed only in the segmentation head.

IV. OUTPUT FUSION

We introduce a novel fusion approach for refining the outputs of semantic and instance segmentation. Our goal is to exploit the performance of FCN based pixel level classification for more general classes and whole object level based classification for instance classes. To achieve this, first, we divide the pixels into foreground and background based on the results from semantic segmentation. This partitioning results also in fine foreground/background boundaries due to stable pixel level classification.

Background In the case of background pixels we rely only on the classification from semantic segmentation in order to determine the semantic subclass of each pixel. In the case of background classes recognition can be achieved also with a

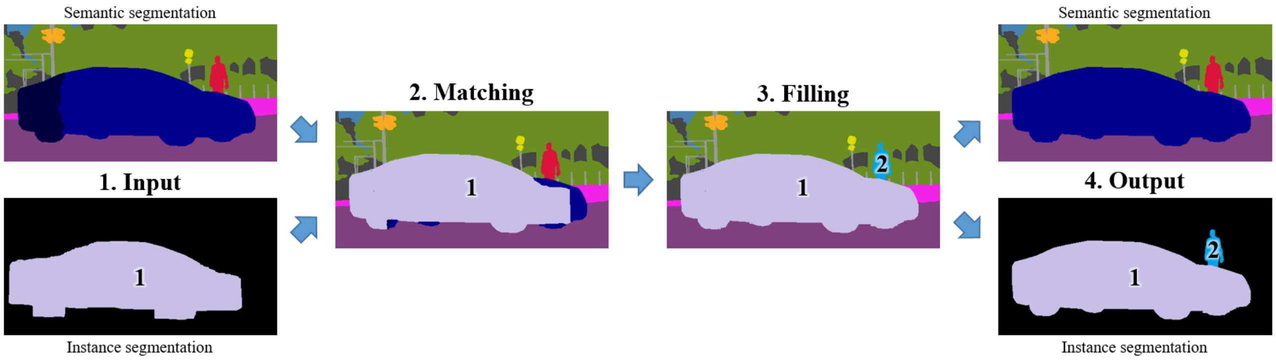

Fig. 2: Fusion process overview. (1) Input: semantic segmentation (car is partially classified as truck) and instance segmentation (mask for car is slightly misaligned and cropped, and the pedestrian behind the car is not detected); (2) Matching: the pixels of the instance segmentation are matched to the pixels from semantic segmentation only if the object class is compatible with the semantic category from semantic segmentation; (3) Filling: semantic region growing is applied to finalize the object shape and the unmatched object segments receive a new object ID; (4) Output: refined segmentation.

simpler classifier with a smaller receptive field that is able to capture color and texture.

Foreground In the case of foreground pixels we use the semantic segmentation to determine the semantic category of each pixel. Note that semantic segmentation approaches generally perform well when segmenting at category level and struggle at subcategory level [6]. To establish the semantic subcategory of each foreground pixel we take into consideration only the classification results from object detection and use the instance segmentation mask in order to guide a pixel-to-pixel matching. The class label and the instance label of a pixel from object detection is retained only if it is consistent with the semantic category of that pixel from the semantic segmentation. The instance mask pixels that correspond to background pixels or to a different semantic subcategory are deleted. After the pixel-to-pixel matching it is possible to have foreground pixels that were not covered by object masks. In this case these pixels are matched to the closest labeled pixel with direct semantic path. This labeling extension can be achieved through a breadth-first-search based region growing. This way, all pixels of an object receive a unique class label resulting in a more stable object level classification in comparison with pixel level classification. The instance masks are aligned with a more precise pixel level semantic segmentation, therefore having better object boundaries. Note that in Mask R-CNN [13] the masks are obtained using a convolutional neural network defined over a grid of 28×28 sampled points. Due to the low resolution of the sampling grid there can be misalignments for the raw masks at the objects' boundaries especially in the case of large objects.

In the case of pixel segments that were labeled as foreground but did not receive labels after region growing (due to being isolated), we assign the semantic subcategory label from semantic segmentation and generate a new instance ID. This way we are able to extend the list of instances with objects that were not initially detected. A threshold has to be employed for the segment size in order to avoid the instance labeling of small foreground noise segments. The steps of the fusion process are illustrated in Figure 2.

