

MGNet: Monocular Geometric Scene Understanding for Autonomous Driving

Markus Schön Michael Buchholz Klaus Dietmayer

Institute of Measurement, Control and Microtechnology, Ulm University

{markus.schoen,michael.buchholz,klaus.dietmayer}@uni-ulm.de

Abstract

We introduce MGNet, a multi-task framework for monocular geometric scene understanding. We define monocular geometric scene understanding as the combination of two known tasks: Panoptic segmentation and self-supervised monocular depth estimation. Panoptic segmentation captures the full scene not only semantically, but also on an instance basis. Self-supervised monocular depth estimation uses geometric constraints derived from the camera measurement model in order to measure depth from monocular video sequences only. To the best of our knowledge, we are the first to propose the combination of these two tasks in one single model. Our model is designed with focus on low latency to provide fast inference in real-time on a single consumer-grade GPU. During deployment, our model produces dense 3D point clouds with instance aware semantic labels from single high-resolution camera images. We evaluate our model on two popular autonomous driving benchmarks, i.e., Cityscapes and KITTI, and show competitive performance among other real-time capable methods. Source code is available at <https://github.com/markusschoen/MGNet>.

1. Introduction

Scene understanding is an essential component for autonomous driving perception systems, since it provides necessary information for higher-level functions such as multi-object-tracking or behavior planning. Recent advances in semantic segmentation [5, 42, 61, 67, 81, 85] and instance segmentation [22, 25, 39, 69, 70] based on deep neural networks show outstanding results, while fast models [9, 13, 26, 51, 78] focus on optimizing the speed-accuracy trade-off for usage in latency-critical applications, e.g., autonomous driving. New tasks emerge regularly, bringing perception systems a step forward towards full scene understanding, but also increasing task complexity.

One of these tasks is panoptic segmentation [32], a combination of semantic and instance segmentation. Semantic segmentation focuses on stuff classes, i.e., amorph regions

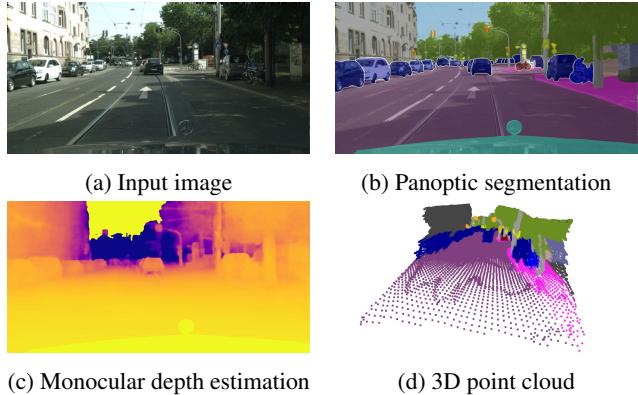


Figure 1: Example prediction of our model, (a) the input image which is fed to the network, (b) the panoptic prediction on top of the input image, (c) the monocular depth estimation and (d) the final 3D point cloud generated by the network in real-time.

like sky or road, while instance segmentation focuses on thing classes, i.e., countable objects like cars, bicycles or pedestrians. Panoptic segmentation handles both stuff and thing classes, providing not only unique class label for each pixel in the image, but also instance IDs for countable objects. Panoptic segmentation is an important step towards scene understanding in autonomous vehicles, since it provides not only the object masks, but also interesting amorph regions like drivable road space or sidewalks. However, the vast majority [3, 7, 12, 31, 36, 38, 40, 46, 52–54, 58, 65, 66, 72, 73, 76] of panoptic segmentation methods focus on high quality rather than inference speed, making them unsuitable for on-board integration into an autonomous vehicle perception system. Only few methods [9, 13, 26, 51] have been proposed in the low-latency regime. Additionally, the camera-based measurement in the two-dimensional image plane limits the capabilities of panoptic segmentation models in autonomous vehicles. The 2D pixel location is not sufficient for higher-level systems such as behavior planning to reason about the current environment. Rather, a 3D representation is required.

Monocular depth estimation [11] tackles this problem by predicting per-pixel depth estimates from single camera images. Using this depth information, pixels from the image can be projected into 3D space. Camera-based depth estimation is an ill-posed problem due to the camera measurement model, making it a very challenging task to solve. Furthermore, accurate depth annotations are hard to acquire. Stereo cameras can provide depth information, but require an accurate calibration between the two cameras. Additionally, resulting depth images are noisy, inaccurate at farther distances, and have a lot of missing regions where the stereo matching failed. In contrast, lidar sensors provide distance measurements with high accuracy. The measurements can be projected into the image plane and used as a label, but this again requires an accurate calibration and synchronization between the sensors. The generated depth images are much more accurate, but very sparse. Therefore, recent methods follow a self-supervised training strategy using either stereo images [14, 16], video sequences [2, 19, 20, 45, 64, 75, 82–84, 86], or both [17, 37, 43] during training. The training objective is formulated as an image synthesis problem based on geometric constraints.

Combining multiple tasks can be resource demanding. The naive approach of training a separate network for each task can reach hardware limitations quickly in low resource environments. Hence, multi-task learning [1] emerged to combine multiple tasks in a single network to reduce latency and memory requirements on the target hardware. While joint task training can potentially boost single-task performance, it brings its own difficulties, *e.g.*, loss balancing and conflicting gradients.

In this work, we introduce the task of monocular geometric scene understanding, a combination of panoptic segmentation and self-supervised monocular depth estimation. We propose a multi-task framework, which we call MGNet, to tackle this new task with focus on latency. MGNet combines the ideas of state-of-the-art methods Panoptic Deeplab [7] and Monodepth2 [17] with a lightweight network architecture. The self-supervised monocular depth estimation formulation requires only video sequences during training, releasing us from the need of hard to acquire ground truth data. Hence, our model can be trained using data from a single camera. We propose an improved version of the Dense Geometrical Constrains Module (DGC), introduced in [75], using our panoptic prediction for scale-aware depth estimation. Similar to [3], we generate pseudo labels for video sequence frames, which reduces the number of annotated frames necessary for panoptic segmentation. The multi-task setting implicitly constrains the model to learn a unified representation for both tasks and reduces the overall latency. We use homoscedastic uncertainty weighting, introduced in [29], but adopt a novel weighting scheme, combining fixed and learnable task weights, to improve multi-

task performance. During deployment, our model produces 3D point clouds with instance aware class labels using a single camera image as input. Figure 1 shows an example prediction of our model. We evaluate our method on Cityscapes [8] and KITTI [15] and are able to outperform previous approaches in terms of latency, while maintaining competitive accuracy. Specifically, on Cityscapes, we achieve 55.7 PQ and 8.3 RMSE with 30 FPS on full resolution 1024×2048 pixel images. On KITTI, we achieve 3.761 RMSE with 82 FPS on 384×1280 pixel images.

2. Related Work

Since, to the best of our knowledge, no combined algorithm exists yet, we discuss the related work separately for the fields of panoptic segmentation and self-supervised depth estimation. Thereafter, we discuss related work in the field of multi-task learning for scene understanding.

