P3Depth: Monocular Depth Estimation with a Piecewise Planarity Prior (Supplementary Material)

Vaishakh Patil[1] Christos Sakaridis[1] Alexander Liniger[1] Luc Van Gool[1,2]
[1]Computer Vision Lab, ETH Zürich [2]PSI, KU Leuven

1. Network Architecture

Encoder: Following the recent works [3, 7], we use a ResNet101 [6] as the encoder for the image. Each ResNet block consists of series of convolution operations with stride of 2 and pooling operations. The receptive field of the convolution is increased by decreasing resolution of the feature maps. This helps to capture more contextual information while compromising the feature map resolution. The final size of the feature map is usually 1/32 of the input image. The original ResNet is designed for the image classification task. To utilize it for a per-pixel prediction task, we remove the last 3 layers, i.e. pooling layer, fully-connected layer and the softmax layer. The ResNet encoder can be divided into 4 different blocks. Each block generates feature maps of different resolution (scales). These feature maps from different scales can be used as skip connections, i.e. fused with decoder outputs to integrate different level of semantic information. The output of the last encoder block is fetched to both decoder heads. Both decoder heads also receive the skip connection information.

Decoder: We base our decoder on [9] following [10]. We replace all ReLU operations with ELU [1] nonlinearities. The decoder is assembled from three modules: 1) *Feature fusion modules:* For each of these modules, residual convolution block is used to transform the skip connection feature map from the ResNet encoder. The output of the residual convolution block is fused with output of last feature fusion block using summation operation. Finally, the feature maps are upsampled to match the resolution of next layers input. 2) *Residual convolution modules:* This module is a series of two units of ELU and 3×3 convolution operations to merge the output of a previous decoder feature map output with a previous feature fusion module output 3) *Adaptive output module:* This is applied at the last stage to get the final output. It consist of two 3×3 convolution operation followed by up-sampling.

Plane coefficient decoder: The last layer of this decoder head is modified to output 4-channels for each planar coefficient instead of single channel depth.

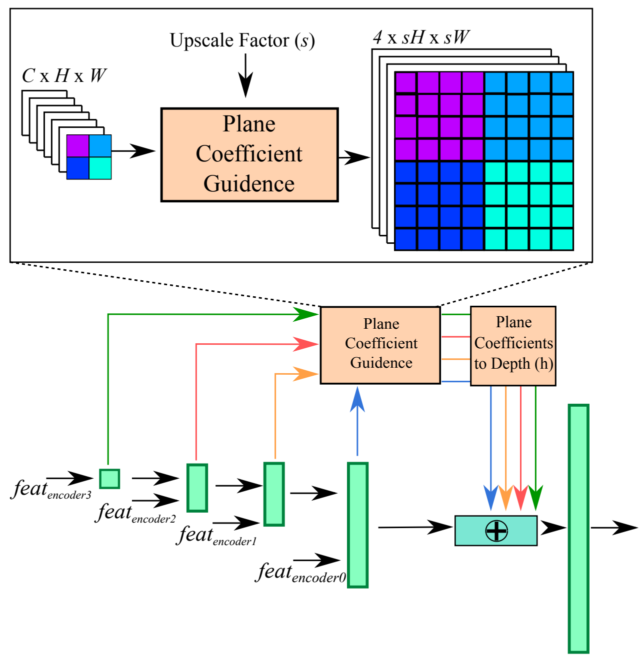

Figure 1: **Plane coefficient guidance module.**

Offset Vector field decoder: The last layer of this decoder head is modified to output 3-channels, i.e. two channels for the offset vector field and one for the confidence. The offset vector field is restricted by tanh layers and the confidence is generated through a sigmoid layer.

Plane coefficient guidance: This module is loosely based on [7]. The output of each decoder block is passed through the Plane coefficient guidance module to generate 4 channels of plane coefficients. The output size of the guidance module is up-sampled to match the input size of last decoder layer. At the end, these plane coefficients from each scale are converted into depth. All these depth maps are concatenated with feature map of the previous decoder layer passed to the last decoder layer.

