Is Pseudo-Lidar needed for Monocular 3D Object detection? Supplementary Material

Dennis Park* Rareș Ambruș* Vitor Guizilini Jie Li Adrien Gaidon
Toyota Research Institute
firstname.lastname@tri.global

1. Details of training DD3D and PL

We provide the training details used for supervised monocular depth pre-training of both DD3D and PackNet.

DD3D. During pre-training, we use 512 as a batch size, and train for 375K steps until convergence. The learning rate starts at 0.02, decayed by 0.1 at the 305K-th and 365K-th steps. The size of the input images (and projected depth map) is 1600×900, and we resize them to 910×512. When resizing the depth maps, we preserve the sparse depth values by assigning all non-zero depth values to the nearest-neighbor pixel in the resized image space (note that this is different from naive nearest-neighbor interpolation, where the target depth value is assigned zero, if the nearest-neighbor pixel in the original image does not have depth value.) We observed that training converges after 30 epochs. We use the Adam optimizer with $\beta = 0.99$. For all supervised depth pre-training splits, we use an L1 loss between predicted depth and projective ground-truth depth.

When training as 3D detectors, the learning rate starts at 0.002, and is decayed by 0.1, when the training reaches 85% and 95% of the entire duration. We use a batch size of 64, and train for 25K and 120K steps for KITTI-3D and nuScenes, respectively. The μ_l and σ_l are initialized as the mean and standard deviation of the depth of the 3D boxes that are associated with each FPN level, α_l as the stride size of the associated FPN level, and c is fixed to $\frac{1}{500}$. The raw predictions are filtered by non-maxima suppression (NMS) using IoU criteria on 2D bounding boxes. For the nuScenes benchmark, to address duplicated detections in the overlapping frustums of adjacent cameras, an additional BEV-based NMS is applied across all 6 synchronized images (i.e. a *sample*) after converting the detected boxes to the global reference frame.

PackNet. When training PackNet [1], the depth network of PL, we use a batch size of 4 and a learning rate of 5×10^{-5} with input resolution of 640×480. We use only front camera images of DDAD15M to pre-train PackNet, and train until convergence, and for 5 epochs over the KITTI *Eigen-clean* split during fine-tuning. The PL detector is trained with a learning rate of 1×10^{-4} for 100 epochs, decayed by 0.1 after 40 and 80 epochs, respectively. For both networks we use the Adam [4] optimizer with $\beta_1 = 0.9$ and $\beta_2 = 0.999$. Both DD3D and PL are implemented using Pytorch [6] and trained on 8 V100 GPUs.

*equal contribution

2. DD3D architecture details

FPN [3] is composed of a *bottom-up* feed-forward CNN that computes feature maps with a subsampling factor of 2, and a *top-down* network with lateral connections that recovers high-resolution features from low-resolution ones. The FPNs yield 5 levels of feature maps. DLA-34 [11] FPN yields three levels of feature maps (with strides of 8, 16, and 32). We add two lower resolution features (with strides of 64, 128) by applying two 3×3 2D convs with stride of 2 (see Figure **??**). V2-99 [2] by default produces 4 levels of features (strides = 4, 8, 16, and 32), so only one additional conv is used to complete 5 levels feature maps. Note that the final resolution of FPN features derived from DLA-34 and V2-99 network are different, strides=8, 16, 32, 64, 128 for DLA-34, strides= 4, 8, 16, 32, 64 for V2-99.

2D detection head. We closely follow the decoder architecture and loss formulation of [9]. In addition, we adopt the positive instance sampling approach introduced in the updated arXiv version [10]. Specifically, only the center-portion of the ground truth bounding box is used to assign positive samples in $\mathcal{L}_{reg}$ and $\mathcal{L}_{\text{3D}}$.

