PixCuboid: Room Layout Estimation from Multi-view Featuremetric Alignment

Supplementary Material

A. Training Details

In this section we provide a more thorough description of how PixCuboid's neural network is trained. We employ a two-stage training procedure where first the edge maps $\mathbf{E}_i$ are pre-trained with a weighted MSE loss (summed over all scale levels), using line renderings of the ground truth cuboids as target images. Pixels that lie on the cuboid edges are up-weighted by a factor of five. The input images are resized to 512 pixels in height while maintaining the aspect ratio. The ResNet-101 [2] encoder is initialized with weights trained on ImageNet-1K [1] and is not frozen during this first training stage. In the first training stage a batch size of 10 is used and 150 examples are randomly sampled from each training scene at every epoch.

Next, the full network is trained with the loss in Eq. (7), applied at each scale and summed. We define a success threshold for the loss at each scale level (48, 12 and 3 pixels, respectively) and if the optimization is not successful the loss on the subsequent level is zeroed out (by setting $\gamma_m = 0$ or $\gamma_f = 0$, otherwise $\gamma_m = \gamma_f = 1$), to prevent learning from examples that are too difficult. Random horizontal cropping is performed after resizing, resulting in images of size 512x512. The 2D-3D correspondences $\{(\boldsymbol{x}_{ik}^{GT}, \boldsymbol{X}_{ik}^{GT})\}$ are found by taking the vertices of the semantic mesh labeled "floor", "wall" or "ceiling" that are visible in each particular view. From these we select 256 points randomly during training to compute the loss. As described in Sec. 3.3 we use guided point sampling for the image points $\{\boldsymbol{x}_{ik}\}$. 256 points are sampled, individually on each scale level with $\gamma = 4$. We set $\beta = 0$ during training since the vanishing point cost does not include any learned component, and let $\alpha = 0.1$. 40 points are sampled on each cuboid edge to compute the edge cost. We run three iterations of LM optimization on each scale level. Cuboids are initialized from the ground truth by applying a random rotation in the $[0°, 15°]$ range, followed by translation of up to 0.5 m in each direction and a resizing of the sides between $[-1, 1.5]$ m. The learned damping parameters of the LM optimization are handled like in [4]. Feature maps $\mathbf{F}_i$ have dimension 128, 128 and 32 on the coarse, medium and fine scale levels, respectively. We use a batch size of 4 and sample 50 examples per training scene at each epoch in the second training stage. The network weights (30 million parameters) are saved at each epoch and we pick the ones that minimize the warp loss Eq. (7) on the validation set.

In both stages the Adam [3] optimizer is used to train the network, with a learning rate of 5×10^{-6}. Gradients are clipped to the $[-1, 1]$ range. The training is run for 10 epochs. The pre-training takes 13 h and the training of the full network finishes in 27 h, using a NVIDIA TITAN V GPU.

B. Results

In Fig. 1 we display additional examples of predicted room layouts in 2D-3D-Semantics. Figs. 2 and 3 present extra failure cases and cuboid initializations, respectively. Table Tab. 1 contains the full set of results for the ablation study in Sec. 5.2.

We also visualize the feature-, edge- and confidence maps (on the finest scale level) for the first five images of an image tuple in our ScanNet++ v2 test in Fig. 4. PixCuboid learns features that are consistent between views (second column) and ignores clutter by assigning high confidence to image points that lie on the floor, walls or ceiling (third column). It can often predict the cuboid edges accurately despite the presence of occluding objects (fourth column).

References

[1] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. ImageNet: A Large-Scale Hierarchical Image Database. In *Computer Vision and Pattern Recognition (CVPR)*, 2009. 1

[2] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep Residual Learning for Image Recognition. In *Computer Vision and Pattern Recognition (CVPR)*, 2016. 1

[3] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014. 1

[4] Paul-Edouard Sarlin, Ajaykumar Unagar, Mans Larsson, Hugo Germain, Carl Toft, Viktor Larsson, Marc Pollefeys, Vincent Lepetit, Lars Hammarstrand, Fredrik Kahl, et al. Back to the Feature: Learning Robust Camera Localization from Pixels to Pose. In *Computer Vision and Pattern Recognition (CVPR)*, 2021. 1