2. Additional Results

KITTI Benchmark [5]: In this section we present the results of KITTI Benchmark server evaluation. Note that we train our model only on the KITTI Eigen split [2] training data. It can be seen in Table 1 that our results are on a par with SOTA methods and superior than the baseline. However, [7] performs better on this test set. In comparison with [11], we have a better absolute relative error and our performance is comparable to [11] in all other metrics. The drop in overall performance is expected considering the design of our method. Our method is specially designed to identify planar regions in the scene, to improve the depth quality. So, as the depth of the scene increases, the projections of distant parts of the scene get smaller. This causes difficulties in predicting offset vector field in these regions. We have already seen that our method produced the SOTA results on the Garg split [4], in which the maximum depth value is 50m. Due to the aforementioned reason, when tested on Eigen split [2] with max depth of 80m, we observe degradation in the performance. The KITTI Benchmark extends beyond that with 80m+ distances, thus affecting our results due to similar reasons.

Table 1: **Results of KITTI Evaluation Server.**

Method	SILog	sqErrorRel	absErrorRel	iRMSE
Official Baseline	18.19	7.32	14.24	18.50
VNL [11]	<u>12.65</u>	<u>2.46</u>	10.15	<u>13.02</u>
BTS [7]	**11.67**	**2.21**	**9.04**	**12.23**
Ours	12.82	2.53	<u>9.92</u>	13.71

Qualitative Results: Here, we present additional qualitative results on both KITTI [5] and NYU Depth-v2 [8] datasets. We start with some examples from the KITTI dataset. We present some of the best cases along with the failure cases on this dataset. Additionally, we provide visualizations of the predicted depth maps and offset vector fields on NYU Depth-v2. Finally, we use the predicted depth maps to reconstruct the scenes and demonstrate quality in 3D. We observe that the predicted depth maps produce 3D reconstructions which are consistent with ground-truth point clouds and preserve the structure of the scene.

References

[1] Djork-Arné Clevert, Thomas Unterthiner, and Sepp Hochreiter. Fast and accurate deep network learning by exponential linear units (elus). *arXiv preprint arXiv:1511.07289*, 2015. 1

[2] 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*, 2014. 2

[3] Huan Fu, Mingming Gong, Chaohui Wang, Kayhan Batmanghelich, and Dacheng Tao. Deep ordinal regression network for monocular depth estimation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018. 1

[4] Ravi Garg, Vijay Kumar B.G., Gustavo Carneiro, and Ian Reid. Unsupervised CNN for single view depth estimation: Geometry to the rescue. In *The European Conference on Computer Vision (ECCV)*, 2016. 2

[5] Andreas Geiger, Philip Lenz, Christoph Stiller, and Raquel Urtasun. Vision meets robotics: The kitti dataset. *The International Journal of Robotics Research*, 32(11):1231–1237, 2013. 2

[6] 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*, 2016. 1

[7] Jin Han Lee, Myung-Kyu Han, Dong Wook Ko, and Il Hong Suh. From big to small: Multi-scale local planar guidance for monocular depth estimation. *arXiv e-prints*, abs/1907.10326, July 2019. 1, 2

[8] Nathan Silberman, Derek Hoiem, Pushmeet Kohli, and Rob Fergus. Indoor segmentation and support inference from rgbd images. In *European conference on computer vision*, 2012. 2

[9] Ke Xian, Chunhua Shen, Zhiguo Cao, Hao Lu, Yang Xiao, Ruibo Li, and Zhenbo Luo. Monocular relative depth perception with web stereo data supervision. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018. 1

[10] Ke Xian, Jianming Zhang, Oliver Wang, Long Mai, Zhe Lin, and Zhiguo Cao. Structure-guided ranking loss for single image depth prediction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020. 1

[11] Wei Yin, Yifan Liu, Chunhua Shen, and Youliang Yan. Enforcing geometric constraints of virtual normal for depth prediction. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2019. 2

