HPS: Holistic End-to-End Panoptic Segmentation Network with Interrelations

Günther Kniewasser, Alexander Grabner, Peter M. Roth
Institute of Computer Graphics and Vision, Graz University of Technology, Austria
{guenther.kniewasser@student,alexander.grabner@icg, pmroth@icg}.tugraz.at

Abstract. *To provide a complete 2D scene segmentation, panoptic segmentation unifies the tasks of semantic and instance segmentation. For this purpose, existing approaches independently address semantic and instance segmentation and merge their outputs in a heuristic fashion. However, this simple fusion has two limitations in practice. First, the system is not optimized for the final objective in an end-to-end manner. Second, the mutual information between the semantic and instance segmentation tasks is not fully exploited. To overcome these limitations, we present a novel end-to-end trainable architecture that generates a full pixel-wise image labeling with resolved instance information. Additionally, we introduce interrelations between the two subtasks by providing instance segmentation predictions as feature input to our semantic segmentation branch. This inter-task link eases the semantic segmentation task and increases the overall panoptic performance by providing segmentation priors. We evaluate our method on the challenging Cityscapes dataset and show significant improvements compared to previous panoptic segmentation architectures.*

1. Introduction

Panoptic segmentation [12] addresses the problem of complete 2D scene segmentation by not only assigning a class label to each pixel of an image but also differentiating between instances within a common class. Thus, it can be seen as a unification of semantic segmentation [22, 24, 3] and instance segmentation [8, 13, 20, 16]. Panoptic segmentation is a new and active research area with applications in augmented reality, robotics, and medical imaging [5, 23, 30].

To predict a panoptic segmentation of an image, recent approaches perform three tasks. First, they perform semantic segmentation to identify regions of uncountable *stuff* classes like sky. Second, they perform instance segmentation to detect individual instances of countable *things* classes like cars. Third, they merge the outputs of these two tasks into a single panoptic prediction.

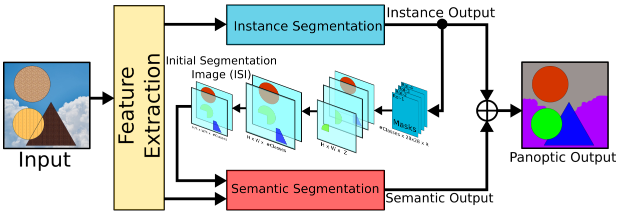

Figure 1: Illustration of our proposed panoptic segmentation network with task interrelations. We provide instance segmentation predictions as additional feature input to our semantic segmentation branch. In this way, we exploit a segmentation prior which increases the overall panoptic performance.

However, this strategy has two limitations in practice. First, because the panoptic output is generated using heuristics, the system cannot be optimized for the final objective in an end-to-end manner. Second, semantic and instance segmentation share mutual information and similarities but the relation between the two tasks is not exploited because they are addressed independently.

To overcome these limitations, we propose a *holistic* end-to-end trainable network for *panoptic segmentation* (HPS) with interrelations between the semantic and the instance segmentation branches, as shown in Figure 1. Our network directly generates a full pixel-wise image labeling with resolved instance information by using differentiable operations instead of heuristics to combine individual results. Moreover, to take advantage of mutual information between the semantic and the instance segmentation tasks, we provide instance segmentation predictions as additional feature input to our semantic segmentation branch. In particular, we gather predicted instance masks into an *initial segmentation image* (ISI) which represents a coarse semantic segmentation for *things* classes. In this way, we exploit a segmentation prior which increases the overall panoptic performance of our system by leveraging similarities between the two previously disjoint subtasks.

We evaluate our method on the challenging Cityscapes dataset [4] for semantic understanding of urban street scenes using the recently introduced panoptic quality [11] metric. We provide an unbiased evaluation and compare four different approaches with an increasing level of entanglement between semantic and instance segmentation. Our experiments show that both end-to-end training and inter-task relations improve panoptic performance in practice.

