Pixel-level Encoding and Depth Layering for Instance-level Semantic Labeling

Jonas Uhrig[1,2], Marius Cordts[1,3], Uwe Franke[1], Thomas Brox[2]
[1]Daimler AG R&D, [2]University of Freiburg, [3]TU Darmstadt
jonas.uhrig@daimler.com

Abstract. Recent approaches for instance-aware semantic labeling have augmented convolutional neural networks (CNNs) with complex multi-task architectures or computationally expensive graphical models. We present a method that leverages a fully convolutional network (FCN) to predict semantic labels, depth and an instance-based encoding using each pixel's direction towards its corresponding instance center. Subsequently, we apply low-level computer vision techniques to generate state-of-the-art instance segmentation on the street scene datasets KITTI and Cityscapes. Our approach outperforms existing works by a large margin and can additionally predict absolute distances of individual instances from a monocular image as well as a pixel-level semantic labeling.

1 Introduction

The task of visual semantic scene understanding is mainly tackled from two opposing facets: pixel-level semantic labeling [4, 21, 22] and bounding-box object detection [11, 12, 23, 24]. The first assigns each pixel in an image with a semantic label segmenting the semantically connected regions in the scene. Such approaches work well with non-compact (*background*) classes such as buildings or ground, yet they do not distinguish individual object instances. Object detection aims to find all individual instances in the scene and describes them via bounding boxes. Therefore, the latter provides a rather coarse localization and is restricted to compact (*object*) classes such as cars or humans.

Recently, instance-level semantic labeling gained increasing interest [8, 19, 34, 35]. This task is at the intersection of both challenges. The aim is to combine the detection task with instance segmentation. Such a representation allows for a precise localization, which in turn enables better scene understanding. Especially in the domain of robotics and autonomous vehicles, instance-level semantic

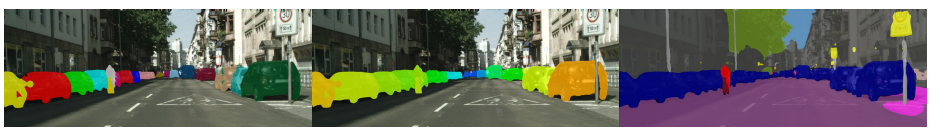


Fig. 1: Example scene representation as obtained by our method: instance segmentation, monocular depth estimation, and pixel-level semantic labeling.

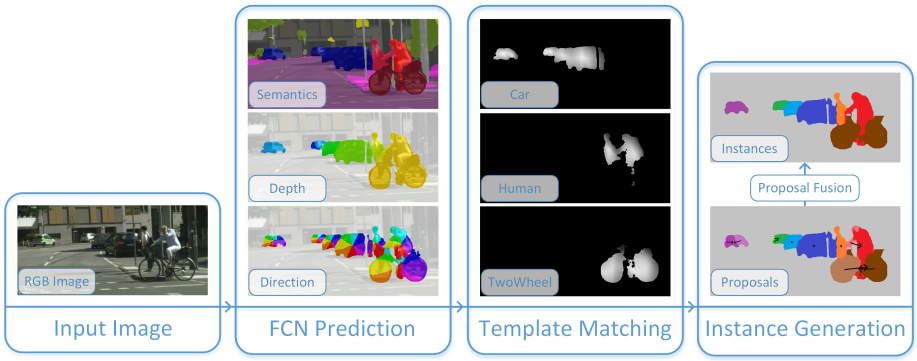


Fig. 2: From a single image, we predict 3 FCN outputs: semantics, depth, and instance center direction. Those are used to compute template matching score maps for semantic categories. Using these, we locate and generate instance proposals and fuse them to obtain our instance segmentation.

segmentation enables an explicit occlusion reasoning, precise object tracking and motion estimation, as well as behavior modeling and prediction.

Most state-of-the-art methods build upon a fully convolutional network (FCN) [21]. Recent approaches typically add post-processing, for example, based on conditional random fields (CRFs) [34, 35]. Other methods score region proposals for instance segmentation [7, 14] or object detection [11, 12, 23, 24], or use a multi-stage neural network for these tasks [8, 19].

In this work, we focus on street scene understanding and use a single monocular image to simultaneously obtain a holistic scene representation, consisting of a pixel-level semantic labeling, an instance-level segmentation of traffic participants, and a 3D depth estimation for each instance. We leverage an FCN that yields powerful pixel-level cues consisting of three output channels: a semantic class, the direction to the object center (where applicable) and the object distance (where applicable). Scene understanding is mainly due to the network and post-processing with standard computer vision methods is sufficient to obtain a detailed representation of an instance-aware semantic segmentation, *c.f.* Figs. 1 and 2. Our method significantly outperforms state-of-the-art methods on the street scene datasets KITTI [10] and Cityscapes [6].

2 Related Work

For the task of instance-level semantic labeling, there exist two major lines of research. The first leverages an over-complete set of object proposals that are either rejected, classified as an instance of a certain semantic class, and refined to obtain an instance segmentation. Common to all such methods is that the performance is depending on the quality of these proposals, since they cannot recover from missing instances in the proposal stage. Generally, such approaches

tend to be slow since all proposals must be classified individually. These properties cause inaccurate proposals to limit the performance of such methods [6, 16]. Our method belongs to the category of proposal-free methods, where the segmentation and the semantic class of object instances are inferred jointly.