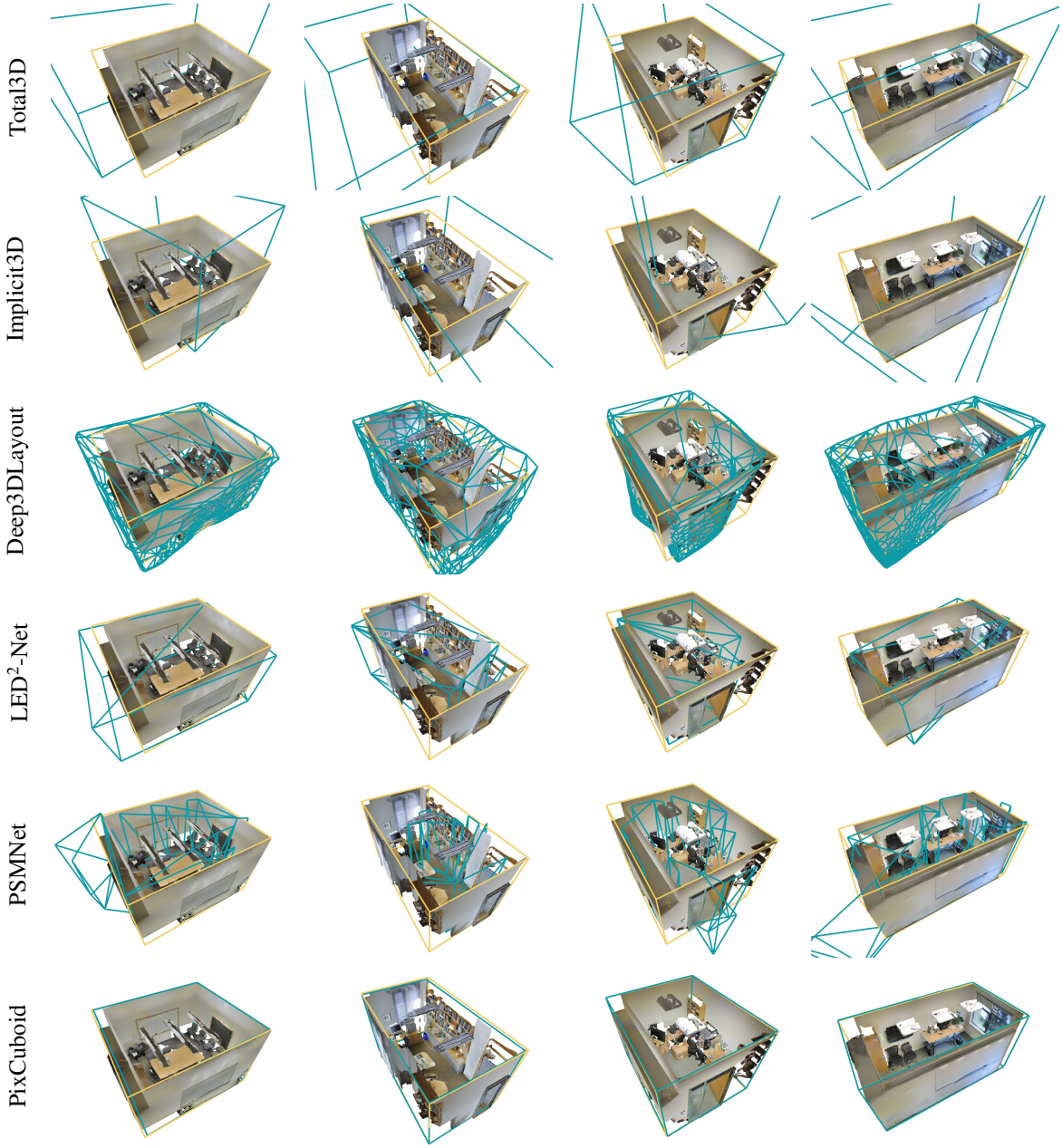

Figure 1. Qualitative comparisons of predicted room layouts for spaces in 2D-3D-Semantics. Predictions are shown in blue and the ground truth cuboids in yellow. None of the methods are trained on this dataset.

