Self-Supervised Pre-training of 3D Point Cloud Networks with Image Data

Andrej Janda, Brandon Wagstaff, Edwin G. Ng, Jonathan Kelly
Institute for Aerospace Studies
University of Toronto
Canada
<first name>.<last name>@robotics.utias.utoronto.ca

Abstract: Reducing the quantity of annotations required for supervised training is vital when labels are scarce and costly. This reduction is especially important for semantic segmentation tasks involving 3D datasets, which are often significantly smaller, and more challenging to annotate, than their image-based counterparts. Self-supervised pre-training on large unlabelled datasets is one way to reduce the amount of manual annotations needed. Previous work has focused on pre-training with point cloud data exclusively; this approach often requires two or more registered views. In the present work, we combine image and point cloud modalities, by first learning self-supervised image features and then using these features to train a 3D model. By incorporating image data, which is often included in many 3D datasets, our pre-training method only requires a single scan of a scene. We demonstrate that our pre-training approach, despite using single scans, achieves comparable performance to other multi-scan, point cloud-only methods.

Keywords: Self-Supervised Learning, Contrastive Learning, Point Clouds

1 Introduction

Maintaining an accurate environment representation is necessary for many tasks in robotics, such as navigation, planning, obstacle avoidance, and scene understanding. The two most common representations, particularly for scene understanding tasks, are *images*, and *point clouds*. Images contain dense and feature-rich information. However, the lack of depth data in images limits how well they can be used on their own to model 3D environments.

Although point clouds circumvent many of the limitations inherent to images, they are notoriously hard to annotate. The annotation difficulty is a key limiting factor for many state-of-the-art data-driven scene understanding algorithms that require large, annotated datasets [1, 2, 3, 4]. Generating point cloud labels requires expert human annotators to manipulate the point clouds by zooming, panning, and rotating to select points of interest. Annotators then have to separate points that belong to a particular object from the background and other occluded points.

The difficulty of annotating 3D point clouds has resulted in considerable effort and labelling times for existing datasets. For example, the SemanticKITTI dataset [5], which has 518 square tiles of 100 metres length each, required 1,700 hours for annotators to label. ScanNet [6], which has 1,600 reconstructed scenes of indoor rooms, took about 600 hours to label [7]. Despite the substantial labelling time, 3D datasets are still significantly smaller than comparable image-only datasets.

The labelling effort required for 3D data is the reason we seek, herein, to reduce the volume of annotations necessary. Previous work [8, 9, 7] has demonstrated that *self-supervised contrastive pre-training* is an effective approach for improving performance on point cloud understanding tasks with raw unlabelled data. A key limitation of existing 3D pre-training methods is that they neglect the information-rich images that are often available as part of 3D datasets. We propose a pre-training method that leverages images as an additional modality, by learning self-supervised image features that can be used to pre-train a 3D model. Our learning method is split into two stages. The first stage (Stage 1) learns image features using a self-supervised contrastive learning framework. The second

CoRL 2022 Workshop on Pre-training Robot Learning, Auckland, New Zealand.

stage (Stage 2) applies the same contrastive learning framework to pre-train a 3D model by making use of the 2D features learned in Stage 1. By incorporating visual data into the pre-training pipeline, we obtain a notable advantage: only a single point cloud scan and the corresponding image are required during pre-training. The use of a single scan obviates the need for two or more overlapping 3D views, which is required by many point-only approaches. The use of a single scan improves the scalability of our pre-training approach, since we require raw 3D data only, as opposed to multiple scans that have been aggregated using a robust mapping pipeline for data association.

Through extensive experimentation, we compare our pre-training approach with existing point cloud-only approaches on several downstream tasks and across several datasets. We find that our method performs competitively with pre-training methods that use multiple overlapping point cloud scans, despite having access to single scans and images only. In short, this work makes the following contributions:

- we describe a self-supervised method for extracting visual features from images and using them as labels to pre-train 3D models via a contrastive loss;
- we provide visualizations demonstrating that the features follow structures, such as lines and surface patches, that are present in the input image and point cloud;
- we demonstrate that a model trained using features learned from raw images improves performance on 3D segmentation and object detection tasks.