2.1. Panoptic Segmentation

Panoptic segmentation [32] was introduced to unify the tasks of semantic and instance segmentation. Top-down approaches [6, 31, 36, 40, 46, 52–54, 58, 72, 73] first generate proposals, usually in form of bounding boxes, which are then used for instance mask prediction. Most methods employ a Mask R-CNN [22] head combined with a separate branch for semantic segmentation and a merging module to handle conflicts [36, 40, 46, 53]. For example, Mohan and Valada [46] propose a new variant of Mask R-CNN along with a new fusion module to achieve state-of-the-art results. In contrast, bottom-up approaches [3, 4, 7, 12, 66, 76] are proposal-free. For example, Panoptic DeepLab [7] represents instance masks as pixel offsets and center keypoints. An efficient merging module is employed to group the logits to final instance masks. Previous works [3, 4, 55, 66] build on this simple and strong framework proving its flexibility. Our work also uses the Panoptic DeepLab framework as a basis. Recently, the first methods for end-to-end panoptic segmentation emerged [38, 65]. In contrast to previous works, these approaches predict panoptic segmentation maps directly using a unified representation for stuff and thing classes. For example, Wang *et al.* [65] introduced MaX-DeepLab, a novel architecture which extends Axial-DeepLab [66] with a mask transformer head. However, their method is computationally demanding, making it unsuitable for real-time applications. Only a few works exist that have a focus on real-time panoptic segmentation [9, 13, 26, 51]. Hou *et al.* [26] set state-of-the-art for real-time panoptic segmentation by proposing a novel panoptic segmentation network with efficient data flow, which can reach up to 30 FPS on full resolution images of Cityscapes. Our model has similar performance while solving the additional task of self-supervised monocular depth estimation.

2.2. Self-Supervised Monocular Depth Estimation

Zhou *et al.* [84] were the first to propose a training scheme for depth estimation that only uses monocular video sequences for supervision. The idea is to synthesize the current frame from adjacent frames in the video sequence by using the predicted depth estimation and predicted relative pose between the frames. The synthesized frames are then compared with the current frame using a photometric loss. During training, multiple depth maps at different scales are used to mitigate the effect of learning from low texture regions. Assumptions for self-supervised depth learning are a moving camera in a static environment and no occlusions between the frames. Furthermore, due to the ambiguous nature of photometric loss, the depth can only be predicted up to an unknown scale factor. Since then, a number of works [2, 17, 19, 20, 37, 43, 45, 64, 75, 82, 83, 86] advanced the field considerably. For example, Godard *et al.* [17] propose to upsample the multi-scale depth maps before loss calculation and use the minimum photometric error to tackle occlusions. Furthermore, they propose an auto-masking strategy to avoid holes of infinite depth in low-texture regions and in regions with dynamic objects. Xue *et al.* [75] build on top of [17] and propose a DGC to estimate the scale factor based on geometric constraints. We incorporate the basic idea of [75] in our framework, but introduce an improved version of the DGC which leverages our panoptic prediction. Some works focus on incorporating semantic [21, 34] or panoptic [57] label maps to explicitly boost the performance of self-supervised depth maps. In contrast, our method uses multi-task optimization to implicitly boost single-task performance.

2.3. Multi-Task Learning

Multi-Task learning [1] has emerged in order to save computational resources by combining multiple tasks in a single network. Furthermore, learning multiple tasks at once can improve generalization ability and lead to better results compared to single-task performance. A number of works exists, which tackle the different tasks for scene understanding in a multi-task setting [18, 28, 29, 48, 55, 63, 68, 74, 79, 80]. Goel *et al.* [18] propose QuadroNet, a real-time capable model to predict 2D bounding boxes, panoptic segmentation, and depth from single images. While similar to our work with respect to prediction, they train depth in a fully supervised way, requiring disparity or lidar ground truth during training. Some works incorporate semantic segmentation and self-supervised depth estimation in multi-task frameworks [33, 35]. Klingner *et al.* [33] show that joint training of self-supervised depth and supervised semantic segmentation can boost performance and increase noise robustness of the model. Our model does not combine semantic segmentation, but rather panoptic segmentation with self-supervised depth estimation.

3. Method

In this section, we describe our MGNet framework to jointly tackle the tasks of panoptic segmentation and self-supervised monocular depth estimation in a single efficient model. Figure 2 gives an overview of our framework. The following sections explain the different parts of our framework in detail.

3.1. MGNet Network Architecture

Our MGNet architecture is designed as an encoder-decoder structure with a single shared encoder and three task-specific decoders.

Encoder: The encoder consists of a feature extraction backbone, which extracts image level features at different scales up to an output stride of 32. The backbone can easily be switched, hence we compare different lightweight backbones in Section 4 with respect to overall performance and latency. Additionally, we add a Global Context Module (GCM) on top of the last feature map of the backbone. The GCM uses global average pooling to extract fine-grained features from the input image.

Task-specific decoders: The task-specific decoders all share the same structure. For panoptic segmentation, we use two decoders as proposed in [7], one for semantic segmentation and one for instance segmentation. A single decoder is used for the task of self-supervised depth estimation. We leverage Attention Refinement Modules (ARMs) and Feature Fusion Modules (FFMs) from [78] to combine the features from different scales in an efficient and effective way. Similar to [78], we use two ARMs on the last two feature maps of the backbone and add them together with the fine-grained features generated by the global context module. In contrast to [78], we do not follow the bilateral feature encoding using a spatial path for low-level and a context path for high-level features. We found that the bilateral approach does not improve performance, but increases latency of the overall model. Instead, we use a skip connection to propagate low-level features from the backbone to the decoders and combine them with the high-level features in the FFM.

Head modules: We add simple head modules to each decoder to generate the logits. All heads share the same structure, using a 3×3 conv followed by a 1×1 conv to map the feature channel size to the final logits channel size, *e.g.*, the number of semantic classes used for semantic segmentation. Similar to [7], we encode instances by their center of mass and pixel-wise 2D offset vectors pointing towards the center. We use a separate head for instance center and offset regression, but share the decoder for the instance task. The logits produced by the depth head are passed through a sigmoid activation as used in [17].