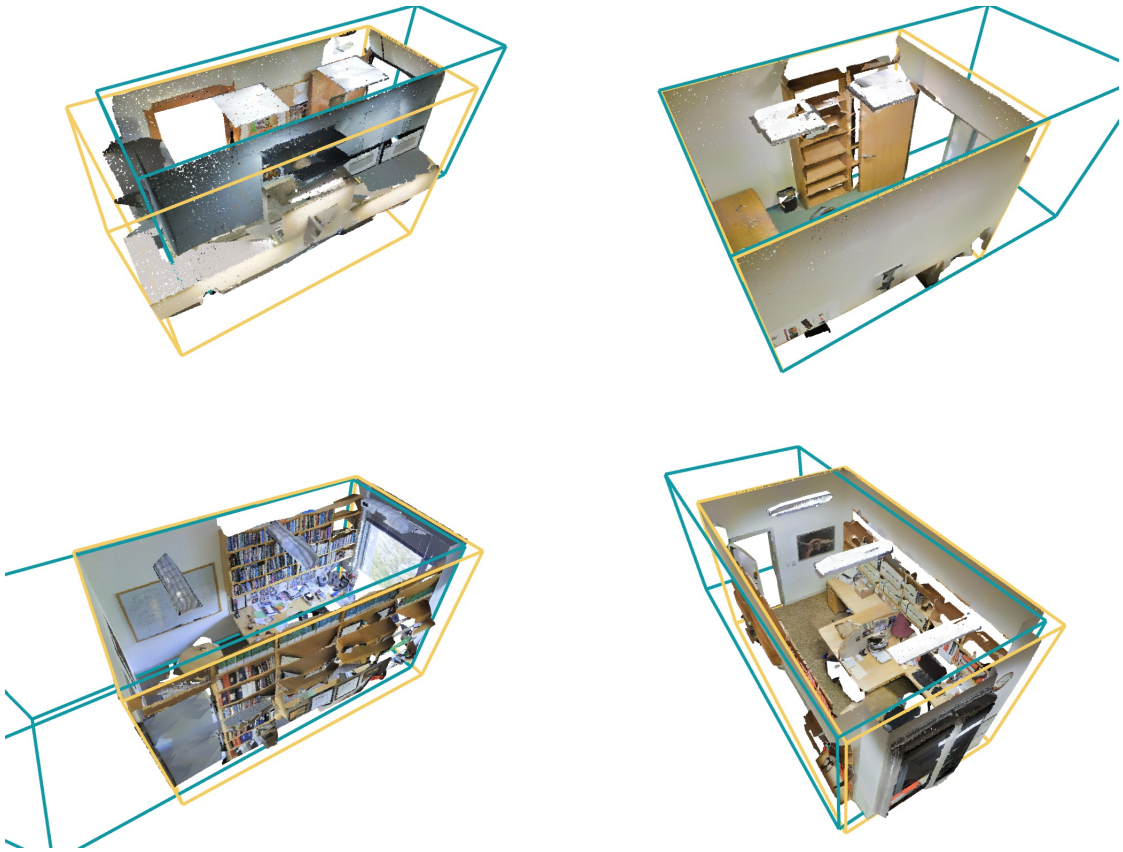

Figure 2. Failure cases of PixCuboid on 2D-3D-Semantics. Predicted room layouts are shown in **blue** and ground truth cuboids in **yellow**.