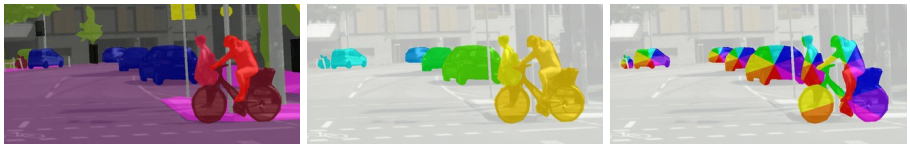
Proposal-based instance segmentation. Driven by the success of deep learning based object detectors such as R-CNN [12] or its variants [11, 24, 25], recent methods rely on these detections for instance segmentation. Either the underlying region proposals, such as MCG [2], are directly used as instance segments [6, 7, 14], or the bounding boxes are refined to obtain instance masks [5, 13]. Instead of bounding boxes, [18] uses a layered pictorial structure (LPS) model, where shape exemplars for object parts are mapped to the image in a probabilistic way. This yields an initial proposal for the object's pose and shape, which is refined using appearance cues. Using a bank of object detectors as proposals, [32] infers the instance masks via occlusion reasoning based on discrete depth layers. In [30], pixel-level semantic labels are used to score object candidates and vice versa in an alternating fashion, while also reasoning about occlusions and scene geometry. Based on proposals that form a segmentation tree, an energy function is constructed in [29] and its solution yields the instance segmentation.

Recently, [8] extended the R-CNN for instance segmentation with a multi-task network cascade. A fully convolutional network combined with three classification stages produces bounding-box proposals, refines these to segments, and ranks them to obtain the final instance-level labeling. They achieve excellent performance on PASCAL VOC [9] and MS COCO [20].

Proposal-free instance segmentation. Pixel-level semantic labeling based on neural networks has been very successful [4, 17, 21, 33, 36]. This triggered interest in casting also instance segmentation directly as a pixel labeling task. In [27], the network predicts for each pixel, whether it lies on an object boundary or not, however, requiring a rather delicate training. Using a long short-term memory (LSTM) network [15], instance segmentations can be sequentially sampled [26].

In [34, 35], instances are encoded via numbers that are further constrained to encode relative depth ordering in order to prevent arbitrary assignments. An FCN predicts these IDs at each pixel and a subsequent Markov Random Field (MRF) improves these predictions and enforces consistency. However, such a method is limited to scenes, where a clear depth ordering is present, *e.g.* a single row of parking cars, and the maximum number of instances is rather low.

The proposal-free network (PFN) [19] is a CNN that yields a pixel-level semantic labeling, the number of instances in the scene, and for each pixel the parameters of a corresponding instance bounding box. Based on these predictions, instances are obtained by clustering. The network has a fairly complex architecture with many interleaved building blocks, making training quite tricky. Further, the overall performance highly depends on the correct prediction of the number of instances in the scene. In street scenes, there can be hundreds of instances per image [6]. Thus, the number of training samples per number of instances is low, mistakes in their estimation can be critical, and the available cues for clustering might not correlate with the estimated number of instances.



(a) Semantic label. **(b)** Depth class. **(c)** Instance direction.

Fig. 3: Ground truth examples of our three proposed FCN channels. Color overlay (a) as suggested by [6], (b) represents depth per object from red (close) to blue (distant), (c) represents directions towards corresponding instance centers.

In this work, we focus on urban street scenes. Besides each pixel's semantic class, our network estimates an absolute depth, which is particularly useful for instance separation in street scenes. We encode instances on a pixel-level by the direction towards their center point. This representation is independent of the number of instances per image and provides strong signals at the instance boundaries.

3 Method

3.1 FCN Feature Representation

Our network extends the FCN-8s model [21] with three output channels that together facilitate instance segmentation. All channels are jointly trained as pixel-wise discrete labeling tasks using standard cross-entropy losses. Our proposed representation consists of (1) a semantic channel that drives the instance classification, (2) a depth channel to incorporate scale and support instance separation, and (3) a 2D geometric channel to facilitate instance detection and segmentation.

We chose the upscaling part of our FCN such that we can easily change the number of classes for each of the three proposed channels without re-initializing all upsampling layers. To this end, after the largest downsampling factor is reached, we use Deconvolution layers together with skip layers [21] to produce a representation of $\frac{1}{8}$ of the input resolution with a depth of 100 throughout all intermediate layers. The number of channels of this abstract representation is then reduced through 1×1 convolutions to the proposed semantic, depth, and instance center channels. To reach full input resolution, bilinear upsampling is applied, followed by a separate cross-entropy loss for each of our three output channels.

Semantics. To cope with different semantic classes, we predict a semantic label for each input pixel, *c.f.* Fig. 3a. These predictions are particularly important as they are the only source of semantic information in our approach. Further, the predicted semantic labels allow us to separate objects from background as well as objects of different classes from each other.

Depth. Urban street scenes typically contain objects at various distances [6]. To guide the post-processing in terms of objects at different scales, we predict a

depth label for each object pixel. We assign all pixels within an object instance to a constant depth value, e.g. the median over noisy measurements or the center of a 3D bounding box, *c.f.* Fig. 3b. These depth estimates also support instance separation, which becomes apparent when considering a row of parking cars, where the depth delta between neighboring cars is a full car length instead of a few centimeters in continuous space. The depth values are discretized into a set of classes so that close objects have a finer depth resolution than distant objects.

Direction. Object instances are defined by their boundary and class. Therefore, it seems natural to train an FCN model to directly predict boundary pixels. However, those boundaries represent a very delicate signal [1] as they have a width of only one pixel, and a single erroneously labeled pixel in the training data has a much higher impact compared to a region-based representation.