2 Related Work

Our approach builds upon work on self-supervised contrastive learning using images and point clouds. These works typically produce two augmented versions of each data point by applying a series of transformations with two different sets of parameters. Subsequently, a contrastive loss minimizes the distance, specified by a given metric, between model outputs of the augmented *positive* pair while maximizing the distance of the pair to other 'dissimilar' data points, referred to as *negative* samples. Most algorithms typically implement the contrastive loss as an InfoNCE (Info Noise Contrastive Estimation) loss function [10], as defined by Equation (1). Here, we provide a brief summary of algorithms that use this approach to pre-train image and point cloud networks.

Contrastive Learning in 2D. Self-supervised pre-training using image-level features has proven to be a successful approach for downstream image-classification tasks, even achieving comparable performance to supervised pre-training, as demonstrated by SimCLR [11]. Variations of this approach, such as using memory banks [12], momentum encoders [13], stop-gradients [14, 15], clustering [16, 17], and pseudo-labels [18] have also proven effective. For segmentation tasks, self-supervised pre-training of pixel-level features [19, 20, 21] offers improved downstream performance when compared to image-level features.

Contrastive Learning in 3D. The architectures used for pre-training image networks can be adapted to work with point cloud networks. PointContrast [8] is an early example, which augments two overlapping 3D scans with random rotations and colour transformations. Corresponding points between the two transformed scans form a positive pair, while all other points are considered as negative samples in the contrastive loss. Contrastive Scene Contexts (CSC) [9] extends PointContrast with an additional partitioning scheme. Meanwhile, DepthContrast [7] augments a single scan and learns feature vectors at the scan-level instead of at the point-level.

Multi-Modal Contrastive Learning. By projecting 3D points into images, algorithms such as Pri3D [22] and SimIPU [23] leverage 3D data when pre-training models for downstream 2D scene understanding tasks. In both [22] and [23], the pixel-point pairs that map to the same physical 3D location in a common reference frame are used as positive pairs in a (pixel-only) contrastive loss. Alternatively, pre-training with image data can improve downstream performance on 3D scene understanding tasks. CrossPoint [24] applies a contrastive learning objective to global scene features generated from synthetic point clouds of computer-modelled objects and corresponding rendered images, while Superpixel-driven Lidar Representations (SLidR) [25] uses point-pixel matches from outdoor driving datasets. Our method pre-trains a 2D and a 3D model separately in two stages, demonstrating the usefulness of image-features on multiple real-world 3D datasets.

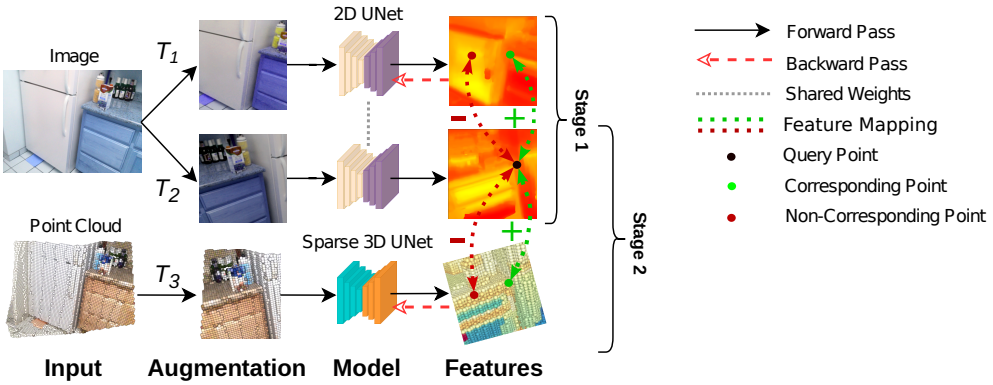


Figure 1: Overview of our multi-modal contrastive learning framework.

3 Method

Our method can be split into two distinct and sequential stages, as shown in Figure 1. The first stage applies a 2D CNN to generate image features at the pixel level, based on a contrastive loss on the individual pixels. The second stage then uses these image features to train a 3D model.