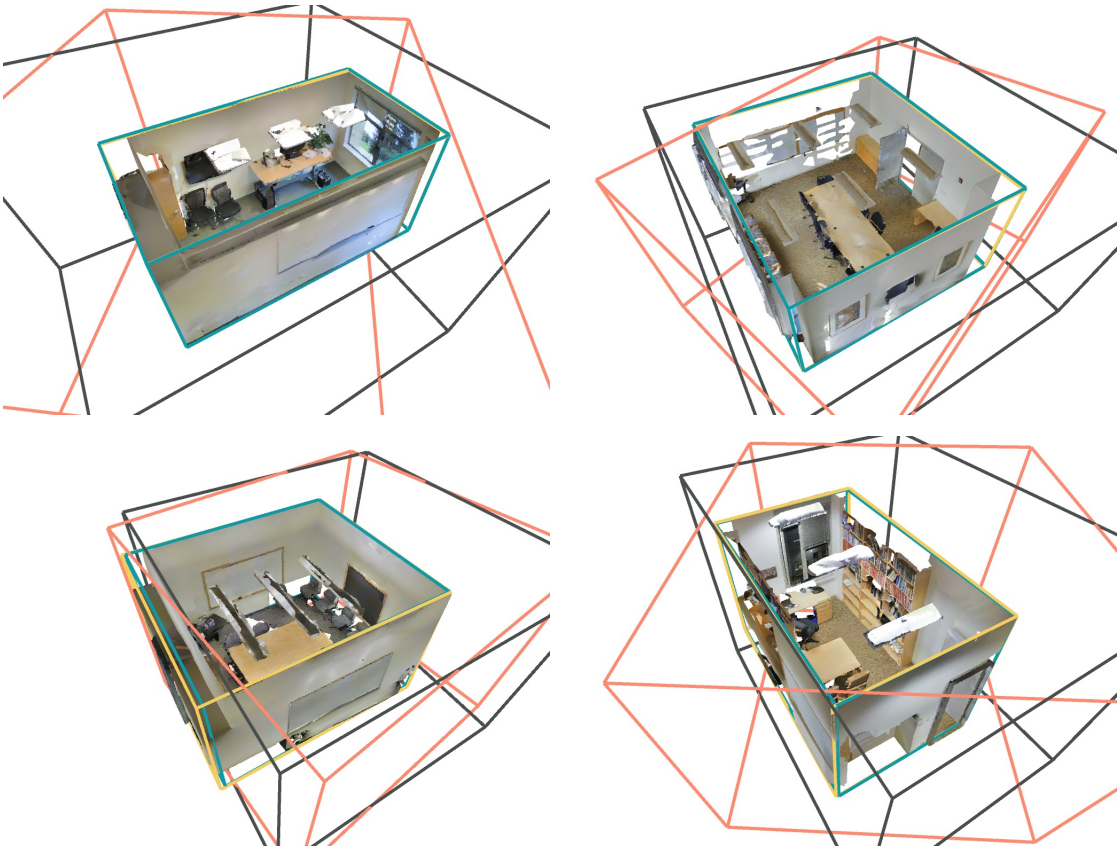

Figure 3. Cuboid initialization examples on 2D-3D-Semantics. Even from poor initial estimates (**red**), PixCuboid can align the cuboids with the help of vanishing points (**gray**) and is able to converge to accurate layouts (**blue**). The ground truth is shown in **yellow**.

		IoU ↑	Chamfer ↓	Rot. ↓	Depth ↓	Normal ↑	Success ↑
Feat.	RGB	24.2	2.51 m	36.4°	1.02 m	7.6	1.4%
	PixLoc (CMU)	25.3	2.30 m	33.4°	1.02 m	10.9	1.1%
	PixCuboid (E_{feat} only)	**35.2**	**1.77 m**	**23.0°**	**0.75 m**	**30.8**	**5.8%**
Cost	E_{feat}	35.2	1.77 m	23.0°	0.75 m	30.8	5.8%
	E_{edge}	76.1	0.51 m	4.7°	0.23 m	86.5	43.6%
	E_{feat} + E_{edge}	81.9	0.39 m	3.8°	0.17 m	88.9	61.4%
	E_{feat} + E_{VP}	44.6	1.24 m	1.5°	0.57 m	79.4	21.7%
	E_{edge} + E_{VP}	83.1	0.29 m	**1.3°**	0.13 m	95.5	51.9%
	E_{feat} + E_{edge} + E_{VP}	**87.2**	**0.22 m**	**1.3°**	**0.09 m**	**96.1**	**67.2%**
Sampl.	Random	86.9	0.23 m	1.4°	0.10 m	95.8	65.8%
	Floor/wall/ceiling	86.6	0.25 m	1.4°	0.10 m	95.7	66.4%
	Guided	**87.2**	**0.22 m**	**1.3°**	**0.09 m**	**96.1**	**67.2%**
Res.	Low (256 px)	84.2	0.28 m	**1.3°**	0.12 m	95.6	63.3%
	Medium (512 px)	**87.2**	**0.22 m**	**1.3°**	**0.09 m**	**96.1**	**67.2%**
	High (768 px)	85.7	0.26 m	**1.3°**	0.12 m	95.6	66.1%
Init.	Random	60.8	1.28 m	18.0°	0.52 m	60.4	40.0%
	Random + VP	72.2	0.79 m	10.9°	0.32 m	74.6	51.9%
	Y down	84.9	0.27 m	2.1°	0.12 m	93.6	64.7%
	Y down + VP	**87.2**	**0.22 m**	**1.3°**	**0.09 m**	**96.1**	**67.2%**
Scales	Coarse	81.0	0.32 m	1.4°	0.15 m	94.6	36.7%
	Coarse + medium	86.5	0.23 m	1.4°	0.10 m	95.8	65.6%
	Coarse + medium + fine	**87.2**	**0.22 m**	**1.3°**	**0.09 m**	**96.1**	**67.2%**

Table 1. Ablation experiments on our ScanNet++ v2 test set.