Method	backbone	AP mask	mIoU
Mask-RCNN [13]	ResNet50	36.4	-
PSPNet [44]	ResNet50-dilated	-	71.7
Unified baseline	ResNet50-FPN	37.0	71.6
+ ASP and RP	ResNet50-FPN	37.2	72.9
+ fusion	ResNet50-FPN	37.3	76.0

TABLE I: Instance and semantic segmentation results on the Cityscapes *VALIDATION* set. Our model is trained only on the fine Cityscapes training set and no test-time augmentation was used.

The output fusion scheme can be applied for any semantic segmentation and instance segmentation output without depending on the employed approaches. It can be used as a fast post processing step.

V. EXPERIMENTAL RESULTS

We evaluate the proposed model on the Cityscapes dataset [6] which provides semantic segmentation (19 classes) and instance segmentation (8 classes) ground truth data for 5000 pixel-level annotated traffic scenes images. Evaluation for semantic segmentation is performed using the standard average Intersection-Over-Union (IoU) metric, while for instance segmentation Average Precision (AP) is used.

The experiments were carried out using the Detectron [12] framework. We train our models with a ResNet50-FPN backbone from [12] that was pretrained for object detection and mask prediction on MS COCO.

In Table I we present the results for our ResNet50-FPN based solution on Cityscapes validation set. Due to multi-objective learning, we observe an improvement of both instance segmentation and semantic segmentation with respect to state-of-the-art ResNet50-based Mask R-CNN [13] and PSPNet [44] solutions. Compared to the baseline, that uses two 3×3 convolutions for the segmentation head of each

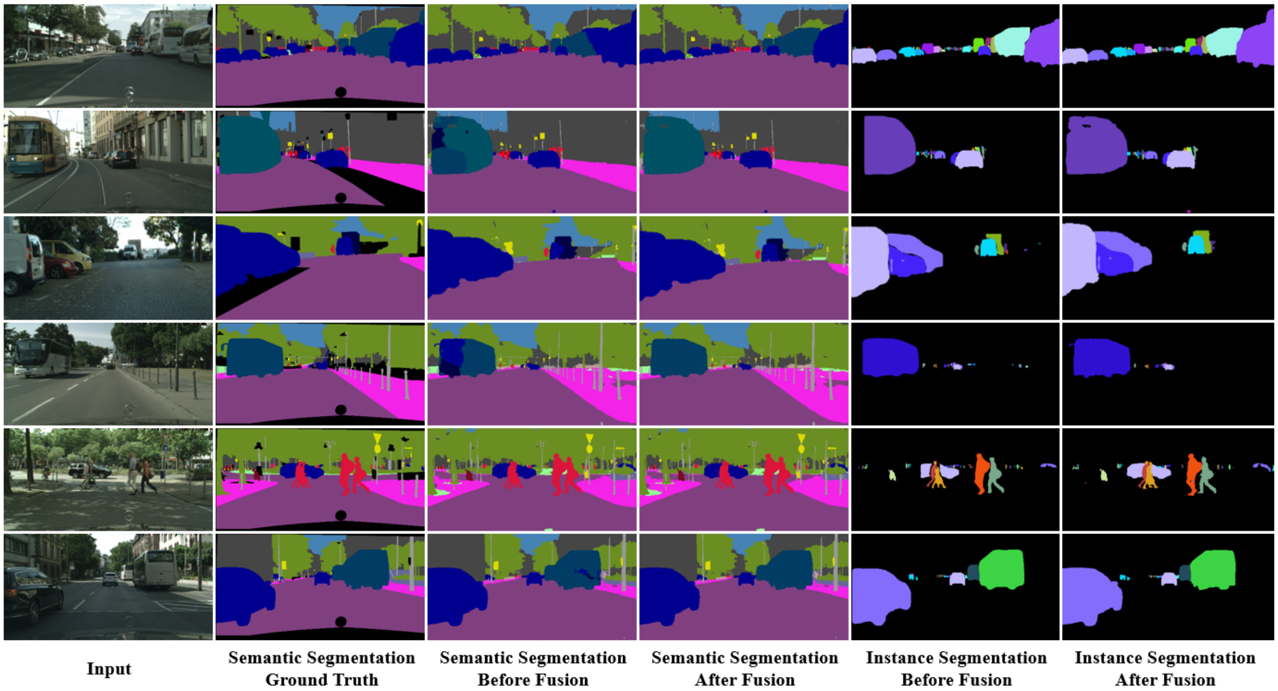
Input
Semantic Segmentation Ground Truth
Semantic Segmentation Before Fusion
Semantic Segmentation After Fusion
Instance Segmentation Before Fusion
Instance Segmentation After Fusion