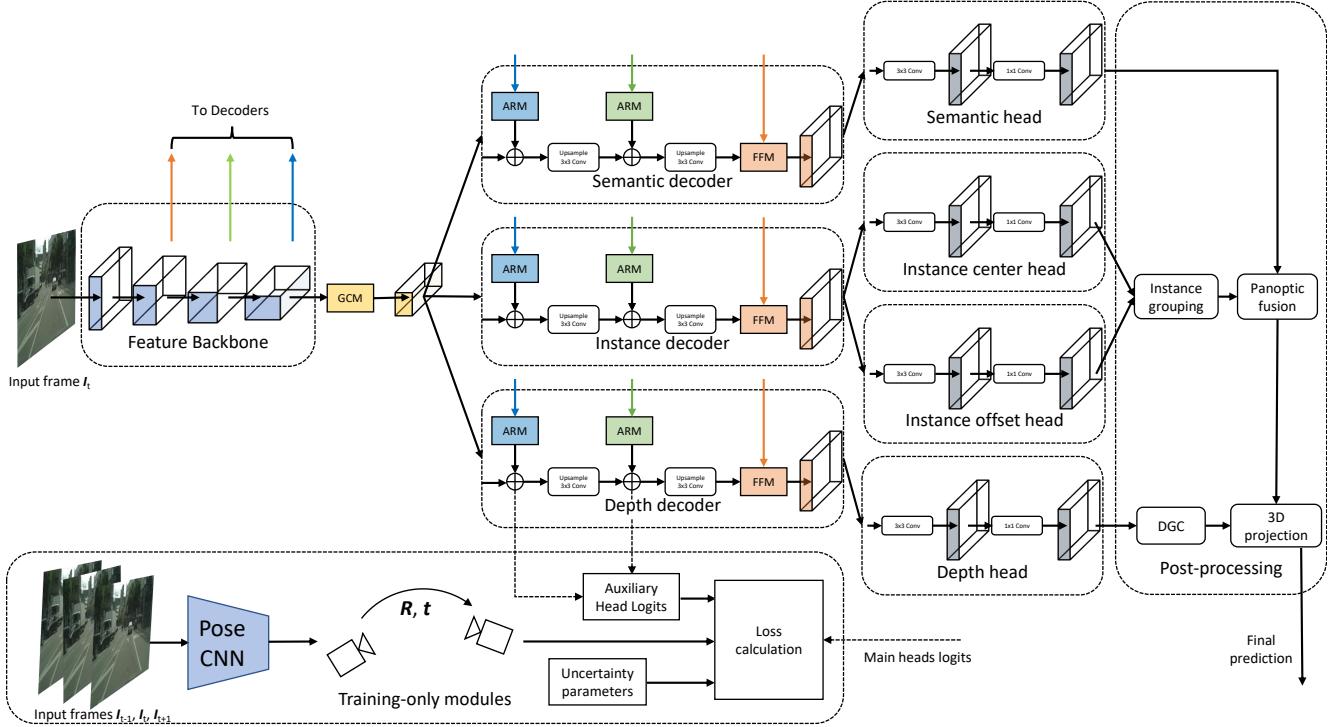


Figure 2: Overview of our MGNet framework. We encode image features using a lightweight backbone and employ a Global Context Module (GCM) to maximize the receptive field. Three task-specific decoders are used to fuse backbone features of different scales using Attention Refinement Modules (ARMs) and Feature Fusion Modules (FFMs). Task specific heads convert the fused features to logits. Object instances are represented as instance centers and pixel-wise offsets, i.e., 2D vectors, representing the x-y-offset to the instance center. The post-processing groups instances to the closest center given the offset prediction and assigns a semantic class based on majority voting. The depth prediction is first scaled using a Dense Geometrical Constrains Module (DGC) and then used to convert the panoptic prediction to the final 3D point cloud prediction. The pose network predicting 6 DOF poses between adjacent camera frames I_{t-1}, I_t and I_{t+1} , auxiliary heads for the multi-view photometric loss, and the uncertainty parameters for multi-task optimization are only used during training.

3.2. Multi-Task Optimization

Panoptic Segmentation: We follow the Panoptic DeepLab approach [7] and use the weighted bootstrapped cross entropy loss, first introduced in [76], for semantic segmentation

$$\mathcal{L}_{\text{seg}} = -\frac{1}{K} \sum_{i=1}^N \omega_i \cdot \mathbb{1}[p_{i,y_i} < t_K] \cdot \log p_{i,y_i}, \quad (1)$$

where p_{i,y_i} is the predicted probability for pixel i and target class y_i . The indicator function $\mathbb{1}[p_{i,y_i} < t_K]$ ensures that the loss is masked using only pixels with the top-K highest loss values. The weight is set to $w_i = 3$ for pixels that belong to instances with a smaller area than 64×64 and $w_i = 1$ for all other pixels. This way, the network is forced to focus on hard pixels and small instances, respectively. For the instance segmentation, we represent each instance by its center of mass. During training, we predict a

2D Gaussian heatmap for each instance center with a fixed sigma of 8 pixels and use a Mean Squared Error (MSE) loss for optimization. For the instance mask prediction, we further encode pixels which belong to thing classes by their offset vector to the corresponding instance center. We use a L_1 loss for optimization of the offset vectors prediction. We weight each loss term according to [76], hence the full panoptic segmentation loss is given by

$$\mathcal{L}_{\text{pan}} = \mathcal{L}_{\text{seg}} + 200 \cdot \mathcal{L}_{\text{mse}} + 0.01 \cdot \mathcal{L}_{L_1}. \quad (2)$$

Self-Supervised Monocular Depth Estimation: We follow recent methods [17, 20], and use a multi-scale photometric loss. We add two auxiliary heads to produce intermediate depth logits. During training, we use a separate pose network based on ResNet18 [23] to predict the 6 DOF relative camera pose between the adjacent frames I_{t-1}, I_t and I_{t+1} , and optimize both networks together. For loss calculation, the context frames $I_c \in \{I_{t-1}, I_{t+1}\}$ are first

warped into the frame \mathbf{I}_t using the predicted depth maps and poses. For each warped image $\hat{\mathbf{I}}_t \in \{\hat{\mathbf{I}}_{t-1 \rightarrow t}, \hat{\mathbf{I}}_{t+1 \rightarrow t}\}$, the photometric loss is calculated according to

$$\mathcal{L}_{\text{phot}}(\mathbf{I}_t, \hat{\mathbf{I}}_t) = \alpha \cdot \frac{1 - \text{SSIM}(\mathbf{I}_t, \hat{\mathbf{I}}_t)}{2} + (1 - \alpha) \cdot |\mathbf{I}_t - \hat{\mathbf{I}}_t| \quad (3)$$

using the Structured Similarity Index (SSIM) [71]. We use a fixed $\alpha = 0.85$ in this work. Following [17], we integrate several improvements into the photometric loss calculation. Instead of averaging over the photometric loss terms for each context frame, we use the minimum reprojection error

$$\mathcal{L}_{\text{phot}}(\mathbf{I}_t, \hat{\mathbf{I}}_s) = \min_{\hat{\mathbf{I}}_s} \mathcal{L}_{\text{phot}}(\mathbf{I}_t, \hat{\mathbf{I}}_s) \quad (4)$$

with $\hat{\mathbf{I}}_s \in \{\hat{\mathbf{I}}_{t-1 \rightarrow t}, \hat{\mathbf{I}}_{t+1 \rightarrow t}, \mathbf{I}_{t-1}, \mathbf{I}_{t+1}\}$. Adding the unwarped image loss terms $\mathcal{L}_{\text{phot}}(\mathbf{I}_t, \mathbf{I}_{t-1})$ and $\mathcal{L}_{\text{phot}}(\mathbf{I}_t, \mathbf{I}_{t+1})$ automatically masks out static pixels from the loss. Furthermore, we use a pixel-wise mask \mathbf{M}_v to mask out pixels without a valid projection. The final photometric loss thus is given by

$$\mathcal{L}_{\text{phot}} = \sum_i \mathcal{L}_{\text{phot}}(\mathbf{I}_t, \mathbf{I}_c) \odot \mathbf{M}_v, \quad (5)$$

where i denotes predictions at different scales. Similar to [16], we regularize the predicted depth map using an image gradient based smoothness term

$$\begin{aligned} \mathcal{L}_{\text{smooth}} = & \sum_i \frac{1}{2^i} (|\delta_x \hat{\mathbf{d}}^*| \exp(-|\delta_x \mathbf{I}_t|) \\ & + |\delta_y \hat{\mathbf{d}}^*| \exp(-|\delta_y \mathbf{I}_t|)). \end{aligned} \quad (6)$$

with the predicted mean-normalized inverse depth $\hat{\mathbf{d}}^*$. The final loss term for self-supervised depth estimation is a weighted sum of the photometric and smoothness loss

$$\mathcal{L}_{\text{depth}} = \mathcal{L}_{\text{phot}} + 0.001 \cdot \mathcal{L}_{\text{smooth}}. \quad (7)$$