3. Pseudo-Lidar 3D confidence head

Our PL 3D detector is based on [5], and outputs 3D bounding boxes with 3 heads, separated based on distance (i.e. near, medium and far). Following [8, 7] we modify each head to output a 3D confidence, trained through the 3D bounding box loss. Specifically, each 3D box estimation head consists of 3 fully connected layers with dimensions $512 \longrightarrow 512 \longrightarrow 256 \longrightarrow (\delta, \gamma)$, where δ denotes the bounding box parameters as described in [5], and γ denotes the 3D bounding box confidence.

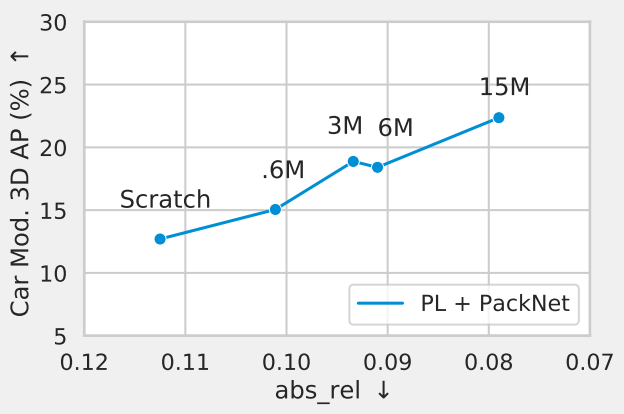

Figure 1: **We evaluate depth performance (abs_rel) against PL 3D detection performance (Car Mod. 3D AP$|_{R_{40}}$) at each pre-training step.** All results are computed on the KITTI-3D *validation* split.

4. The impact of data on Pseudo-Lidar depth and 3D detection accuracy

We evaluate depth quality against the 3D detection accuracy of the PL detector, with results shown in Figure 1. Our results indicate an almost perfect linear relationship between depth quality as measured by the *abs rel* metric and 3D detection accuracy for our PL-based detector.

References

[1] Vitor Guizilini, Rares Ambrus, Sudeep Pillai, Allan Raventos, and Adrien Gaidon. 3d packing for self-supervised monocular depth estimation. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020. 1

[2] Youngwan Lee and Jongyoul Park. Centermask: Real-time anchor-free instance segmentation. 2020. 1

[3] Tsung-Yi Lin, Piotr Dollár, Ross Girshick, Kaiming He, Bharath Hariharan, and Serge Belongie. Feature pyramid networks for object detection. In *CVPR*, 2017. 1

[4] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017. 1

[5] Xinzhu Ma, Shinan Liu, Zhiyi Xia, Hongwen Zhang, Xingyu Zeng, and Wanli Ouyang. Rethinking pseudo-lidar representation. In *European Conference on Computer Vision*, pages 311–327. Springer, 2020. 1

[6] Adam Paszke, Sam Gross, Soumith Chintala, Gregory Chanan, Edward Yang, Zachary DeVito, Zeming Lin, Alban Desmaison, Luca Antiga, and Adam Lerer. Automatic differentiation in pytorch. 2017. 1

[7] Andrea Simonelli, Samuel Rota Bulò, Lorenzo Porzi, Peter Kontschieder, and Elisa Ricci. Demystifying pseudo-lidar for monocular 3d object detection. *arXiv preprint arXiv:2012.05796*, 2020. 1

[8] Andrea Simonelli, Samuel Rota Bulo, Lorenzo Porzi, Manuel López-Antequera, and Peter Kontschieder. Disentangling monocular 3d object detection. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 1991–1999, 2019. 1

[9] Zhi Tian, Chunhua Shen, Hao Chen, and Tong He. Fcos: Fully convolutional one-stage object detection. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 9627–9636, 2019. 1

[10] Tai Wang, Xinge Zhu, Jiangmiao Pang, and Dahua Lin. Fcos3d: Fully convolutional one-stage monocular 3d object detection. *arXiv preprint arXiv:2104.10956*, 2021. 1

[11] Fisher Yu, Dequan Wang, Evan Shelhamer, and Trevor Darrell. Deep layer aggregation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2403–2412, 2018. 1