Fig. 3: Demo results for fusion based semantic and instance segmentation refinement

Method	road	sidewalk	building	wall	fence	pole	traffic light	traffic sign	vegetation	terrain	sky	person	rider	car	truck	bus	train	motorbike	bike	mIoU
ResNet50-FPN+ASP	97.7	82.3	91.2	48.6	51.3	56.9	66.9	73.1	91.5	61.8	93.1	80.1	59.8	93.1	63.0	77.5	64.1	59.7	75.3	72.9
ResNet50-FPN+ASP+fusion	97.7	82.3	91.2	48.6	51.3	56.9	66.9	73.1	91.5	61.8	93.1	**81.0**	**65.6**	**94.0**	**81.6**	**89.8**	**80.1**	**61.9**	**75.6**	76.0

TABLE II: Class mIoU on Cityscapes *VALIDATION* set before and after fusion

layer, the Atrous Spatial Pyramid (ASP) and the refinement pyramid (RP) bring an improvement of approximately 1.5% in mIoU for semantic segmentation. The fusion scheme provides a further increase of 3% for semantic segmentation. The semantic segmentation for foreground classes is improved due to a more robust object level classification and the use of a unique label per instance. Moreover, the instance masks are better aligned with objects. The output fusion module also generates output in the form of panoptic segmentation as seen in Figure 4.

In Figure 3 we present demo results for semantic and instance segmentation before and after fusion on Cityscapes validation images. In the case of semantic segmentation it can be seen that the pixel level classification can result in erroneous semantic labels for larger scale difficult objects, such as buses, trams or trucks. These errors are corrected after the fusion process due to the use of the object level classification results for the foreground classes. In the case of instance masks the improvements are more visible at the object boundaries due to the better alignment and preservation of details. The improvements are confirmed also by the results from Table II. A significant improvement in IoU is achieved in the case of large scale semantic classes for

Method	**mIoU class**
DeepLabv2-CRF [2]	70.4
Deep Layer Cascade [23]	71.1
ML-CRNN [8]	71.2
Adelaide context [25]	71.6
FRRN [31]	71.8
LRR-4x [11]	71.8
RefineNet [24]	73.6
FoveaNet [22]	74.1
Ladder DenseNet [21]	74.3
PEARL [16]	75.4
Global Local Refinement [42]	77.3
SAC multiple [43]	78.1
SegModel [36]	79.2
TuSimple [38]	80.1
Netwarp [10]	80.5
ResNet-38 [40]	80.6
PSPNet [44]	81.2
DeepLabV3 [3]	81.3
Mappilary [35]	82.0
DeepLabV3+ [5]	82.1
Proposed - ISS-Fusion	72.7