Stage 1. We utilize the ResUNet architecture from Godard at al. [26] to extract 2D pixel-level features and modify the decoder to compute a 16-dimensional feature vector for each pixel in the input image. We pre-load the weights of the encoder from a model trained on the large ImageNet corpus [27]. To pre-train the full model, images are selected from a desired pre-training dataset. We follow roughly the same data augmentation and use the same InfoNCE loss function as SimCLR [11], except that we compare pixel-level features instead of image-level features. Pixels that map back to the same coordinates in the original image are considered as positive samples, while all others, including those from other images in a batch, are considered as negative samples.

Stage 2. In this stage, we pre-train a 3D point-level feature extraction model using pixel-level features from Stage 1. We use the 3D model from [8] and treat the final 1×1 convolution as the decoder, which we initialize from scratch for training on downstream tasks. Each point cloud is also augmented so that the model learns to be invariant to differences in orientation, point density and colour fluctuations. The 2D network is held frozen and the fixed 2D features act as a target for the 3D model to learn. Mapping between 3D points and 2D features is done via perspective projection. Each pixel-point match forms a positive pair $(\mathbf{z}_i, \mathbf{z}_i^+)$, where $\mathbf{z}_i$ and $\mathbf{z}_i^+$ represent the feature vectors of a 3D point and the corresponding pixel respectively. The feature vector of any other point is considered a negative sample and represented as $\mathbf{z}_j^-$. We use an InfoNCE loss function, defined as

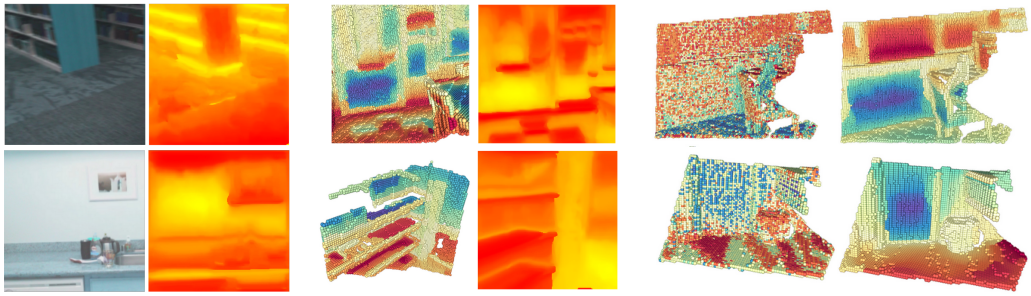
$$\mathcal{L} = -\sum_{i=1}^{N} \log \frac{\exp(\mathbf{z}_i \cdot \mathbf{z}_i^+ / \tau)}{\exp(\mathbf{z}_i \cdot \mathbf{z}_i^+ / \tau) + \sum_j^K \exp(\mathbf{z}_i \cdot \mathbf{z}_j^- / \tau)}, \tag{1}$$

where τ is a smoothing parameter and K represents the number of negative samples. The final parameters of the 3D model are then used on downstream 3D scene understanding tasks.

4 Experimental Results

We compare our method to three point-only pre-training methods: PointContrast [8], Contrastive Scene Contexts (CSC) [9] and DepthContrast [7]. We investigate the performance gain of all pre-training algorithms on three downstream tasks: semantic segmentation, instance segmentation and object detection. These tasks are applied on three datasets of indoor office environments: ScanNet [6], S3DIS [28], SunRGBD [29, 30, 31, 32]; as well as one outdoor driving dataset, SemanticKITTI [5, 33]. Pre-training is performed with the ScanNet dataset and all downstream tasks are run using the pipeline and parameters from CSC [9].

Feature Visualization. We verify visually that the 2D features learned using our 2D pre-training scheme have some connection to the original image by following the approach from [34]. Figure 2a



(a) Input image on the left with corresponding feature visualization on the right.

(b) Point features on the left with corresponding image features on the right.

(c) Point features before (left) and after (right) pre-training with our method.