2. Related Work

Fusing semantic and instance information has a rich history in computer vision [25, 26]. However, only recently [12] formalized the task of panoptic segmentation and introduced a panoptic quality (PQ) metric to assess the performance of complete 2D scene segmentation in an interpretable and unified manner. This formalization and the availability of large datasets with corresponding annotations [19] motivated research on panoptic segmentation.

Early approaches to panoptic segmentation use two highly specialized networks for semantic segmentation [22, 24, 3] and instance segmentation [21, 8, 17, 27] and combine their predictions heuristically [1]. Instead, recent methods address the two segmentation tasks with a single network by training a multi-task system that performs semantic and instance segmentation on top of a shared feature representation [11]. This reduces the number of parameters, the computational complexity, and the time required for training. To improve the panoptic quality, newer approaches propose a differentiable fusion of semantic and instance segmentation instead of a heuristic combination. In this way, they learn to combine the individual predictions and optimize directly for the final objective in an end-to-end manner. For example, UPSNet [28] introduces a parameter-free merging technique to generate panoptic predictions using a single network.

Another strategy to improve accuracy is to exploit mutual information and similarities between semantic and instance segmentation network branches. In this context, AUNet [15] incorporates region proposal information as an attention mechanism in the semantic segmentation branch. In this way, the semantic segmentation focuses more on *stuff* classes and less on *things* classes, which are eventually replaced by predicted instance masks. TASCNet [14] enforces L2-consistency between predicted semantic and instance segmentation masks to exploit mutual information. SOGNet [29] addresses the overlapping issue of instances using a scene graph representation which computes a relational embedding for each object based on geometry and appearance.

Similar to our approach, IMP [6] which has been developed at the same time uses predicted instance segmentation masks as additional input for the semantic segmentation branch. Compared to our approach, a different normalization technique is used and the instance masks are combined using the max operator instead of averaging.

3. Holistic End-to-End Panoptic Segmentation Network with Interrelations

An overview of our end-to-end trainable panoptic segmentation network with inter-task relations is shown in Figure 1. We first present our end-to-end trainable architecture which combines semantic and instance segmentation predictions in a differentiable way in Sec. 3.1. Then, we introduce our interrelations module which provides instance segmentation predictions as additional feature input to our semantic segmentation branch in Sec. 3.2.

3.1. End-to-End Panoptic Architecture

Our network architecture builds upon Panoptic Feature Pyramid Networks [11]. Like many recent panoptic segmentation methods, this approach extends the generalized Mask R-CNN framework [8] with a semantic segmentation branch. This results in a multi-task network that predicts a dense semantic segmentation in addition to sparse instance segmentation masks. For our implementation, we use a shared ResNet-101 [9] feature extraction backbone with a Feature Pyramid Network [18] architecture to obtain combined low- and high-level features. These features serve as shared input to our semantic and instance segmentation branches, as shown in Figure 2.

For the semantic segmentation branch, we process each stage of the feature pyramid $\{P2, \ldots, P5\}$ by a series of upsampling modules. These modules con-