Homoscedastic uncertainty weighting: The naive approach of adding up both loss terms may be suboptimal for optimization. Hence, we use the homoscedastic uncertainty σ_i^2 to weight each task i as proposed in [29]. We add uncertainty terms not only to the final task losses, but to each component of each loss. We argue that the fixed loss weights give a good first estimate for optimization, although this might change during training. Therefore, including learnable uncertainties enables the model to adjust the individual loss term relations based on training data. In practice, we learn $s_i = \log(\sigma_i^2)$ instead of σ_i^2 , leading to the final loss

$$\begin{aligned} \mathcal{L} = & \exp(-s_0) \cdot \mathcal{L}_{\text{seg}} + 0.5 \cdot \exp(-s_1) \cdot 200 \cdot \mathcal{L}_{\text{mse}} \\ & + 0.5 \cdot \exp(-s_2) \cdot 0.01 \cdot \mathcal{L}_{L_1} + 0.5 \cdot \exp(-s_3) \cdot \mathcal{L}_{\text{phot}} \\ & + 0.5 \cdot \exp(-s_4) \cdot 0.001 \cdot \mathcal{L}_{\text{smooth}} + 0.5 \cdot \sum_i s_i. \end{aligned} \quad (8)$$

3.3. Post-Processing

The post-processing includes instance grouping, panoptic fusion, depth scaling, and 3D projection.

Instance grouping: Center keypoints are extracted from the heatmap prediction using a keypoint-based non-maximum suppression (NMS) as proposed in [7]. In this method, max pooling with kernel size 7 is applied to the heatmap prediction and pixels are kept only if the pooled prediction is equal to the unpooled prediction. Additionally, a fixed threshold of 0.3 is used to filter out low confidence keypoints. The offset prediction is masked with the semantic prediction to only preserve pixels belonging to thing classes. Instance masks can then be generated by grouping the thing pixels to their closest center keypoint after applying the predicted offsets.

Panoptic fusion: Panoptic fusion assigns class labels to the grouped instances based on the semantic prediction. We use the simple and efficient majority voting introduced in [76] to assign to each instance the class label with the highest pixel count for that mask.

Depth scaling: The depth estimation can only be done up to an unknown scale factor due to the inherent ambiguity of the photometric loss. To rescale the estimated depth map \mathbf{d}_{rel} back to the original scale, we leverage the DGC from [75], but use our panoptic prediction to improve the module. The idea is to estimate a camera height for each ground point based on the predicted depth using the camera geometry. In the original DGC, \mathbf{d}_{rel} is used to generate 3D points $\mathbf{p}_i = (x, y, z)^\top$ from the image and a surface normal $\mathbf{N}(\mathbf{p}_i)$ is calculated for each 3D point to classify points as ground points, where the surface normal is close to the ideal ground surface normal $\mathbf{n} = (0, 1, 0)^\top$. However, this approach produces lots of false positives, e.g., roofs of vehicles. To mitigate this, we use our panoptic prediction, i.e., points with class label `road`, for ground point classification instead. Using these ground points, we estimate a camera height for each ground point using

$$h(\mathbf{p}_i) = \mathbf{N}(\mathbf{p}_i)^\top \mathbf{p}_i. \quad (9)$$

The scale factor is then calculated using the real camera height h and the median of the estimated camera heights \hat{h}_m . Hence, the scaled depth prediction can be calculated as

$$\mathbf{d}_{\text{abs}} = \frac{h}{\hat{h}_m} \cdot \mathbf{d}_{\text{rel}}. \quad (10)$$

3D projection: The 3D projection of the panoptic prediction $\hat{\mathbf{y}}(u, v)$ is calculated by

$$\hat{\mathbf{y}}(x, y, z) = \mathbf{K}^{-1} \mathbf{d}_{\text{abs}} \hat{\mathbf{y}}(u, v) \quad (11)$$

using the scaled depth prediction \mathbf{d}_{abs} and the intrinsic camera matrix \mathbf{K} . In order to only generate 3D points with a valid depth prediction, we exclude certain semantic classes, in our case `sky` and `ego-car`, from the projection.

4. Experimental Results

In this section, we evaluate our multi-task framework quantitatively and qualitatively and provide comparisons with state-of-the-art methods of both domains, panoptic segmentation and self-supervised depth estimation.

4.1. Datasets

We conduct experiments on two popular autonomous driving benchmarks, Cityscapes [8] and KITTI [15].

Cityscapes: The Cityscapes dataset provides video sequences recorded in 50 different cities primarily in Germany. It contains 5 000 images that are annotated with fine-grained panoptic labels for 30 classes, of which 19 are used during evaluation. The dataset is split in 2 975 images for training, 500 for validation and 1 525 for testing. Additional 20 000 images provide coarse panoptic labels, which are not used in this work. Depth annotations are given as pre-computed disparity maps from SGM [24] for the 5 000 fine annotated images, which are only used for evaluation. We train with 20 classes, including the 19 classes used for evaluation and the additional class *ego-car*. We found that using this extra class improves panoptic prediction and enables us to exclude pixels belonging to this class from photometric loss calculation.

KITTI Eigen split: We use the KITTI Eigen split introduced in [11] with pre-processing from Zhou *et al.* [84] to remove static frames. This leads to an image split of 39 810 images for training and 4424 for validation. We use the 652 improved ground truth depth maps introduced in [62] for testing. The depth maps are generated using accumulated lidar points from 5 consecutive frames and stereo information to handle dynamic objects. KITTI does not provide panoptic segmentation annotations for the Eigen split. Hence, we use our best Cityscapes model to generate pseudo labels as in [3] and train on pseudo labels only. We exclude the *ego-car* class, as KITTI images do not have a visible part of the ego vehicle.

4.2. Implementation Details

We use the PyTorch framework [50] and train all models across 4 NVIDIA RTX 2080Ti GPUs. We use InPlace-ABNSync [56] with LeakyReLU [44] activation after each convolution except the last ones in the heads. We adopt a similar training protocol as in [7]. In particular, we use the Adam Optimizer [30] without weight decay and adopt the “poly” learning rate schedule [41] with an initial learning rate of 0.001. We multiply the learning rate by a factor of 10 in all our decoders and heads. Our model backbone as well as the pose network are initialized with pre-trained weights from ImageNet [10]. We do not use other external data such as Mapillary Vistas [47]. During training, we do random color jitter augmentation. Additionally, on

Backbone	PQ \uparrow	RMSE \downarrow	FPS \uparrow
MobileNetV3 [27]	48.6	9.4	29.8
MNASNet100 [59]	50.8	9.0	28.3
EfficientNetLite0 [60]	52.8	8.6	27.7
ResNet18 [23]	53.3	8.8	30.0

Table 1: Comparison of different feature backbones. We compare four lightweight ImageNet [10] pre-trained backbones.

Method	PQ \uparrow	RMSE \downarrow
Single-Task Baseline	54.1	9.3
Multi-Task Baseline	53.3	8.8
+ Uncertainty	54.2	8.6
+ Video Sequence [3]	55.7	8.3

Table 2: Ablation study on Cityscapes. While PQ decreases in the multi-task baseline compared to two single-task models, adding uncertainty weighting and video sequence training show improvements in both, PQ and RMSE.