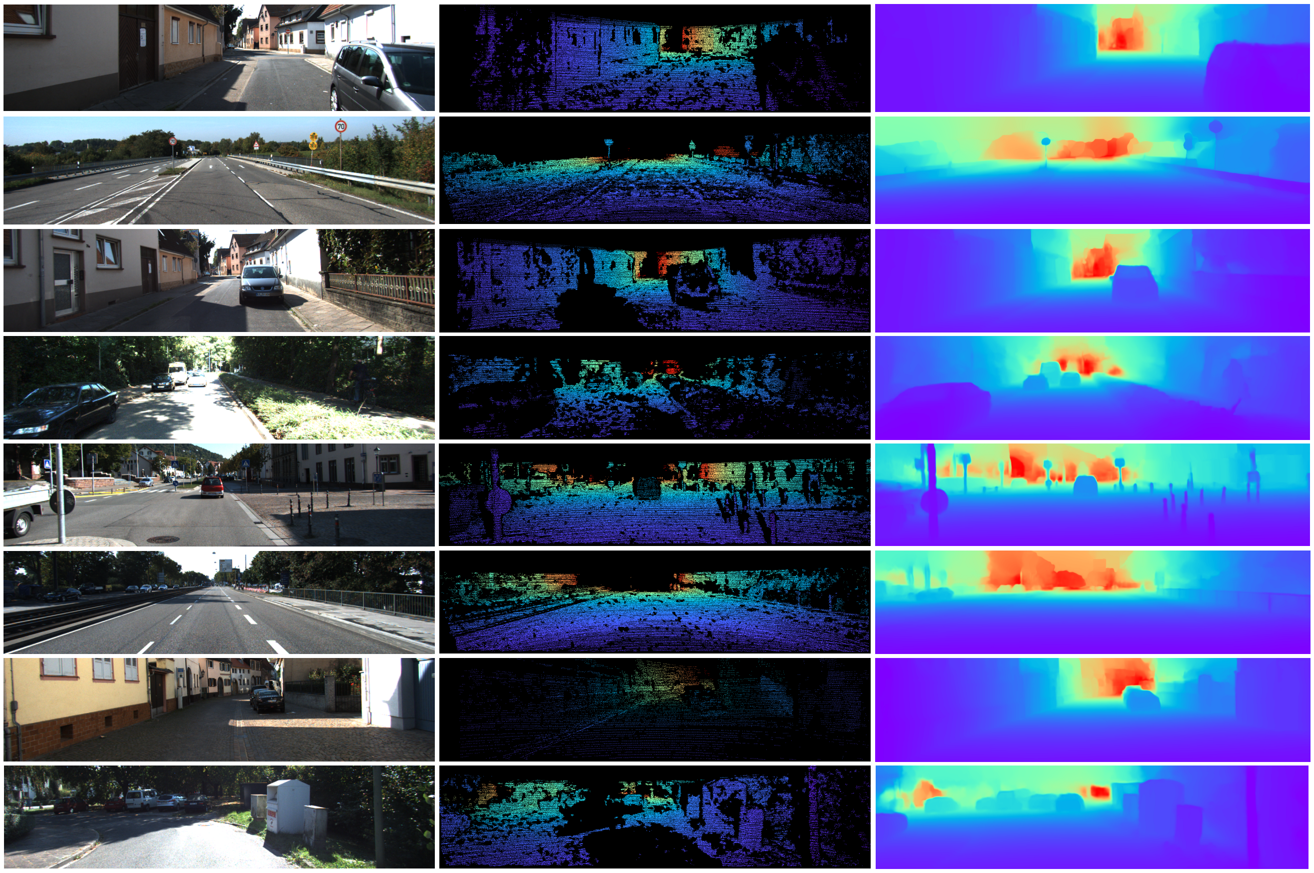

Image **Ground-truth Depth** **Predicted Depth**

Figure 2: **Visualization of predictions on KITTI dataset.**

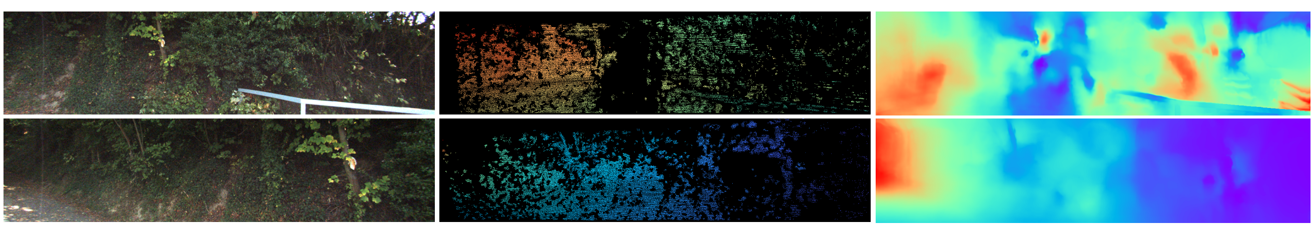

Image **Ground-truth Depth** **Predicted Depth**

Figure 3: **Visualization of some failure cases on KITTI dataset.**

Image **Predicted off. vec. field** **Predicted Depth**

Figure 4: **Visualization of predictions on NYU Depth-v2.**

Figure 5: **Additional reconstruction examples from NYU Depth-v2.**