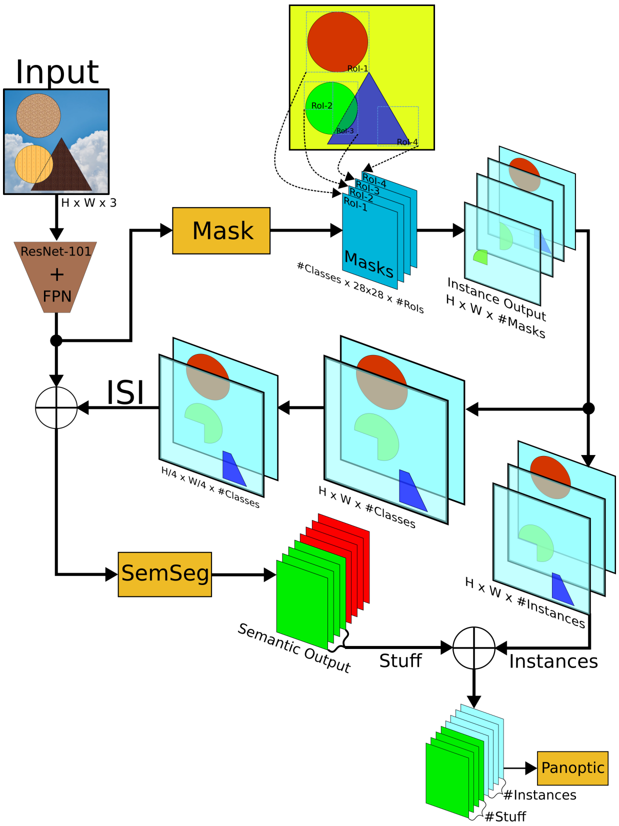

Figure 2: Detailed illustration of our end-to-end panoptic segmentation network with task interrelations. We internally merge predictions from our semantic and instance segmentation branches in a differentiable way. In particular, we concatenate *stuff* class predictions from our semantic segmentation branch with *things* class predictions in the form of canvas collections from our instance segmentation branch. Our instance canvas collections can also be transformed into an initial segmentation image (ISI) which serves as additional feature input for our semantic segmentation branch.

sists of 3×3 convolutions, batch normalization [10], ReLU [7], and $2\times$ bilinear upsampling. Because the individual stages have different spatial dimensions, we process each stage by a different number of upsampling modules to generate $H/4 \times W/4 \times 128$ feature maps, where H and W are the input image dimensions. The resulting outputs of all stages are concatenated and processed using a final 1×1 convolution to reduce the channel dimension to the desired number of classes.

For the instance segmentation branch, we implemented a Mask R-CNN [8]. We use a region proposal network to detect regions of interest, perform non-maximum suppression, execute ROI alignment, and predict 28×28 binary masks as well as class probabilities for each detected instance.

In order to combine the semantic and instance segmentation outputs, we use an internal differentiable fusion instead of external heuristics. For this purpose, we first select the most likely class label for each detected instance using a differentiable

$$\textit{soft argmax} = \sum_{i}^{N} \lfloor \frac{e^{z_i \cdot \beta}}{\sum_{k}^{N} e^{z_k \cdot \beta}} \rceil \cdot i \qquad (1)$$

operation [2], where N is the number of *things* classes, β is a large constant, and z is the predicted class logit. Using β in the exponent in combination with the round function allows us to squash all non-maxium values to zero. In this way, we approximate the non-differentiable *argmax* function, allowing us to backpropagate gradients.

We then resize the predicted 28×28 mask logits for each detected instance according to its predicted 2D bounding box size and place them in empty canvas layers at the predicted 2D location, as shown in Figure 2 (*top right*). Additionally, we merge the canvas layers for regions of interest with the same class id and high mask IOU. The resulting canvas collection from the instance segmentation branch is then concatenated with the *stuff* class logits of the semantic segmentation branch to generate our panoptic output, as illustrated in Figure 2 (bottom). The pixel-wise panoptic segmentation output is attained by applying a softmax layer on top of the stacked semantic and instance segmentation information. The shape of the final output is $H \times W \times (\#\,\textit{stuff classes} + \#\,\textit{detected instances})$. For *stuff* classes, the output is a class ID. For *things* classes, the output is an instance ID. The corresponding class ID for each instance can be gathered from our semantic or instance segmentation output.

During training, it is important to reorder the detected instances to match the order of the ground truth instances. For this purpose, we use a ground truth instance ID lookup table. All parameters of our network are optimized jointly.

3.2. Inter-task Relations

Our differentiable fusion of semantic and instance segmentation predictions allows us to join the outputs of our two branches internally for end-to-end training. However, it also allows us to provide instance predictions as additional feature input to our semantic segmentation branch, as shown in Figure 3.