Cityscapes, we perform random scaling within [0.5, 2.0] and, respectively, randomly crop or pad the images to a fixed size of 1024×1024 pixels. We adjust the intrinsic camera matrix according to the augmentation and calculate a reprojection mask to exclude padded regions as well as *sky* and *ego-car* pixels from the photometric loss calculation. Similar to [3], we use video sequences to generate pseudo labels for training, which enables us to use the full video sequence set for a second training iteration. On Cityscapes, we train for 60k iterations on the fine-annotated set and another 60k iterations on the video sequence set using pseudo labels with batch size 12. On KITTI, we use our best Cityscapes model to generate pseudo labels for panoptic segmentation and train our model for 20 epochs with batch size 12.

In order to evaluate the panoptic segmentation task, we report the panoptic quality (PQ), introduced in [32], where a higher PQ states a more accurate prediction. RMSE is used as the main metric for the evaluation of self-supervised depth estimation. Furthermore, we report other common depth metrics, namely absolute relative error and accuracy under delta threshold for $\delta < 1.25$, $\delta < 1.25^2$, and $\delta < 1.25^3$. In case of RMSE and absolute relative error, a lower value is better, while higher values are better in case of the delta threshold metrics. Additionally, we measure 500 forward passes through our network, including data loading to GPU and post-processing, and report the average runtime or FPS, respectively, to assess the real-time capability of the models. We use TensorRT optimization [49] unless stated otherwise.

Method	Backbone	Input size	PQ \uparrow	GPU	Runtime (ms) \downarrow
Naive Student [3] \dagger	WideResNet41	1025 \times 2049	70.8	V100	396.5
Porzi <i>et al.</i> [52] $*\dagger$	HRNet-W48+	1024 \times 2048	66.7	V100	—
Axial-DeepLab-XL [66] $*$	Axial-ResNet-XL	1025 \times 2049	65.1	V100	—
Panoptic DeepLab [7]	Xception71	1025 \times 2049	63.0	V100	175
EfficientPS [46]	Mod. EfficientNetB5	1024 \times 2048	63.6	Titan RTX	166
Panoptic FCN [38] $*$	Res50-FPN	1024 \times 2048	61.4	V100	—
FPSNet [9]	ResNet50-FPN	1024 \times 2048	55.1	Titan RTX	114
Hou <i>et al.</i> [26]	ResNet50-FPN	1024 \times 2048	58.8	V100	99
Petrovai and Nedevschi [51]	VoVNet2-39	1024 \times 2048	63.0	V100	82
Panoptic DeepLab [4]	SWideRNet-(0.25, 0.25, 0.75)	1025 \times 2049	58.4	V100	63.05
Panoptic DeepLab [7]	MobileNetV3	1025 \times 2049	55.4	V100	63
Ours	ResNet18	1024 \times 2048	55.7	RTX 2080 Ti	44.4

Table 3: Comparison of our method to state-of-the-art for the panoptic segmentation task on the Cityscapes validation set. Methods marked with $*$ do not report runtimes, but based on model size we estimate >100 ms. Methods using external data, *i.e.*, Mapillary Vistas [47], are marked with \dagger .

Method	DS	Resolution	Abs Rel \downarrow	RMSE \downarrow	$\delta < 1.25 \uparrow$	$\delta < 1.25^2 \uparrow$	$\delta < 1.25^3 \uparrow$
SfMLearner [84]	CS+K	416 \times 128	0.176	6.129	0.758	0.921	0.971
Vid2Depth [45]	CS+K	416 \times 128	0.134	5.501	0.827	0.944	0.981
GeoNet [77]	CS+K	416 \times 128	0.132	5.240	0.883	0.953	0.985
DDVO [64]	CS+K	416 \times 128	0.126	4.932	0.851	0.958	0.986
EPC++ [43]	K	640 \times 192	0.120	4.755	0.856	0.961	0.987
Monodepth2 [17]	K	640 \times 192	0.090	3.942	0.914	0.983	0.995
SynDistNet [35]	K	640 \times 192	0.076	3.406	0.931	0.988	0.996
PackNet-SfM [20]	CS+K	1280 \times 384	0.071	3.153	0.944	0.990	0.997
Ours	CS+K	1280 \times 384	0.095	3.761	0.902	0.979	0.992

Table 4: Comparison of our method to state-of-the-art self-supervised depth estimation methods on the KITTI 2015 Eigen split for a distance up to 80 m. We use the improved ground truth maps from [62] for comparison.

4.3. Ablation Studies

We perform ablation studies on Cityscapes. First, we compare four lightweight backbones, namely MobileNetV3 [27], MNASNet100 [59], EfficientNetLite0 [60] and ResNet18 [23]. The results are reported in Table 1. ResNet18 is performing best in terms of PQ and FPS, while being only slightly worse in terms of RMSE compared to EfficientNetLite0. MobileNetV3 and MNASNet100 perform significantly worse. Hence, we use ResNet18 as backbone for all further experiments. Additionally, we investigate the effect of multi-task training, uncertainty weighting, and video sequence training in Table 2. The naive multi-task setting with fixed weights improves RMSE, but deteriorates PQ compared to two separate single-task models. However, adding uncertainty weighting, we see an improvement of 0.1% PQ and 0.7 m RMSE compared to the single-task baseline. By adding video sequence training, we can further increase performance to 55.7 PQ and 8.3 RMSE.

4.4. Cityscapes

We compare our final model with state-of-the-art methods on the Cityscapes validation set in Table 3 with respect to PQ and end-to-end runtime. For a fair runtime comparison, we only infer our panoptic segmentation network part and report the unoptimized performance. Our model is the fastest model with a total end-to-end runtime of 44 ms and over 30 % faster compared to the second fastest method. The margin most likely is even higher than reported, given the fact that all other model runtimes were reported on faster GPUs. In terms of PQ, there is a big gap to top performing methods. This is not surprising, given the fact that these methods need more than 100 ms using much heavier architectures than our method. Compared to Panoptic DeepLab [7] with a MobileNetV3 backbone, which is the most similar variant in literature to our method, our model provides a better speed-accuracy trade-off with 0.3 % better PQ and over 30 % faster runtime. Hou *et al.* [26] state an