We introduce a class-based representation which implicitly combines information about an instance's boundary with the location of its visible center. For each object pixel we compute the direction towards its corresponding center and discretize this angle to a set of classes, *c.f.* Fig. 3c. This information is easier to grasp within a local region and is tailored for an FCN's capability to predict pixel-wise labels. Especially for pixels on the boundary between neighboring objects, our representation clearly separates the instances as predictions have nearly opposite directions. Since we predict the center of the visible area of an object and not its physical center, we can handle most types of occlusions very well. Furthermore, instance centers have a distinct pattern, *c.f.* Fig. 3c, which we exploit by applying template matching, as described in Sec. 3.2. Even though our proposed representation does not directly yield instance IDs, it is well defined even for an arbitrary number of instances per image.

To obtain an accurate direction estimation for each pixel, we assign the average direction by weighting all direction vectors with their respective FCN score (after softmax normalization). This allows us to recover a continuous direction estimation from the few discretized classes.

3.2 Template Matching

To extract instance centers, we propose template matching on the direction predictions, where templates are rectangular and contain the distinct pattern visible in Fig. 3c. We adjust the template's aspect ratio depending on its semantic class, so we can better distinguish between pedestrians and vehicles. In order to detect also distant objects with consistent matching scores, we scale the size of the templates depending on the predicted depth class.

To reduce induced errors from confusions between objects of similar semantic classes, we combine multiple semantic classes into the categories *human*, *car*, *large vehicle*, and *two wheeler*.

Normalized cross-correlation (NCC) is used to produce a score map for each category by correlating all pixels with their respective template. These maps indicate the likelihood of pixels being an instance center, *c.f.* Fig. 2. In the following, we predict instances for each category separately. After all instances are found, we assign them the majority semantic class label.

3.3 Instance Generation

Instance Centers. To determine instance locations, we iteratively find maxima in the generated template matching score maps via non-maximum suppression within an area that equals the template size. This helps avoid multiple detections of the same instance while incorporating typical object sizes. Those maxima represent our *temporary instance centers*, which are refined and merged in the following steps.

Instance Proposals. Each pixel with a predicted direction from the FCN is assigned to the closest temporary instance center where the relative location and predicted direction agree. Joining all assigned pixels per instance hypothesis yields a set of *instance proposals*.

Proposal Fusion. Elongated objects and erroneous depth predictions cause an over-segmentation of the instances. Thus, we refine the generated instances by accumulating estimated directions within each proposal. When interpreting direction predictions as vectors, they typically compensate each other within instance proposals that represent a complete instance, *i.e.* there are as many predictions pointing both left and right. However, incomplete instance proposals are biased to a certain direction. If there is a neighboring instance candidate with matching semantic class and depth in the direction of this bias, the two proposals are fused.

To the remaining instances we assign the average depth and the most frequent semantic class label within the region. Further, we merge our instance prediction with the pixel-level semantic labeling channel of the FCN by assigning the argmax semantic label to all non-instance pixels. Overall, we obtain a consistent scene representation, consisting of object instances paired with depth estimates and pixel-level labels for background classes.

4 Experiments

4.1 Datasets and Metrics

We evaluated our approach on the KITTI object detection dataset[10] extended with instance-level segmentations [3, 35] as well as Cityscapes [6]. Both datasets provide pixel-level annotations for semantic classes and instances, as well as depth information, which is essential for our approach. For the ground truth instance depths we used the centers of their 3D bounding box annotation in KITTI and the median disparity for each instance in Cityscapes based on the provided disparity maps. We used the official splits for training, validation and test sets.

We evaluated the segmentation based on the metrics proposed in [35] and [6]. To evaluate the depth prediction, we computed the mean absolute error (MAE), the root mean squared error (RMSE), the absolute relative difference (ARD), and the relative inlier ratios (δ_1, δ_2, δ_3) for thresholds $\delta_i = 1.25^i$ [31]. These metrics are computed on an instance level using the depths in meters. We only considered instances that overlap by more than 50 % with the ground truth.

Table 1: Evaluation of our variants on KITTI *val* (top) and comparison with baselines (*Best* [34]/[35]) on KITTI *test* (bottom) using metrics from [35]. For AvgFP and AvgFN lower is better, all other numbers are in percent and larger is better. *Mix* [35] shows the best results per metric from all baseline variants.

Method	Set	IoU	MWCov	MUCov	AvgPr	AvgRe	AvgFP	AvgFN	InsPr	InsRe	InsF1
Ours-D-F	val	79.4	41.5	43.4	**92.8**	54.4	0.042	1.33	16.6	29.8	21.4
Ours-F	val	**82.2**	35.9	35.7	83.6	**86.7**	0.158	**0.100**	31.4	69.5	43.3
Ours-D	val	79.6	**82.4**	**79.9**	89.9	54.6	**0.017**	1.33	**96.0**	42.8	59.2
Ours	val	**82.2**	80.7	76.3	83.7	**86.7**	0.100	**0.100**	91.8	**82.3**	**86.8**
Best [34]	test	77.4	67.0	49.8	82.0	61.3	0.479	0.840	48.9	43.8	46.2
Best [35]	test	77.0	69.7	51.8	83.9	57.5	0.375	1.139	65.3	50.0	56.6
Mix [35]	test	77.6	69.7	53.9	83.9	63.4	0.354	0.618	65.3	52.2	56.6
Ours	test	**84.1**	**79.7**	**75.8**	**85.6**	**82.0**	**0.201**	**0.159**	**86.3**	**74.1**	**79.7**

Fig. 4: Example results of our instance segmentation (right) and corresponding ground truth (middle) on KITTI. We even detect objects at very large distances.

4.2 Network Details

For Cityscapes, we used the 19 semantic classes and combined the 8 object classes into 4 categories (*car*, *human*, *two-wheeler*, and *large vehicle*). For KITTI, only *car* instance segmentations are available. For both datasets, we used 19 depth classes and an explicit class for background. We chose ranges for each depth class and template sizes differently for each dataset to account for different characteristics of present objects and used camera settings [6]. This is necessary as distances and semantic classes of objects differ remarkably. Details are provided in the supplementary material. The instance directions were split into 8 equal parts, each covering an angle of 45° for both datasets.