For this purpose, we first evaluate our instance segmentation branch and build an instance canvas collection as described in Sec. 3.1. Next, we merge canvas layers of instances that belong to the same class using weighted average and insert empty canvas layers for missing or undetected classes. In this way, we generate an *initial segmentation image* (ISI) which represents a coarse semantic segmentation for *things* classes.

To exploit this segmentation prior in our semantic segmentation branch, we downsample our ISI to $H/4 \times W/4 \times$*# things classes* and concatenate it with the output of our semantic segmentation upsampling modules, as shown in Figure 3. Next, we apply four network blocks consisting of 3×3 convolution, batch normalization, and ReLU followed by a single 1×1 convolution, batch normalization, and ReLU block to reduce the channel dimension to the number of classes. Finally, we use bilinear upsampling to obtain semantic segmentation logits at the original input image dimensions and apply a softmax non-linearity.

By exploiting the segmentation prior given by ISI, the upsampling modules of our semantic segmentation branch focus more on the prediction of *stuff* classes and boundaries between individual classes instead of *things* classes. This is a huge advantage compared to disjoint semantic and instance segmentation branches where redundant predictions are performed in the semantic segmentation branch. As a consequence, this link between the individual tasks increases the panoptic performance of our system.

4. Experimental Results

To demonstrate the benefits of our end-to-end panoptic architecture with interrelations, we evaluate it on the challenging Cityscapes dataset [4] for semantic understanding of urban street scenes. We follow the protocol of [4] and train and evaluate on 19 classes (11 *stuff* and 8 *things*). We use the recently introduced panoptic quality [11] metric to assess the segmentation performance.

4.1. Experimental Setup

Due to our limited computational resources, we limited the maximum number of instances per image to 30 and excluded samples with more instances from the evaluation. In this way, we use 2649 of 2975 training images ($\approx 89\%$) and 415 of 500 publicly available validation images ($\approx 83\%$). Additionally, we reduce the spatial image resolution from 2048×1024 to 1024×512. For this reason, we cannot not benchmark against other state-of-the-art approaches. To provide an unbiased evaluation, we compare four different approaches with an increasing level of entanglement between semantic and instance segmentation. All methods use the same backbone, training protocol, and hyper-parameters:

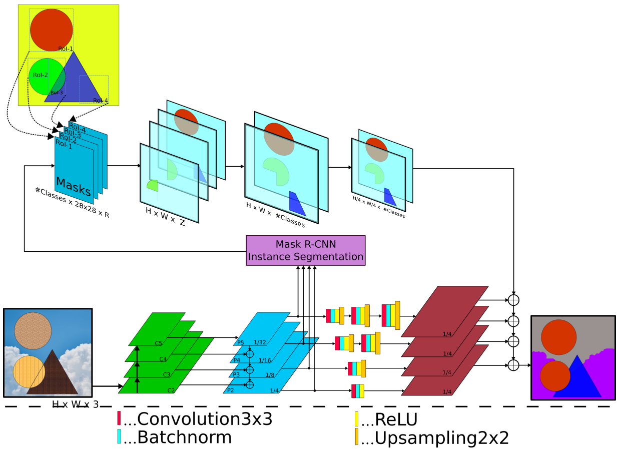

Figure 3: Illustration of our proposed semantic and instance segmentation branches with inter-task relations. We first run the instance segmentation branch and then provide instance segmentation predictions as additional feature input to the semantic segmentation branch via an initial segmentation image (ISI). Finally, we evaluate the semantic segmentation branch and exploit the segmentation prior given by ISI to improve the overall panoptic performance.

Semantic + Instance. This approach uses two different networks based on a ResNet-101 [9] backbone which independently perform semantic and instance segmentation. A heuristic is used to combine the individual results.