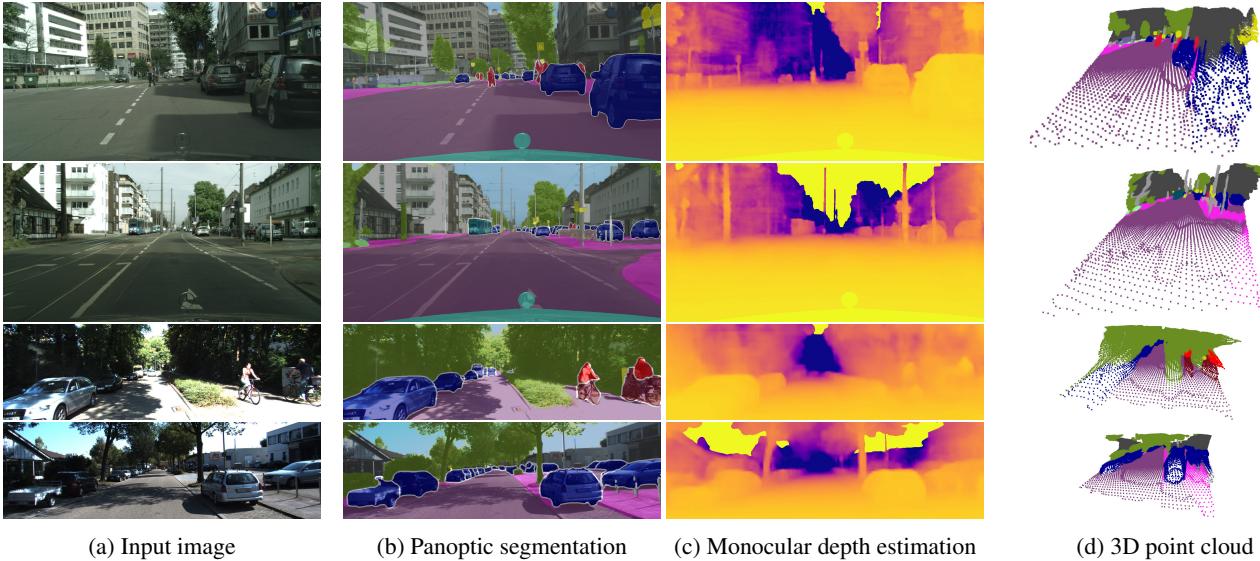


Figure 3: Qualitative results on unseen images from the Cityscapes and KITTI dataset. The columns (from left to right) show the input image, the panoptic prediction, the monocular depth estimation, and the final 3D point cloud prediction, respectively. Instances are omitted in the 3D point clouds for better visualization. The first two rows show predictions on the Cityscapes dataset, while the last two rows show predictions on the KITTI dataset.

optimized runtime of 30 FPS in their paper, being as fast as our optimized network with a higher PQ of 58.8. However, our model additionally predicts depth in a self-supervised fashion. We show qualitative results of our method on unseen images from the Cityscapes dataset in Figure 3.

4.5. KITTI

For KITTI, we use our best model from Cityscapes to generate pseudo labels on the KITTI training set. This enables us to train on the full Eigen split and optimize both tasks in conjunction, similar to Cityscapes. We again use additional uncertainty weighting to balance the loss terms. Qualitative results of our method on unseen images from the KITTI dataset can be found in Figure 3. We report quantitative results for depth estimation and compare them to other self-supervised methods on the KITTI Eigen split in Table 4. Compared to previous methods, our model is on par with Monodepth2 [17] and exceeds most other previous methods. Only two methods, PackNet-SfM [20] and SynDistNet [35], are able to consistently outperform our model. PackNet-SfM uses a heavier network architecture, reaching only 6.25 FPS on full resolution 1280×384 pixel images. In contrast, our method reaches 82 FPS on full resolution images and thus being real-time capable. SynDistNet uses a multi-task approach with explicit semantic guidance. While SynDistNet cannot perform panoptic segmentation, the model shows, that explicit methods can further boost single-task performance compared to our implicit method.

5. Conclusion

In this work, we introduced the task of monocular geometric scene understanding as the combination of panoptic segmentation and self-supervised depth estimation. Tackling this complex task, we introduced MGNet, an efficient architecture, which operates in real-time by carefully combining state-of-the-art concepts from both domains with a lightweight network architecture. We evaluated our model on Cityscapes and KITTI and show competitive results. On Cityscapes, we achieve 55.7 PQ and 8.3 RMSE with 30 FPS on 1024×2048 pixel images. On KITTI, we achieve 3.761 RMSE with 82 FPS on 384×1280 pixel images. While state-of-the-art methods provide higher accuracy on single tasks compared to our work, they are not suited for real-time applications, *e.g.*, autonomous driving.

We hope that our work will inspire researchers to further investigate the task of monocular geometric scene understanding. Future research can focus on using self-supervised methods for panoptic segmentation, integrating related tasks such as 3D object detection into the framework, or investigating the relation between the tasks using explicit methods to boost single-task performance.

Acknowledgment

This research is accomplished within the project UNICARagil (FKZ 16EM00290). We acknowledge the financial support for the project by the Federal Ministry of Education and Research of Germany (BMBF).

References

- [1] Rich Caruana. Multitask learning. *Machine learning*, 28(1):41–75, 1997.
- [2] Vincent Casser, Soeren Pirk, Reza Mahjourian, and Anelia Angelova. Depth prediction without the sensors: Leveraging structure for unsupervised learning from monocular videos. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 8001–8008, 2019.
- [3] Liang-Chieh Chen, Raphael Gontijo Lopes, Bowen Cheng, Maxwell D Collins, Ekin D Cubuk, Barret Zoph, Hartwig Adam, and Jonathon Shlens. Naive-student: Leveraging semi-supervised learning in video sequences for urban scene segmentation. In *Proceedings of the European conference on computer vision (ECCV)*, pages 695–714. Springer, 2020.
- [4] Liang-Chieh Chen, Huiyu Wang, and Siyuan Qiao. Scaling wide residual networks for panoptic segmentation. *arXiv preprint arXiv:2011.11675*, 2020.
- [5] Liang-Chieh Chen, Yukun Zhu, George Papandreou, Florian Schroff, and Hartwig Adam. Encoder-decoder with atrous separable convolution for semantic image segmentation. In *Proceedings of the European conference on computer vision (ECCV)*, pages 801–818. Springer, 2018.
- [6] Yifeng Chen, Guangchen Lin, Songyuan Li, Omar Bourahla, Yiming Wu, Fangfang Wang, Junyi Feng, Mingliang Xu, and Xi Li. Banet: Bidirectional aggregation network with occlusion handling for panoptic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020.
- [7] Bowen Cheng, Maxwell D Collins, Yukun Zhu, Ting Liu, Thomas S Huang, Hartwig Adam, and Liang-Chieh Chen. Panoptic-deeplab: A simple, strong, and fast baseline for bottom-up panoptic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020.
- [8] Marius Cordts, Mohamed Omran, Sebastian Ramos, Timo Rehfeld, Markus Enzweiler, Rodrigo Benenson, Uwe Franke, Stefan Roth, and Bernt Schiele. The cityscapes dataset for semantic urban scene understanding. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.
- [9] D. de Geus, P. Meletis, and G. Dubbelman. Fast panoptic segmentation network. *IEEE Robotics and Automation Letters*, 5(2):1742–1749, 2020.
- [10] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 248–255, 2009.
- [11] David Eigen, Christian Puhrsch, and Rob Fergus. Depth map prediction from a single image using a multi-scale deep network. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 27. Curran Associates, Inc., 2014.
- [12] Naiyu Gao, Yanhu Shan, Yupei Wang, Xin Zhao, Yinan Yu, Ming Yang, and Kaiqi Huang. Ssap: Single-shot instance segmentation with affinity pyramid. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2019.
- [13] Naiyu Gao, Yanhu Shan, Xin Zhao, and Kaiqi Huang. Learning category- and instance-aware pixel embedding for fast panoptic segmentation. *IEEE Transactions on Image Processing*, 30:6013–6023, 2021.
- [14] Ravi Garg, Vijay Kumar Bg, Gustavo Carneiro, and Ian Reid. Unsupervised cnn for single view depth estimation: Geometry to the rescue. In *Proceedings of the European conference on computer vision (ECCV)*, pages 740–756. Springer, 2016.
- [15] Andreas Geiger, Philip Lenz, Christoph Stiller, and Raquel Urtasun. Vision meets robotics: The kitti dataset. *International Journal of Robotics Research (IJRR)*, 2013.
- [16] Clement Godard, Oisin Mac Aodha, and Gabriel J. Brostow. Unsupervised monocular depth estimation with left-right consistency. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.
- [17] Clement Godard, Oisin Mac Aodha, Michael Firman, and Gabriel J. Brostow. Digging into self-supervised monocular depth estimation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2019.
- [18] Kratarth Goel, Praveen Srinivasan, Sarah Tariq, and James Philbin. Quadronet: Multi-task learning for real-time semantic depth aware instance segmentation. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pages 315–324, 2021.
- [19] Ariel Gordon, Hanhan Li, Rico Jonschkowski, and Anelia Angelova. Depth from videos in the wild: Unsupervised monocular depth learning from unknown cameras. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 8977–8986, 2019.
- [20] Vitor Guizilini, Rares Ambrus, Sudeep Pillai, Allan Raventos, and Adrien Gaidon. 3d packing for self-supervised monocular depth estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020.
- [21] Vitor Guizilini, Rui Hou, Jie Li, Rares Ambrus, and Adrien Gaidon. Semantically-guided representation learning for self-supervised monocular depth. In *International Conference on Learning Representations (ICLR)*, 2020.
- [22] Kaiming He, Georgia Gkioxari, Piotr Dollar, and Ross Girshick. Mask r-cnn. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 2017.
- [23] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR)*, pages 770–778, 2016.
- [24] Heiko Hirschmuller. Stereo processing by semiglobal matching and mutual information. *IEEE Transactions on pattern analysis and machine intelligence*, 30(2):328–341, 2007.
- [25] Namdar Homayounfar, Yuwen Xiong, Justin Liang, Wei-Chiu Ma, and Raquel Urtasun. Levelset r-cnn: A deep variational method for instance segmentation. In *Proceedings of the European conference on computer vision (ECCV)*, pages 555–571. Springer, 2020.
- [26] Rui Hou, Jie Li, Arjun Bhargava, Allan Raventos, Vitor Guizilini, Chao Fang, Jerome Lynch, and Adrien Gaidon. Real-time panoptic segmentation from dense detections. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020.