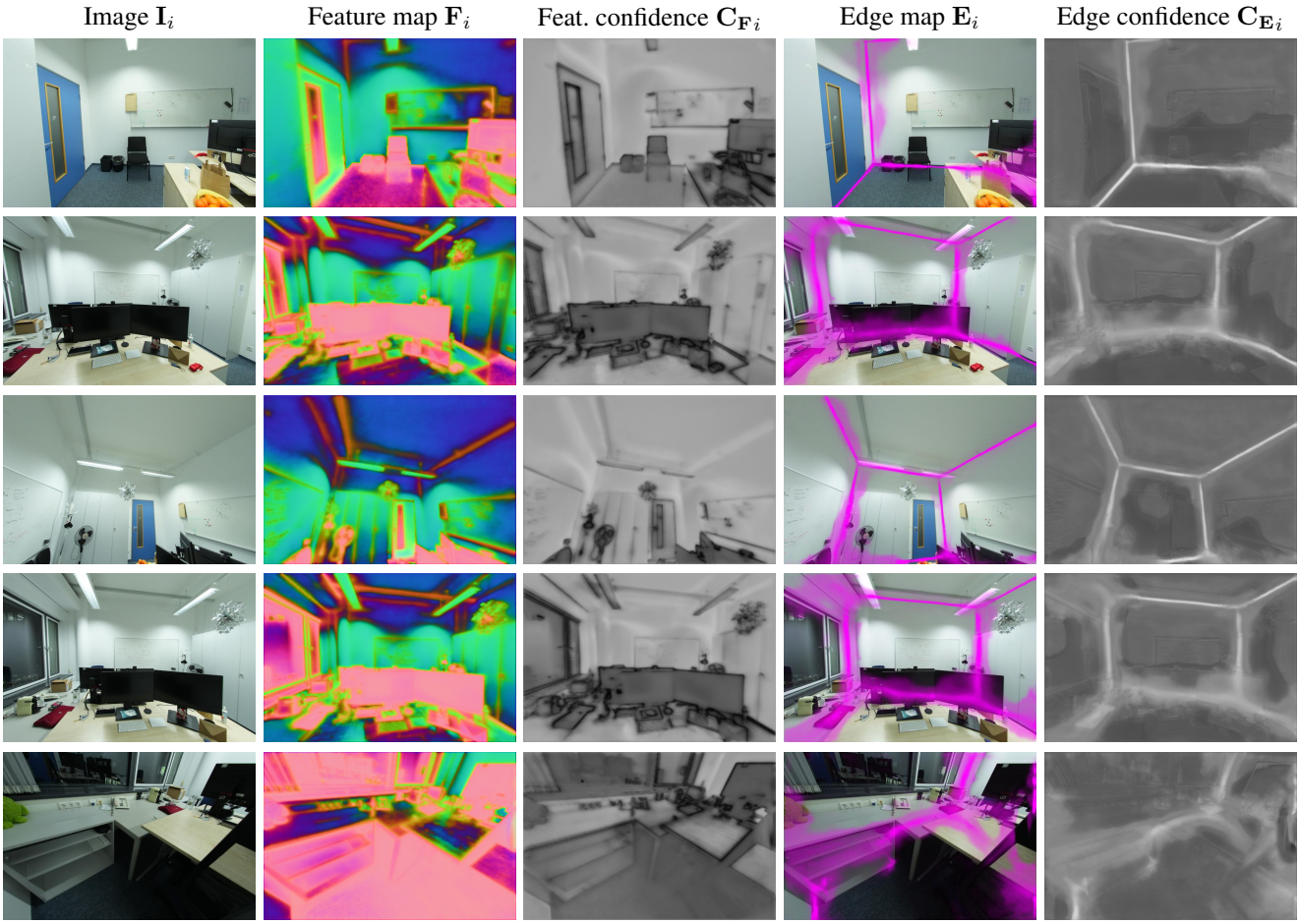

Figure 4. Example feature- and edge maps with corresponding confidence maps for one image tuple in our ScanNet++ v2 test set. Results are shown only on the finest level. Feature maps have been mapped to RGB using PCA. Confidence ranges from low (black) to high (white).