Panoptic FPN. This method is a reimplementation of Panoptic Feature Pyramid Networks [11] with a ResNet-101 [9] backbone. In contrast to *Semantic + Instance*, the semantic and instance segmenation branches use a single shared feature representation. The results, however, are still merged heuristically.

HPS. Our holistic panoptic segmentation network (HPS) extends *Panoptic FPN* as described in Sec. 3.1. Our network internally builds the panoptic segmentation output using differentiable operations which enables us to optimize for the final objective.

HPS + ISI. This method augments our *HPS* with inter-task relations between the semantic and in-

Method	PQ	SQ	RQ	PQ^{Th}	SQ^{Th}	RQ^{Th}	PQ^{St}	SQ^{St}	RQ^{St}
Semantic + Instance	40.6	70.9	51.3	40.3	75.4	53.0	40.9	67.6	50.0
Panoptic FPN	41.9	73.7	53.4	43.0	75.2	56.6	41.2	72.5	51.1
HPS	42.9	74.5	54.3	43.4	75.7	56.7	42.6	73.6	52.5
HPS + ISI	**44.0**	**74.8**	**55.5**	**44.4**	**76.4**	**57.5**	**43.7**	**73.6**	**54.1**

Table 1: Quantitative results on the Cityscapes dataset. The results show that a shared feature backbone reduces overfitting compared to two disjoint networks (*Semantic + Instance* vs *Panoptic FPN*). Also, generating the final panoptic output internally and training the system end-to-end increases the performance (*Panoptic FPN* vs *HPS*). Finally, using inter-task relations in the form of an initial segmentation image (ISI) provides an effective segmentation prior and increases the overall panoptic quality as well as all other metrics (*HPS* vs *HPS + ISI*).

stance segmentation branches by using an initial segmentation image (ISI), as introduced in Sec. 3.2.

4.2. Results

The thus obtained results of the four methods described above on the Cityscapes dataset are summarized in Table 1. In addition, to the panoptic quality (PQ), we show the segmentation quality (SQ) and the recognition quality (RQ) for all classes, *things* (Th) classes only, and *stuff* (St) classes only. Since PQ is a measurement of semantic (SQ) and instance (RQ) segmentation quality an improvement in either part will increase the accuracy of the overall system.

Interestingly, *Semantic + Instance* performs worse than *Panoptic FPN*. We hypothesize that this is because the number of training images in Cityscapes is low. Thus, the shared feature backbone of *Panoptic FPN* acts as a regularizer which reduces overfitting compared to training two individual networks without shared features on this dataset.

Next, *HPS* improves upon *Panoptic FPN* across all metrics and classes, because we optimize for the final panoptic segmentation output. Our system minimizes a panoptic loss in addition to the semantic and instance segmentation losses which provides better guidance for the network. In this way, we do not rely on the heuristic merging of subtask predictions but directly generate the desired output internally which results in improved accuracy in practice.

Finally, *HPS + ISI* significantly outperforms all other methods because it additionally leverages inter-task relations. Compared to *Panoptic FPN*, *HPS + ISI* improves PQ by $+5\%$ relative from 41.9 to 44.0. Providing instance segmentation predictions as additional feature input for the semantic segmentation branch gives a segmentation prior. By exploiting this prior, the semantic segmentation branch can focus more on the prediction of *stuff* classes and boundaries between individual classes which results in improved accuracy across all metrics. Additionally, our architectural advances only add a neglible computational overhead during both training and inference compared to *Panoptic FPN*.

This quantitative improvement is also reflected qualitatively, as shown in Figure 4. We observe that *HPS + ISI* handles occlusions more accurately (1st row) and resolves overlapping issues on its own while being less sensitive to speckle noise in semantically coherent regions (2nd row). Thanks to our end-to-end training and inter-task relations, we predict more accurate semantic label transitions (3rd row) and reduce confusion between classes with similar semantic meaning like *bus* and *car* (4th row).