TABLE III: Cityscapes results on the *TEST* set

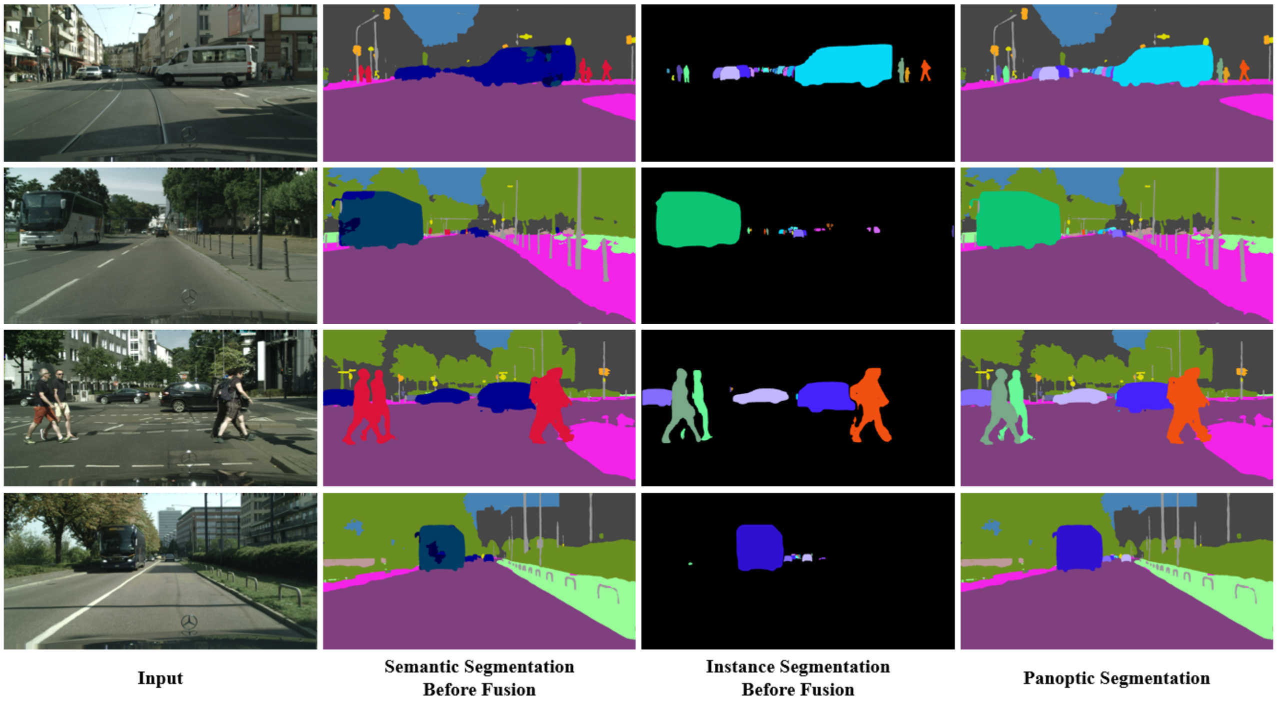
Input
Semantic Segmentation Before Fusion
Instance Segmentation Before Fusion
Panoptic Segmentation

Fig. 4: Panoptic segmentation by unifying semantic and instance segmentation

example truck, bus and train.

In Table III we provide a comparison with other approaches based on the performance on the Cityscapes test set. The proposed solution provides competitive results. Currently it is outperformed by solutions that use larger backbone networks such as ResNet101 which were trained using large batch sizes. Unfortunately, larger backbone networks increase computational costs and memory requirements. The network for semantic segmentation can be trained with image crops, but in the case of object detection and instance segmentation full images are preferred. Training with 2048 × 1024 pixel Cityscapes images is possible only with a single image per GPU. For our experiments we used two GPUs for training, however a batch size of 2 images can result in an unstable stochastic gradient descent during training. In order to further improve the results it is important to explore robust and memory efficient architectures, that would enable training with large batch sizes.

The execution time of the fusion scheme is 3 ms on a Nvidia GTX 1080Ti GPU. It is a fast post processing step which is achieved by linear parsings of the outputs.

VI. CONCLUSION

In this work, we propose a solution for improving both semantic segmentation and instance segmentation using a unified end-to-end learnable deep neural network architecture. The first contribution of the paper is an improved semantic segmentation network head based on Atrous Spatial Pyramids. The second contribution relies in a novel output fusion scheme for generating a panoptic segmentation that propagates instance labels based on semantic segmentation. The proposed solution provides improvements at low computational costs for both tasks.

ACKNOWLEDGMENT

This work was supported by the EU H2020 UP-Drive project under grant nr. 688652, SEPCA project PN III PCCF, no. 9/2018 and MULTISPECT project PN III PCE, no. 60/2017.