We use the 8-stride version of an FCN, which is initialized using the ImageNet dataset [28]. After initializing the upsampling layers randomly, we fine-tune the network on KITTI and Cityscapes to obtain all three output channels.

4.3 Ablation Studies

We evaluated the influence of each proposed component by leaving out one or more components from the complete processing pipeline (*Ours*). The performance was evaluated on the respective validation sets and is listed in Tables 1 and 2 (top) for both datasets.

Table 2: Evaluation on Cityscapes *val* (top) and *test* (center) using metrics in [6]. Further, we compare the performance for the most frequent label *car*, where we include KITTI *test* (bottom). All numbers are in percent and larger is better.

Variant	Dataset	Labels	AP	$AP^{50\%}$	AP^{100m}	AP^{50m}
Ours-D-F	CS val	all	2.4	5.7	3.6	4.9
Ours-F	CS val	all	7.0	17.5	11.1	12.8
Ours-D	CS val	all	6.8	15.8	10.9	14.2
Ours	CS val	all	**9.9**	**22.5**	**15.3**	**17.5**
MCG+R-CNN [6]	CS test	all	4.6	12.9	7.7	10.3
Ours	CS test	all	**8.9**	**21.1**	**15.3**	**16.7**
MCG+R-CNN [6]	CS test	car	10.5	26.0	17.5	21.2
Ours	CS test	car	22.5	37.8	36.4	40.7
Ours	KITTI test	car	41.6	69.1	49.3	49.3

For *Ours-D*, we removed the depth channel and chose the template size scale-agnostic. It turned out that a rather small template size, which leads to a large number of instance proposals, produces the best results. This is possible when post-processing heavily relies on correct direction predictions, which induces successful instance fusion. However, the performance is significantly worse in most metrics on both datasets compared to our full system, which shows that the depth information is an essential component of our approach. When the fusion component was also removed (*Ours-D-F*), a larger template size was needed to prevent an over-segmentation. However, performance dropped by an even larger margin than for *Ours-D*. In our last variant we kept the depth information but directly used the instance proposals as final instance predictions (*Ours-F*). The performance was even slightly worse than *Ours-D*, which shows that all our components are important to obtain accurate object instances. These observations are consistent on both datasets.

4.4 Instance Evaluation

KITTI. We clearly outperform all existing works on KITTI (*Best* [34]/[35]), *c.f.* Table 1 (bottom). Compared to the better performing work *Best* [35], we achieve a margin of 37 % relative improvement averaged over all metrics. Even when comparing our single variant with the best numbers over all existing variants for each metric individually (*Mix* [35]), we achieve a significantly better performance. We also evaluated our approach using the metrics introduced in [6] to enable comparisons in future publications, *c.f.* Table 2 (bottom). Qualitative results are shown in Fig. 4.

Cityscapes. On the Cityscapes dataset, our approach outperforms the baseline *MCG+R-CNN* [6] in all proposed metrics as evaluated by the dataset's submis-

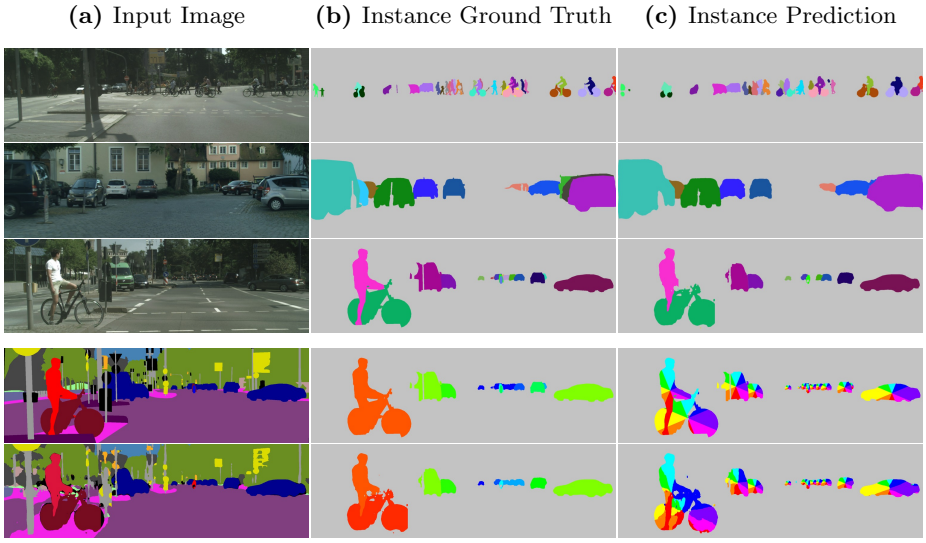


Fig. 5: Example results of our instance segmentation and corresponding ground truth (rows 1–3) on Cityscapes. We also include the three FCN output channels (row 5) and their ground truth (row 4). It can be seen that even distant objects are segmented well and the approach can handle occlusions.

sion server, *c.f.* Table 2 (center). We nearly double the performance in terms of the main score AP. Compared to the performance on KITTI, *c.f.* Table 2 (bottom), the numbers are significantly lower, indicating the higher complexity of scenes in Cityscapes. Qualitative results are shown in Fig. 5.

4.5 Depth Evaluation

As shown in Table 3, the average relative and mean absolute error of our predicted instances are as low as 7.7 % and 1.7 m, respectively, on the KITTI dataset. On the Cityscapes dataset, which contains much more complex scenes, with many and distant object instances, we achieve 11.3 % and 7.7 m, respectively. These results are particularly impressive, since we used only single monocular images as input for our network. We hope that future publications compare their depth estimation performance using the proposed metrics.