5. Conclusion

Panoptic segmentation is a challenging but important and practically highly relevant problem. As approaching panoptic segmentation by independently addressing semantic and instance segmentation has several limitations, we propose a single end-to-end trainable network architecture that directly optimizes for the final objective. Moreover, we present a way to share mutual information between the tasks by providing instance segmentation predictions as additional feature input for our semantic segmentation branch. This inter-task link allows us to exploit a segmentation prior and improves the overall panoptic quality. In this way, our work is a first step towards fully entangled panoptic segmentation.

Acknowledgment. This work was partially supported by the Christian Doppler Laboratory for Semantic 3D Computer Vision, funded in part by Qualcomm Inc.

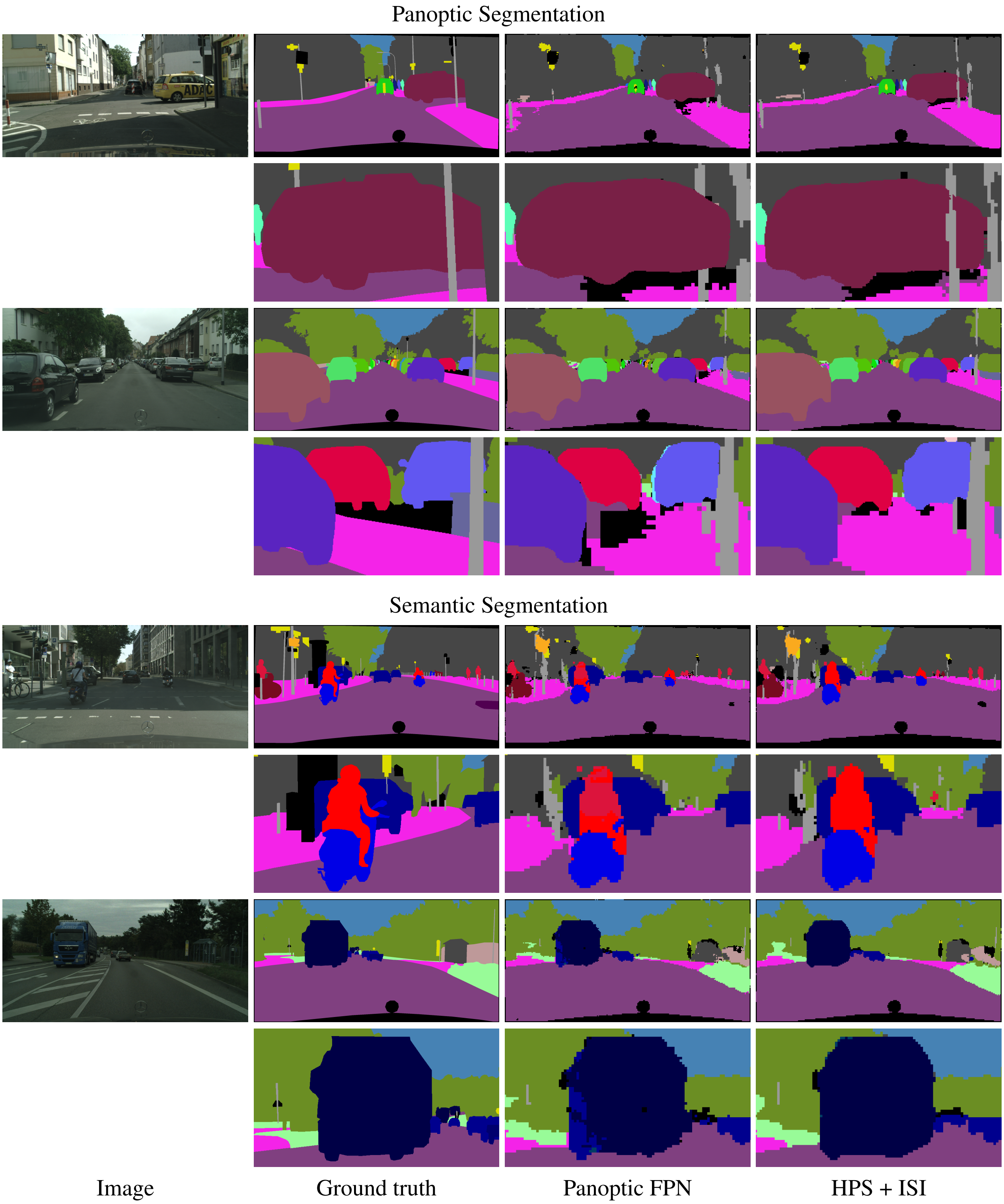

Figure 4: Qualitative results on the Cityscapes dataset. Compared to *Panoptic FPN*, *HPS + ISI* handles occlusions more accurately (1st row) and is less sensitive to speckle noise in semantically coherent regions (2nd row). Additionally, we predict more accurate semantic label transitions (3rd row) and reduce confusion between classes with similar semantic meaning like *rider* and *person* or *bus* and *car* (4th row). Both our end-to-end training as well as inter-task relations increase panoptic quality. **Best viewed in digital zoom.**

References

[1] COCO 2018 Panoptic Segmentation Task. http://cocodataset.org/index.htm#panoptic-leaderboard. Accessed: 2020-01-31.

[2] E. Brachmann, A. Krull, S. Nowozin, J. Shotton, F. Michel, S. Gumhold, and C. Rother. Dsac-differentiable ransac for camera localization. In *Conference on Computer Vision and Pattern Recognition*, pages 6684–6692, 2017.