REFERENCES

[1] 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):2481–2495, 2017.

[2] L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille. Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. *IEEE transactions on pattern analysis and machine intelligence*, 40(4):834–848, 2018.

[3] L.-C. Chen, G. Papandreou, F. Schroff, and H. Adam. Rethinking atrous convolution for semantic image segmentation. In *arXiv preprint arXiv:1706.05587*, 2017.

[4] L.-C. Chen, Y. Yang, J. Wang, W. Xu, and A. L. Yuille. Attention to scale: Scale-aware semantic image segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3640–3649, 2016.

[5] L.-C. Chen, Y. Zhu, G. Papandreou, F. Schroff, and H. Adam. Encoder-decoder with atrous separable convolution for semantic image segmentation. *arXiv preprint arXiv:1802.02611*, 2018.

[6] 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 *CVPR*, 2016.

[7] D. Eigen and R. Fergus. Predicting depth, surface normals and semantic labels with a common multi-scale convolutional architecture. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2650–2658, 2015.

[8] H. Fan, X. Mei, D. Prokhorov, and H. Ling. Multi-level contextual rnns with attention model for scene labeling. *IEEE Transactions on Intelligent Transportation Systems*, pages 1–11, 2018.

[9] C. Farabet, C. Couprie, L. Najman, and Y. LeCun. Learning hierarchical features for scene labeling. *IEEE transactions on pattern analysis and machine intelligence*, 35(8):1915–1929, 2013.

[10] R. Gadde, V. Jampani, and P. V. Gehler. Semantic video cnns through representation warping. In *2017 IEEE International Conference on Computer Vision (ICCV)*, pages 4463–4472, Oct 2017.

[11] G. Ghiasi and C. C. Fowlkes. Laplacian pyramid reconstruction and refinement for semantic segmentation. In *European Conference on Computer Vision*, pages 519–534. Springer, 2016.

[12] R. Girshick, I. Radosavovic, G. Gkioxari, P. Dollár, and K. He. Detectron. `https://github.com/facebookresearch/detectron`, 2018.

[13] K. He, G. Gkioxari, P. Dollár, and R. Girshick. Mask r-cnn. In *ICCV*, 2017.

[14] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *CVPR*, 2016.

[15] S. Ioffe and C. Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *Proceedings of the 32Nd International Conference on International Conference on Machine Learning - Volume 37*, ICML'15, pages 448–456. JMLR.org, 2015.

[16] X. Jin, X. Li, H. Xiao, X. Shen, Z. Lin, J. Yang, Y. Chen, J. Dong, L. Liu, Z. Jie, J. Feng, and S. Yan. Video scene parsing with predictive feature learning. In *2017 IEEE International Conference on Computer Vision (ICCV)*, pages 5581–5589, Oct 2017.

[17] A. Kendall, Y. Gal, and R. Cipolla. Multi-task learning using uncertainty to weigh losses for scene geometry and semantics. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.

[18] A. Kirillov, K. He, R. Girshick, and P. Dollár. A unified architecture for instance and semantic segmentation. `http://presentations.cocodataset.org/COCO17-Stuff-FAIR.pdf`, 2017.

[19] A. Kirillov, K. He, R. Girshick, C. Rother, and P. Dollr. Panoptic segmentation. *arXiv preprint arXiv:1801.00868*, 2018.

[20] I. Kokkinos. Ubernet: Training a universal convolutional neural network for low-, mid-, and high-level vision using diverse datasets and limited memory. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5454–5463, 2017.

[21] J. Krapac and I. K. S. egvic. Ladder-style densenets for semantic segmentation of large natural images. In *2017 IEEE International Conference on Computer Vision Workshops (ICCVW)*, pages 238–245, Oct 2017.

[22] X. Li, Z. Jie, W. Wang, C. Liu, J. Yang, X. Shen, Z. Lin, Q. Chen, S. Yan, and J. Feng. Foveanet: Perspective-aware urban scene parsing. In *2017 IEEE International Conference on Computer Vision (ICCV)*, pages 784–792, Oct 2017.