4.6 Evaluation of semantic class labels

Our method also yields a pixel-level semantic labeling including background classes that we evaluate on Cityscapes, *c.f.* Table 4. We compare to two baselines, *FCN 8s* [21] that uses the same FCN architecture as our approach and *Dilation10* [33], which is the currently best performing approach on Cityscapes [6].

Table 3: Instance-based depth evaluation on KITTI test and Cityscapes validation. MAE and RMSE are in meters, the others in percent. MAE, RMSE, and ARD denote error metrics, where smaller is better, δ_i represent accuracy, where higher is better.

Dataset	MAE	RMSE	ARD	δ_1	δ_2	δ_3
KITTI (test)	1.7	2.8	7.7	95.1	99.3	99.8
Cityscapes (val)	7.7	24.8	11.3	86.2	95.1	97.7

Table 4: Semantic pixel-level evaluation on Cityscapes test compared to baselines and using the corresponding metrics [6]. All values are in percent and larger is better.

Method	$\text{IoU}_{\text{class}}$	$\text{iIoU}_{\text{class}}$	$\text{IoU}_{\text{category}}$	$\text{iIoU}_{\text{category}}$
FCN 8s [6]	65.3	41.7	85.7	70.1
Dilation10 [33]	**67.1**	**42.0**	**86.5**	71.1
Ours	64.3	41.6	85.9	**73.9**

It can be seen that our approach is on par with the state-of-the-art although this work focuses on the harder instance segmentation task.

5 Conclusion

In this work, we present a fully convolutional network that predicts pixel-wise depth, semantics, and instance-level direction cues to reach an excellent level of holistic scene understanding. Instead of complex architectures or graphical models for post-processing, our approach performs well using only standard computer vision techniques applied to the network's three output channels. Our approach does not depend on region proposals and scales well for arbitrary numbers of object instances in an image.

We outperform existing works on the challenging urban street scene datasets Cityscapes [6] and KITTI [34, 35] by a large margin. On KITTI, our approach achieves 37 % relative improvement averaged over all metrics and we almost double the performance on Cityscapes. As our approach can reliably predict absolute depth values per instance, we provide an instance-based depth evaluation. Our depth predictions achieve a relative error of only a few meters, even though the datasets contain instances in more than one hundred meters distance. The main focus of this work is instance segmentation, but we also achieve state-of-the-art performance for pixel-level semantic labeling on Cityscapes, with a new best performance on an instance-based score over categories.

References

1. Arbelaez, P., Maire, M., Fowlkes, C., Malik, J.: Contour detection and hierarchical image segmentation. Trans. PAMI 33(5) (2011) 5
2. Arbelez, P., Pont-Tuset, J., Barron, J., Marques, F., Malik, J.: Multiscale combinatorial grouping. In: CVPR (2014) 3
3. Chen, L.C., Fidler, S., Urtasun, R.: Beat the MTurkers: Automatic image labeling from weak 3d supervision. In: CVPR (2014) 6
4. Chen, L., Papandreou, G., Kokkinos, I., Murphy, K., Yuille, A.L.: Semantic image segmentation with deep convolutional nets and fully connected CRFs (2015) 1, 3, iii, v
5. Chen, Y.T., Liu, X., Yang, M.H.: Multi-instance object segmentation with occlusion handling. In: CVPR (2015) 3
6. 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: CVPR (2016) 2, 3, 4, 6, 7, 8, 9, 10, i, ii, iii, iv, v
7. Dai, J., He, K., Sun, J.: Convolutional feature masking for joint object and stuff segmentation. In: CVPR (2015) 2, 3
8. Dai, J., He, K., Sun, J.: Instance-aware semantic segmentation via multi-task network cascades. In: CVPR (2016) 1, 2, 3
9. Everingham, M., Gool, L., Williams, C.K.I., Winn, J., Zisserman, A.: The pascal visual object classes (VOC) challenge. IJCV 88(2) (2009) 3
10. Geiger, A., Lenz, P., Urtasun, R.: Are we ready for autonomous driving? The KITTI vision benchmark suite. In: CVPR (2012) 2, 6, i, iii
11. Girshick, R.: Fast R-CNN. In: ICCV (2015) 1, 2, 3
12. Girshick, R., Donahue, J., Darrell, T., Malik, J.: Rich feature hierarchies for accurate object detection and semantic segmentation. In: CVPR (2014) 1, 2, 3
13. Hariharan, B., Arbelez, P., Girshick, R., Malik, J.: Hypercolumns for object segmentation and fine-grained localization. In: CVPR (2015) 3
14. Hariharan, B., Arbeláez, P., Girshick, R., Malik, J.: Simultaneous detection and segmentation. In: ECCV (2014) 2, 3
15. Hochreiter, S., Schmidhuber, J.: Long short-term memory. Neural Comput. 9(8) (1997) 3
16. Hosang, J., Benenson, R., Dollr, P., Schiele, B.: What makes for effective detection proposals? Trans. PAMI 38(4) (2016) 3
17. Kirillov, A., Schlesinger, D., Forkel, W., Zelenin, A., Zheng, S., Torr, P., Rother, C.: Efficient likelihood learning of a generic CNN-CRF model for semantic segmentation. In: arXiv:1511.05067v2 [cs.CV] (2015) 3
18. Kumar, M.P., Ton, P.H.S., Zisserman, A.: OBJ CUT. In: CVPR (2005) 3
19. Liang, X., Wei, Y., Shen, X., Yang, J., Lin, L., Yan, S.: Proposal-free network for instance-level object segmentation. In: arXiv:1509.02636v2 [cs.CV] (2015) 1, 2, 3
20. Lin, T.Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., Zitnick, C.L.: Microsoft COCO: Common objects in context. In: ECCV (2014) 3
21. Long, J., Shelhamer, E., Darrell, T.: Fully convolutional networks for semantic segmentation. In: CVPR (2015) 1, 2, 3, 4, 9
22. Papandreou, G., Chen, L., Murphy, K., Yuille, A.L.: Weakly- and semi-supervised learning of a DCNN for semantic image segmentation. In: ICCV (2015) 1, iii, v
23. Redmon, J., Divvala, S.K., Girshick, R.B., Farhadi, A.: You only look once: Unified, real-time object detection. In: CVPR (2016) 1, 2