- [27] Andrew Howard, Mark Sandler, Grace Chu, Liang-Chieh Chen, Bo Chen, Mingxing Tan, Weijun Wang, Yukun Zhu, Ruoming Pang, Vijay Vasudevan, et al. Searching for mobilenetv3. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 1314–1324, 2019.
- [28] Ankit Jha, Awanish Kumar, Biplob Banerjee, and Subhasis Chaudhuri. Adamt-net: An adaptive weight learning based multi-task learning model for scene understanding. In *Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, pages 706–707, 2020.
- [29] Alex Kendall, Yarin Gal, and Roberto Cipolla. Multi-task learning using uncertainty to weigh losses for scene geometry and semantics. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 7482–7491, 2018.
- [30] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *International Conference on Learning Representations (ICLR)*, 2015.
- [31] Alexander Kirillov, Ross Girshick, Kaiming He, and Piotr Dollar. Panoptic feature pyramid networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019.
- [32] Alexander Kirillov, Kaiming He, Ross Girshick, Carsten Rother, and Piotr Dollár. Panoptic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 9404–9413, 2019.
- [33] Marvin Klingner, Andreas Bar, and Tim Fingscheidt. Improved noise and attack robustness for semantic segmentation by using multi-task training with self-supervised depth estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, pages 320–321, 2020.
- [34] Marvin Klingner, Jan-Aike Termöhlen, Jonas Mikolajczyk, and Tim Fingscheidt. Self-supervised monocular depth estimation: Solving the dynamic object problem by semantic guidance. In *Proceedings of the European conference on computer vision (ECCV)*, pages 582–600. Springer, 2020.
- [35] Varun Ravi Kumar, Marvin Klingner, Senthil Yogamani, Stefan Milz, Tim Fingscheidt, and Patrick Mader. Syndistnet: Self-supervised monocular fisheye camera distance estimation synergized with semantic segmentation for autonomous driving. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pages 61–71, 2021.
- [36] Jie Li, Allan Raventos, Arjun Bhargava, Takaaki Tagawa, and Adrien Gaidon. Learning to fuse things and stuff. *arXiv preprint arXiv:1812.01192*, 2018.
- [37] Ruihao Li, Sen Wang, Zhiqiang Long, and Dongbing Gu. Undeepvo: Monocular visual odometry through unsupervised deep learning. In *IEEE international conference on robotics and automation (ICRA)*, pages 7286–7291, 2018.
- [38] Yanwei Li, Hengshuang Zhao, Xiaojuan Qi, Liwei Wang, Zeming Li, Jian Sun, and Jiaya Jia. Fully convolutional networks for panoptic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 214–223, 2021.
- [39] Justin Liang, Namdar Homayounfar, Wei-Chiu Ma, Yuwen Xiong, Rui Hu, and Raquel Urtasun. Polytransform: Deep polygon transformer for instance segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 9131–9140, 2020.
- [40] Huanyu Liu, Chao Peng, Changqian Yu, Jingbo Wang, Xu Liu, Gang Yu, and Wei Jiang. An end-to-end network for panoptic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 6172–6181, 2019.
- [41] Wei Liu, Andrew Rabinovich, and Alexander C Berg. Parsenet: Looking wider to see better. *arXiv preprint arXiv:1506.04579*, 2015.
- [42] Jonathan Long, Evan Shelhamer, and Trevor Darrell. Fully convolutional networks for semantic segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3431–3440, 2015.
- [43] Chenxu Luo, Zhenheng Yang, Peng Wang, Yang Wang, Wei Xu, Ram Nevatia, and Alan Yuille. Every pixel counts++: Joint learning of geometry and motion with 3d holistic understanding. *IEEE transactions on pattern analysis and machine intelligence*, 42(10):2624–2641, 2019.
- [44] Andrew L Maas, Awni Y Hannun, and Andrew Y Ng. Rectifier nonlinearities improve neural network acoustic models. In *Proceedings of the 30th International Conference on Machine Learning (ICML)*, 2013.
- [45] Reza Mahjourian, Martin Wicke, and Anelia Angelova. Unsupervised learning of depth and ego-motion from monocular video using 3d geometric constraints. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5667–5675, 2018.
- [46] Rohit Mohan and Abhinav Valada. Efficientps: Efficient panoptic segmentation. *International Journal of Computer Vision (IJCV)*, 2021.
- [47] Gerhard Neuhold, Tobias Ollmann, Samuel Rota Bulò, and Peter Kontschieder. The mapillary vistas dataset for semantic understanding of street scenes. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 2017.
- [48] Davy Neven, Bert De Brabandere, Stamatios Georgoulis, Marc Proesmans, and Luc Van Gool. Fast scene understanding for autonomous driving. *arXiv preprint arXiv:1708.02550*, 2017.
- [49] NVIDIA. Tensorrt library. <https://developer.nvidia.com/tensorrt>.
- [50] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. Pytorch: An imperative style, high-performance deep learning library. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 32, pages 8024–8035. Curran Associates, Inc., 2019.
- [51] Andra Petrovai and Sergiu Nedevschi. Real-time panoptic segmentation with prototype masks for automated driving. In *IEEE Intelligent Vehicles Symposium (IV)*, pages 1400–1406, 2020.