[3] L.-C. Chen, G. Papandreou, F. Schroff, and H. Adam. Rethinking Atrous Convolution for Semantic Image Segmentation. *arXiv:1706.05587*, 2017.

[4] 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 *Conference on Computer Vision and Pattern Recognition*, pages 3213–3223, 2016.

[5] D. Feng, C. Haase-Schuetz, L. Rosenbaum, H. Hertlein, F. Duffhauss, C. Glaeser, W. Wiesbeck, and K. Dietmayer. Deep Multi-Modal Object Detection and Semantic Segmentation for Autonomous Driving: Datasets, Methods, and Challenges. *arXiv:1902.07830*, 2019.

[6] C.-Y. Fu, T. L. Berg, and A. C. Berg. IMP: Instance Mask Projection for High Accuracy Semantic Segmentation of Things. In *International Conference on Computer Vision*, pages 5178–5187, 2019.

[7] R. H. Hahnloser, R. Sarpeshkar, M. A. Mahowald, R. J. Douglas, and S. H. Seung. Digital Selection and Analogue Amplification Coexist in a Cortex-Inspired Silicon Circuit. *Nature*, 405(6789):947–951, 2000.

[8] K. He, G. Gkioxari, P. Dollár, and R. Girshick. Mask R-CNN. In *International Conference on Computer Vision*, pages 2961–2969, 2017.

[9] K. He, X. Zhang, S. Ren, and J. Sun. Deep Residual Learning for Image Recognition. In *Conference on Computer Vision and Pattern Recognition*, pages 770–778, 2016.

[10] S. Ioffe and C. Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. *arXiv:1502.03167*, 2015.

[11] A. Kirillov, R. Girshick, K. He, and P. Dollár. Panoptic Feature Pyramid Networks. In *Conference on Computer Vision and Pattern Recognition*, pages 6399–6408, 2019.

[12] A. Kirillov, K. He, R. Girshick, C. Rother, and P. Dollár. Panoptic Segmentation. In *Conference on Computer Vision and Pattern Recognition*, pages 9404–9413, 2019.

[13] A. Kirillov, E. Levinkov, B. Andres, B. Savchynskyy, and C. Rother. Instancecut: From Edges to Instances with Multicut. In *Conference on Computer Vision and Pattern Recognition*, pages 5008–5017, 2017.