24. Ren, S., He, K., Girshick, R., Sun, J.: Faster R-CNN: Towards real-time object detection with region proposal networks. In: NIPS (2015) 1, 2, 3
25. Ren, S., He, K., Girshick, R.B., Zhang, X., Sun, J.: Object detection networks on convolutional feature maps. In: arXiv:1504.06066v1 [cs.CV] (2015) 3
26. Romera-Paredes, B., Torr, P.: Recurrent instance segmentation. In: arXiv:1511.08250v2 [cs.CV] (2015) 3
27. Ronneberger, O., Fischer, P., Brox, T.: U-net: Convolutional networks for biomedical image segmentation. In: MICCAI (2015) 3
28. Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., Bernstein, M., Berg, A., Fei-Fei, L.: ImageNet Large Scale Visual Recognition Challenge. IJCV 115(3), 211–252 (2015) 7
29. Silberman, N., Sontag, D., Fergus, R.: Instance segmentation of indoor scenes using a coverage loss. In: ECCV (2014) 3
30. Tighe, J., Niethammer, M., Lazebnik, S.: Scene parsing with object instances and occlusion ordering. In: CVPR (2014) 3
31. Wang, P., Shen, X., Lin, Z., Cohen, S., Price, B., Yuille, A.: Towards unified depth and semantic prediction from a single image. In: CVPR (2015) 6
32. Yang, Y., Hallman, S., Ramanan, D., Fowlkes, C.: Layered object models for image segmentation. Trans. PAMI 34(9) (2012) 3
33. Yu, F., Koltun, V.: Multi-scale context aggregation by dilated convolutions. In: ICLR (2016) 3, 9, 10, iii, v
34. Zhang, Z., Schwing, A.G., Fidler, S., Urtasun, R.: Monocular object instance segmentation and depth ordering with cnns. In: ICCV (2015) 1, 2, 3, 7, 8, 10
35. Zhang, Z., Fidler, S., Urtasun, R.: Instance-level segmentation with deep densely connected MRFs. In: CVPR (2016) 1, 2, 3, 6, 7, 8, 10
36. Zheng, S., Jayasumana, S., Romera-Paredes, B., Vineet, V., Su, Z., Du, D., Huang, C., Torr, P.: Conditional random fields as recurrent neural networks. In: ICCV (2015) 3, iii, v

Supplementary Material for Pixel-level Encoding and Depth Layering for Instance-level Semantic Labeling

Jonas Uhrig[1,2], Marius Cordts[1,3], Uwe Franke[1], Thomas Brox[2]

[1]Daimler AG R&D, [2]University of Freiburg, [3]TU Darmstadt
jonas.uhrig@daimler.com

A Qualitative Results

Figures 1 and 2 show further qualitative examples of our instance segmentation on urban scenes from KITTI [10] and Cityscapes [6]. It can be seen that our approach can segment even high numbers of instances despite heavy occlusions and clutter.

B Depth Ranges

As mentioned in Sec. 3.1, we discretized continuous instance depths into 19 depth classes. Instead of equidistantly splitting them, we chose the ranges for each class such that the sizes of objects within each depth class are similar. We found this option to yield slightly better results, since the subsequent template matching is based on our FCN's depth prediction and equal object sizes per depth class result in more reliable template matching scores.

We defined the values as in Table 2 to provide a good trade-off between number of depth classes and depth resolution, as well as number of samples per depth class in the training data. As the Cityscapes dataset contains a lot of object instances labeled for very high distances of over 200 meters [6], the depth ranges had to be chosen differently than for KITTI [10].

C Class-level Evaluation

C.1 Instance-level Evaluation

We list class-level performances of our approach for instance-level semantic labeling (*Ours*) and the baseline *MCG+R-CNN* [6] in Table 1. Our approach has difficulties especially for semantic classes that are least reliably classified by our FCN, such as bus, truck, and train *c.f.* Tables 3 to 5. Best results are achieved for cars and humans, while we outperform the proposal-based baseline for all other classes by large margins in all used metrics.

C.2 Pixel-level Evaluation

A detailed evaluation of our performance for pixel-level semantic labeling can be found in Tables 3 to 5. Even though our main focus lies on instance-level semantic labeling, we achieve competitive results for all classes compared to the baselines listed in [6]. Using the instance-aware metric iIoU, we even outperform most existing works by a few percent points for the object classes *person*, *car*, and *bicycle*.

The reason for a comparably low performance on the classes *bus*, *truck*, and *train* becomes evident by inspecting Tables 3 and 4. We achieve comparably low semantic labeling results on a pixel-level for these classes and therefore our template matching and instance generation steps perform significantly worse than on all other object classes.

References

36. Badrinarayanan, V., Kendall, A., Cipolla, R.: Segnet: A deep convolutional encoder-decoder architecture for image segmentation (2015) iii, v
37. Lin, G., Shen, C., Reid, I.D., van den Hengel, A.: Efficient piecewise training of deep structured models for semantic segmentation (2015) iii, v
38. Liu, Z., Li, X., Luo, P., Loy, C.C., Tang, X.: Semantic image segmentation via deep parsing network. In: ICCV. pp. 1377–1385 (2015) iii, v

Table 1: Class-based evaluation of existing works and our approach for instance-level segmentation on Cityscapes *test* using metrics proposed in [6]. All numbers are in percent and larger is better.