[23] X. Li, Z. Liu, P. Luo, C. C. Loy, and X. Tang. Not all pixels are equal: Difficulty-aware semantic segmentation via deep layer cascade. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 6459–6468, July 2017.

[24] G. Lin, A. Milan, C. Shen, and I. Reid. Refinenet: Multi-path refinement networks for high-resolution semantic segmentation. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

[25] G. Lin, C. Shen, A. v. d. Hengel, and I. Reid. Efficient piecewise training of deep structured models for semantic segmentation. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3194–3203, June 2016.

[26] G. Lin, C. Shen, A. Van Den Hengel, and I. Reid. Efficient piecewise training of deep structured models for semantic segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3194–3203, 2016.

[27] T.-Y. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan, and S. Belongie. Feature pyramid networks for object detection. In *CVPR*, 2017.

[28] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick. Microsoft coco: Common objects in context. In *ECCV*, 2014.

[29] J. Long, E. Shelhamer, and T. Darrell. Fully convolutional networks for semantic segmentation. In *CVPR*, 2015.

[30] A. Paszke, A. Chaurasia, S. Kim, and E. Culurciello. Enet: A deep neural network architecture for real-time semantic segmentation. *arXiv preprint arXiv:1606.02147*, 2016.

[31] T. Pohlen, A. Hermans, M. Mathias, and B. Leibe. Full-resolution residual networks for semantic segmentation in street scenes. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3309–3318, July 2017.

[32] S. Ren, K. He, R. Girshick, and J. Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. In *Proceedings of the 28th International Conference on Neural Information Processing Systems - Volume 1*, NIPS'15, pages 91–99, Cambridge, MA, USA, 2015. MIT Press.

[33] E. Romera, J. M. Alvarez, L. M. Bergasa, and R. Arroyo. Erfnet: Efficient residual factorized convnet for real-time semantic segmentation. *IEEE Transactions on Intelligent Transportation Systems*, 19(1):263–272, 2018.

[34] O. Ronneberger, P. Fischer, and T. Brox. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*, pages 234–241. Springer, 2015.

[35] S. Rota Bulò, L. Porzi, and P. Kontschieder. In-place activated batchnorm for memory-optimized training of dnns. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018.

[36] F. Shen, R. Gan, S. Yan, and G. Zeng. Semantic segmentation via structured patch prediction, context crf and guidance crf. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5178–5186, July 2017.

[37] M. Teichmann, M. Weber, M. Zoellner, R. Cipolla, and R. Urtasun. Multinet: Real-time joint semantic reasoning for autonomous driving. *arXiv preprint arXiv:1612.07695*, 2016.

[38] P. Wang, P. Chen, Y. Yuan, D. Liu, Z. Huang, X. Hou, and G. Cottrell. Understanding convolution for semantic segmentation. In *arXiv preprint arXiv:1702.08502*, 2017.

[39] Y. Wu and K. He. Group normalization. *arXiv preprint arXiv:1803.08494*, 2018.

[40] Z. Wu, C. Shen, and A. v. d. Hengel. Wider or deeper: Revisiting the resnet model for visual recognition. *arXiv preprint arXiv:1611.10080*, 2016.

[41] S. Xie, R. Girshick, P. Dollár, Z. Tu, and K. He. Aggregated residual transformations for deep neural networks. In *CVPR*, 2017.

[42] R. Zhang, S. Tang, M. Lin, J. Li, and S. Yan. Global-residual and local-boundary refinement networks for rectifying scene parsing predictions. In *Proceedings of the 26th International Joint Conference on Artificial Intelligence*, pages 3427–3433. AAAI Press, 2017.

[43] R. Zhang, S. Tang, Y. Zhang, J. Li, and S. Yan. Scale-adaptive convolutions for scene parsing. In *Proc. 26th Int. Conf. Comput. Vis.*, pages 2031–2039, 2017.

[44] H. Zhao, J. Shi, X. Qi, X. Wang, and J. Jia. Pyramid scene parsing network. In *CVPR*, 2017.