- [52] Lorenzo Porzi, Samuel Rota Bulò, and Peter Kontschieder. Improving panoptic segmentation at all scales. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 7302–7311, 2021.
- [53] Lorenzo Porzi, Samuel Rota Bulò, Aleksander Colovic, and Peter Kontschieder. Seamless scene segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019.
- [54] Siyuan Qiao, Liang-Chieh Chen, and Alan Yuille. Detectors: Detecting objects with recursive feature pyramid and switchable atrous convolution. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 10213–10224, 2021.
- [55] Siyuan Qiao, Yukun Zhu, Hartwig Adam, Alan Yuille, and Liang-Chieh Chen. Vip-deeplab: Learning visual perception with depth-aware video panoptic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3997–4008, 2021.
- [56] Samuel Rota Bulò, Lorenzo Porzi, and Peter Kontschieder. In-place activated batchnorm for memory-optimized training of dnns. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.
- [57] Faraz Saeedan and Stefan Roth. Boosting monocular depth with panoptic segmentation maps. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pages 3853–3862, 2021.
- [58] Konstantin Sofiiuk, Olga Barinova, and Anton Konushin. Adaptis: Adaptive instance selection network. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2019.
- [59] Mingxing Tan, Bo Chen, Ruoming Pang, Vijay Vasudevan, Mark Sandler, Andrew Howard, and Quoc V Le. Mnasnet: Platform-aware neural architecture search for mobile. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2820–2828, 2019.
- [60] Mingxing Tan and Quoc Le. Efficientnet: Rethinking model scaling for convolutional neural networks. In *International Conference on Machine Learning*, pages 6105–6114. PMLR, 2019.
- [61] A. Tao, K. Sapra, and Bryan Catanzaro. Hierarchical multi-scale attention for semantic segmentation. *arXiv preprint arXiv:2005.10821*, 2020.
- [62] Jonas Uhrig, Nick Schneider, Lukas Schneider, Uwe Franke, Thomas Brox, and Andreas Geiger. Sparsity invariant cnns. In *International Conference on 3D Vision (3DV)*, 2017.
- [63] Simon Vandenhende, Stamatios Georgoulis, and Luc Van Gool. Mti-net: Multi-scale task interaction networks for multi-task learning. In *Proceedings of the European conference on computer vision (ECCV)*, pages 527–543. Springer, 2020.
- [64] Chaoyang Wang, José Miguel Buenaposada, Rui Zhu, and Simon Lucey. Learning depth from monocular videos using direct methods. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2022–2030, 2018.
- [65] Huiyu Wang, Yukun Zhu, Hartwig Adam, Alan Yuille, and Liang-Chieh Chen. Max-deeplab: End-to-end panoptic segmentation with mask transformers. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5463–5474, 2021.
- [66] Huiyu Wang, Yukun Zhu, Bradley Green, Hartwig Adam, Alan Yuille, and Liang-Chieh Chen. Axial-deeplab: Stand-alone axial-attention for panoptic segmentation. In *Proceedings of the European conference on computer vision (ECCV)*, pages 108–126. Springer, 2020.
- [67] Jingdong Wang, Ke Sun, Tianheng Cheng, Borui Jiang, Chaorui Deng, Yang Zhao, Dong Liu, Yadong Mu, Mingkui Tan, Xinggang Wang, Wenyu Liu, and Bin Xiao. Deep high-resolution representation learning for visual recognition. *IEEE transactions on pattern analysis and machine intelligence*, 2020.
- [68] Lijun Wang, Jianming Zhang, Oliver Wang, Zhe Lin, and Huchuan Lu. Sdc-depth: Semantic divide-and-conquer network for monocular depth estimation. In *Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 541–550, 2020.
- [69] Xinlong Wang, Tao Kong, Chunhua Shen, Yuning Jiang, and Lei Li. Solo: Segmenting objects by locations. In *Proceedings of the European conference on computer vision (ECCV)*, pages 649–665. Springer, 2020.
- [70] Xinlong Wang, Rufeng Zhang, Tao Kong, Lei Li, and Chunhua Shen. Solov2: Dynamic and fast instance segmentation. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 33, pages 17721–17732. Curran Associates, Inc., 2020.
- [71] Zhou Wang, Alan C Bovik, Hamid R Sheikh, and Eero P Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE transactions on image processing*, 13(4):600–612, 2004.
- [72] Yuwen Xiong, Renjie Liao, Hengshuang Zhao, Rui Hu, Min Bai, Ersin Yumer, and Raquel Urtasun. Upsnet: A unified panoptic segmentation network. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 8818–8826, 2019.
- [73] Yuwen Xiong, Renjie Liao, Hengshuang Zhao, Rui Hu, Min Bai, Ersin Yumer, and Raquel Urtasun. Upsnet: A unified panoptic segmentation network. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019.
- [74] Dan Xu, Wanli Ouyang, Xiaogang Wang, and Nicu Sebe. Pad-net: Multi-tasks guided prediction-and-distillation network for simultaneous depth estimation and scene parsing. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 675–684, 2018.
- [75] F. Xue, G. Zhuo, Z. Huang, W. Fu, Z. Wu, and M. H. Ang. Toward hierarchical self-supervised monocular absolute depth estimation for autonomous driving applications. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 2330–2337, 2020.
- [76] Tien-Ju Yang, Maxwell D Collins, Yukun Zhu, Jyh-Jing Hwang, Ting Liu, Xiao Zhang, Vivienne Sze, George Papandreou, and Liang-Chieh Chen. Deeperlab: Single-shot image parser. *arXiv preprint arXiv:1902.05093*, 2019.
- [77] Zhichao Yin and Jianping Shi. Geonet: Unsupervised learning of dense depth, optical flow and camera pose. In *Pro-*

- ceedings of the IEEE conference on computer vision and pattern recognition (CVPR)*, pages 1983–1992, 2018.
- [78] Changqian Yu, Jingbo Wang, Chao Peng, Changxin Gao, Gang Yu, and Nong Sang. Bisenet: Bilateral segmentation network for real-time semantic segmentation. In *Proceedings of the European conference on computer vision (ECCV)*, pages 325–341. Springer, 2018.
 - [79] Amir R Zamir, Alexander Sax, Nikhil Cheerla, Rohan Suri, Zhangjie Cao, Jitendra Malik, and Leonidas J Guibas. Robust learning through cross-task consistency. In *Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 11197–11206, 2020.
 - [80] Zhenyu Zhang, Zhen Cui, Chunyan Xu, Yan Yan, Nicu Sebe, and Jian Yang. Pattern-affinitive propagation across depth, surface normal and semantic segmentation. In *Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4106–4115, 2019.
 - [81] Hengshuang Zhao, Jianping Shi, Xiaojuan Qi, Xiaogang Wang, and Jiaya Jia. Pyramid scene parsing network. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2881–2890, 2017.
 - [82] Hang Zhou, David Greenwood, Sarah Taylor, and Han Gong. Constant velocity constraints for self-supervised monocular depth estimation. In *European Conference on Visual Media Production*, pages 1–8, 2020.
 - [83] Junsheng Zhou, Yuwang Wang, Kaihuai Qin, and Wenjun Zeng. Unsupervised high-resolution depth learning from videos with dual networks. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 6872–6881, 2019.
 - [84] Tinghui Zhou, Matthew Brown, Noah Snavely, and David G Lowe. Unsupervised learning of depth and ego-motion from video. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1851–1858, 2017.
 - [85] Yi Zhu, Karan Sapra, Fitsum A. Reda, Kevin J. Shih, Shawn Newsam, Andrew Tao, and Bryan Catanzaro. Improving semantic segmentation via video propagation and label relaxation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019.
 - [86] Yuliang Zou, Zelun Luo, and Jia-Bin Huang. Df-net: Unsupervised joint learning of depth and flow using cross-task consistency. In *Proceedings of the European conference on computer vision (ECCV)*, pages 36–53. Springer, 2018.