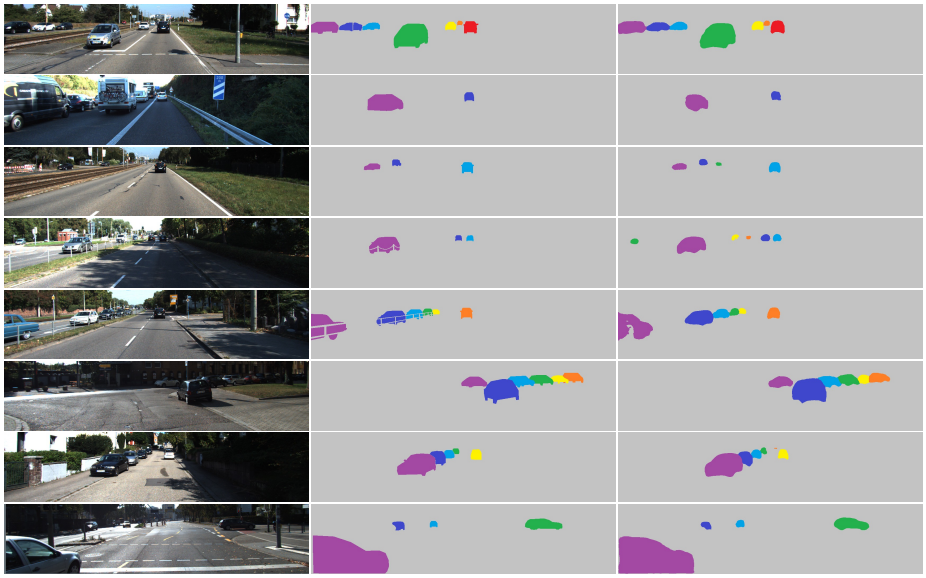
	Metric	person	rider	car	truck	bus	train	motorcycle	bicycle	**mean score**
MCG+R-CNN [6]	AP	1.3	0.6	10.5	**6.1**	**9.7**	**5.9**	1.7	0.5	4.6
Ours	AP	**12.5**	**11.7**	**22.5**	3.3	5.9	3.2	**6.9**	**5.1**	**8.9**
MCG+R-CNN [6]	$AP^{50\%}$	5.6	3.9	26.0	**13.8**	**26.3**	**15.8**	8.6	3.1	12.9
Ours	$AP^{50\%}$	**31.8**	**33.8**	**37.8**	7.6	12.0	8.5	**20.5**	**17.2**	**21.1**
MCG+R-CNN [6]	AP^{100m}	2.6	1.1	17.5	**10.6**	**17.4**	**9.2**	2.6	0.9	7.7
Ours	AP^{100m}	**24.4**	**20.3**	**36.4**	5.5	10.6	5.2	**10.5**	**9.2**	**15.3**
MCG+R-CNN [6]	AP^{50m}	2.7	1.1	21.2	**14.0**	**25.2**	**14.2**	2.7	1.0	10.3
Ours	AP^{50m}	**25.0**	**21.0**	**40.7**	6.7	13.5	6.4	**11.2**	**9.3**	**16.7**

Table 2: Assignment of depth classes with corresponding depth ranges for the two used datasets KITTI [10] and Cityscapes [6].

Class	Depth Ranges KITTI	Depth Stepsize [m]	Depth Ranges Cityscapes	Depth Stepsize [m]
1	0-2 m,	2	0-6 m,	6
2	2-3.5 m,	1.5	6-8 m,	2
3	3.5-5 m,	1.5	8-10 m,	2
4	5-6 m,	1	10-12 m,	2
5	6-7 m,	1	12-14 m,	2
6	7-8.5 m,	1.5	14-17 m,	3
7	8.5-10 m,	1.5	17-20 m,	3
8	10-12 m,	2	20-23 m,	3
9	12-14 m,	2	23-27 m,	4
10	14-17 m,	3	27-31 m,	4
11	17-20 m,	3	31-36 m,	5
12	20-24 m,	4	36-41 m,	5
13	24-29 m,	5	41-47 m,	6
14	29-35 m,	6	47-54 m,	7
15	35-43 m,	8	54-63 m,	9
16	43-52 m,	9	63-73 m,	10
17	52-63 m,	11	73-86 m,	13
18	63-76 m,	13	86-100 m,	14
19	76-∞ m		100-∞ m	

Table 3: Evaluation of our class-level performance for pixel-level semantic labeling on Cityscapes *test* using the IoU metric proposed in [6]. All numbers are in percent and larger is better.