[14] J. Li, A. Raventos, A. Bhargava, T. Tagawa, and A. Gaidon. Learning to Fuse Things and Stuff. *arXiv:1812.01192*, 2018.

[15] Y. Li, X. Chen, Z. Zhu, L. Xie, G. Huang, D. Du, and X. Wang. Attention-Guided Unified Network for Panoptic Segmentation. In *Conference on Computer Vision and Pattern Recognition*, pages 7026–7035, 2019.

[16] Y. Li, H. Qi, J. Dai, X. Ji, and Y. Wei. Fully Convolutional Instance-Aware Semantic Segmentation. In *Conference on Computer Vision and Pattern Recognition*, pages 2359–2367, 2017.

[17] X. Liang, L. Lin, Y. Wei, X. Shen, J. Yang, and S. Yan. Proposal-Free Network for Instance-Level Object Segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 40(12):2978–2991, 2017.

[18] T.-Y. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan, and S. Belongie. Feature Pyramid Networks for Object Detection. In *Conference on Computer Vision and Pattern Recognition*, pages 2117–2125, 2017.

[19] 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 *European Conference on Computer Vision*, pages 740–755, 2014.

[20] S. Liu, J. Jia, S. Fidler, and R. Urtasun. SGN: Sequential Grouping Networks for Instance Segmentation. In *International Conference on Computer Vision*, pages 3496–3504, 2017.

[21] S. Liu, L. Qi, H. Qin, J. Shi, and J. Jia. Path Aggregation Network for Instance Segmentation. In *Conference on Computer Vision and Pattern Recognition*, pages 8759–8768, 2018.

[22] J. Long, E. Shelhamer, and T. Darrell. Fully Convolutional Networks for Semantic Segmentation. In *Conference on Computer Vision and Pattern Recognition*, pages 3431–3440, 2015.

[23] A. Petrovai and S. Nedevschi. Multi-Task Network for Panoptic Segmentation in Automated Driving. In *Intelligent Transportation Systems Conference*, pages 2394–2401, 2019.

[24] O. Ronneberger, P. Fischer, and T. Brox. U-Net: Convolutional Networks for Biomedical Image Segmentation. In *Medical Image Computing and Computer-Assisted Intervention*, pages 234–241, 2015.

[25] J. Tighe, M. Niethammer, and S. Lazebnik. Scene Parsing with Object Instances and Occlusion Ordering. In *Conference on Computer Vision and Pattern Recognition*, pages 3748–3755, 2014.

[26] Z. Tu, X. Chen, A. L. Yuille, and S.-C. Zhu. Image Parsing: Unifying Segmentation, Detection, and Recognition. *International Journal of Computer Vision*, 63(2):113–140, 2005.

[27] J. Uhrig, M. Cordts, U. Franke, and T. Brox. Pixel-Level Encoding and Depth Layering for Instance-Level Semantic Labeling. In *German Conference on Pattern Recognition*, pages 14–25, 2016.

[28] Y. Xiong, R. Liao, H. Zhao, R. Hu, M. Bai, E. Yumer, and R. Urtasun. Upsnet: A Unified Panoptic Segmentation Network. In *Conference on Computer Vision and Pattern Recognition*, pages 8818–8826, 2019.

[29] Y. Yang, H. Li, X. Li, Q. Zhao, J. Wu, and Z. Lin. SOGNet: Scene Overlap Graph Network for Panoptic Segmentation. *arXiv:1911.07527*, 2019.

[30] D. Zhang, Y. Song, D. Liu, H. Jia, S. Liu, Y. Xia, H. Huang, and W. Cai. Panoptic Segmentation with an End-to-End Cell R-CNN for Pathology Image Analysis. In *Medical Image Computing and Computer-Assisted Intervention*, pages 237–244, 2018.