	road	sidewalk	building	wall	fence	pole	traffic light	traffic sign	vegetation	terrain	sky	person	rider	car	truck	bus	train	motorcycle	bicycle	**mean IoU**
[36] ext.	95.6	70.1	82.8	29.9	31.9	38.1	43.1	44.6	87.3	62.3	91.7	67.3	50.7	87.9	21.7	29.0	34.7	40.5	56.6	56.1
[36] basic	96.4	73.2	84.0	28.5	29.0	35.7	39.8	45.2	87.0	63.8	91.8	62.8	42.8	89.3	38.1	43.1	44.2	35.8	51.9	57.0
[38]	96.3	71.7	86.7	43.7	31.7	29.2	35.8	47.4	88.4	63.1	**93.9**	64.7	38.7	88.8	48.0	56.4	49.4	38.3	50.0	59.1
[36]	96.3	73.9	88.2	**47.6**	41.3	35.2	49.5	59.7	90.6	66.1	93.5	70.4	34.7	90.1	39.2	57.5	55.4	43.9	54.6	62.5
[4]	97.3	77.7	87.7	43.6	40.5	29.7	44.5	55.4	89.4	67.0	92.7	71.2	49.4	91.4	48.7	56.7	49.1	47.9	58.6	63.1
[22]	97.4	78.3	88.1	47.5	44.2	29.5	44.4	55.4	89.4	67.3	92.8	71.0	49.3	91.4	**55.9**	**66.6**	**56.7**	48.1	58.1	64.8
[37]	97.3	78.5	88.4	44.5	**48.3**	34.1	55.5	61.7	90.1	**69.5**	92.2	72.5	52.3	91.0	54.6	61.6	51.6	**55.0**	63.1	66.4
[6]	97.4	78.4	89.2	34.9	44.2	47.4	**60.1**	65.0	91.4	69.3	**93.9**	77.1	51.4	92.6	35.3	48.6	46.5	51.6	**66.8**	65.3
[33]	**97.6**	**79.2**	**89.9**	37.3	47.6	**53.2**	58.6	**65.2**	**91.8**	69.4	93.7	**78.9**	**55.0**	**93.3**	45.5	53.4	47.7	52.2	66.0	**67.1**
Ours	97.4	77.7	88.8	27.7	40.1	51.5	**60.1**	64.7	91.1	67.6	93.5	77.7	54.2	92.4	33.7	42.0	42.5	52.5	66.5	64.3

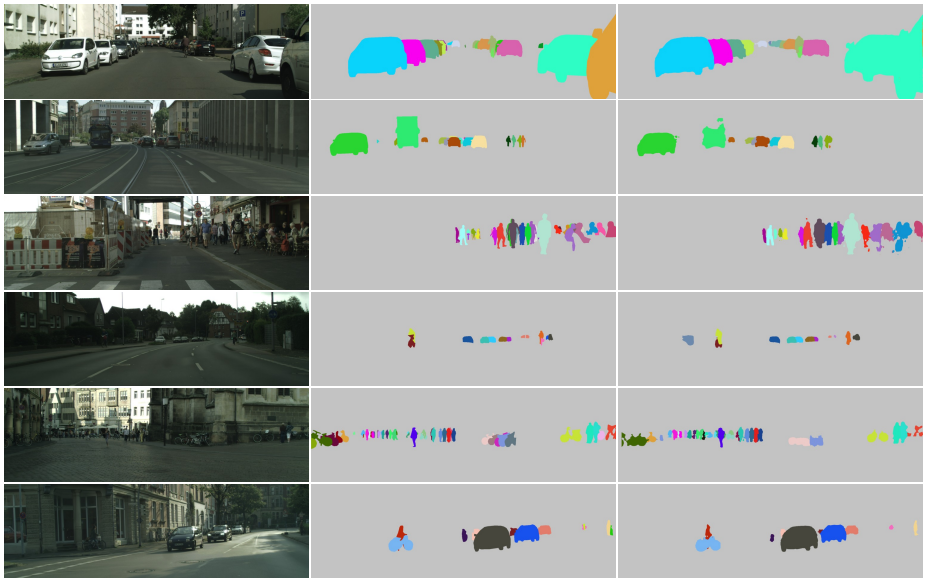


(a) Input Image **(b)** Instance Ground Truth **(c)** Instance Prediction

Fig. 1: Further example results of our instance segmentation (right) and corresponding ground truth (middle) on KITTI.

Table 4: Confusion matrix of our method's performance for pixel-level semantic labeling on Cityscapes *validation* using all 8 object classes [6]. All numbers are in percent.

	person	rider	car	truck	bus	train	motorcycle	bicycle	**prior**
person	91	1	1	0	0	0	0	1	1.14
rider	24	61	2	0	0	0	2	7	0.19
car	0	0	97	0	0	0	0	0	5.70
truck	0	0	25	56	1	1	0	0	0.26
bus	0	0	14	4	67	2	0	0	0.34
train	0	0	5	0	16	42	0	0	0.10
motorcycle	6	4	13	0	0	0	64	7	0.07
bicycle	3	3	2	0	0	0	1	85	0.62



(a) Input Image **(b)** Instance Ground Truth **(c)** Instance Prediction

Fig. 2: Further example results of our instance segmentation (right) and corresponding ground truth (center) on Cityscapes *validation*.

Table 5: Class-level evaluation of our object-related performance for semantic segmentation on Cityscapes *test* using the iIoU metric proposed in [6]. All numbers are in percent and larger is better.

	person	rider	car	truck	bus	train	motorcycle	bicycle	**mean iIoU**
[36] ext.	49.9	27.1	81.1	15.3	23.7	18.5	19.6	38.4	34.2
[36] basic	44.3	22.7	78.4	16.1	24.3	20.7	15.8	33.6	32.0
[38]	38.9	12.8	78.6	13.4	24.0	19.2	10.7	27.2	28.1
[36]	50.6	17.8	81.1	18.0	25.0	30.3	22.3	30.1	34.4
[4]	40.5	23.3	78.8	20.3	31.9	24.8	21.1	35.2	34.5
[22]	40.7	23.1	78.6	21.4	32.4	27.6	20.8	34.6	34.9
[37]	56.2	**38.0**	77.1	**34.0**	**47.0**	**33.4**	**38.1**	49.9	**46.7**
[6]	55.9	33.4	83.9	22.2	30.8	26.7	31.1	49.6	41.7
[33]	56.3	34.5	85.8	21.8	32.7	27.6	28.0	49.1	42.0
Ours	**60.6**	33.4	**86.7**	19.5	25.6	25.8	30.5	**50.5**	41.6