Figure 2: Feature vector visualizations.

Pre-Training Method		S3DIS Semantic	S3DIS Instance	ScanNet Semantic	ScanNet Instance	ScanNet Object	KITTI Semantic	SUNRGBD Object
	Scratch	65.1	53.0	67.4	49.0	35.2	41.0	32.0
Multi-Scan	PointContrast	66.2 (+1.1)	54.8 (+1.8)	66.9 (-0.5)	49.1 (+0.1)	**36.7 (+1.5)**	42.1 (+1.1)	34.2 (+2.2)
	CSC	**69.0 (+3.9)**	**57.8 (+4.8)**	**67.6 (+0.2)**	**49.3 (+0.3)**	36.1 (+0.9)	**43.0 (+2.0)**	**35.1 (+3.1)**
Single-Scan	DepthContrast	64.9 (-0.2)	52.3 (-0.7)	67.4 (+0.0)	**48.7 (-0.3)**	33.9 (-1.3)	42.0 (+1.0)	32.9 (+0.9)
	Ours	**66.5 (+1.4)**	**55.8 (+2.8)**	**67.7 (+0.3)**	48.5 (-0.5)	**37.7 (+2.5)**	42.0 (+1.0)	**33.1 (+1.1)**

Table 1: Downstream performance comparison of pre-training methods. Semantic segmentation uses the mIOU metric, while both instance segmentation and object detection tasks use the mAP@0.5 metric with a minimum correct overlap ratio of 0.5. The best performance on each task and dataset for single- and multi-view categories are highlighted in bold.

shows a comparison between the heatmap visualizations and the original image. The visualizations show a clear mapping between input image and output features. The heatmaps tend to 'highlight' structures such as lines and surface patches that are present in the input image. Figure 2b shows the relation between the 2D and 3D features. There is a clear mapping between the image and the corresponding point cloud heatmaps. This correlation verifies that the 3D model has indeed learned to 'mimic' the features of the 2D model without relying on the image. Parts of the scene, such as walls and shelves, have consistent features that are present across both modalities. Figure 2c shows the difference between randomly initialized and pre-trained features. Features that are pre-trained follow visible object boundaries.

Downstream Performance. A comparison of the downstream performance with different pre-training methods is shown in Table 1. We find that our algorithm is the best performing of single-scan methods and comparable to multi-scan methods, often outperforming PointContrast. Interestingly, pre-trained weights on the indoor ScanNet dataset were able to improve performance on the outdoor SemanticKITTI dataset. However, those same weights offered no improvement when also fine-tuned on ScanNet segmentation tasks, perhaps due to the lack of new information available between pre-training and fine-tuning. The improvement in ScanNet object detection could be due to new information at the object-level which was not available during point-level pre-training.

5 Conclusion

In this work, we presented a framework for learning dense pixel-level features from raw unlabelled images, which are then used as targets in a contrastive loss to pre-train a 3D model. We visualized these features and found that the 2D features were successfully 'mimicked' by the 3D model. We found that our method was the best performing single-scan method and performed comparably to other multi-scan methods. This result confirms that incorporating visual data into the pre-training procedure is a viable strategy to reduce the requirement for registered point clouds as part of pre-training. By relaxing this requirement, our method scales more readily when compared to other pre-training techniques that require multiple (registered) scans for contrastive learning.

Acknowledgments

If a paper is accepted, the final camera-ready version will (and probably should) include acknowledgments. All acknowledgments go at the end of the paper, including thanks to reviewers who gave useful comments, to colleagues who contributed to the ideas, and to funding agencies and corporate sponsors that provided financial support.

References

[1] K. He, G. Gkioxari, P. Dollar, and R. Girshick. Mask R-CNN. In *International Conference on Computer Vision (ICCV)*, 2017.

[2] C. R. Qi, H. Su, K. Mo, and L. J. Guibas. PointNet: Deep learning on point sets for 3D classification and segmentation. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 77–85, 2017. doi:10.1109/CVPR.2017.16.

[3] L. Jiang, H. Zhao, S. Shi, S. Liu, C.-W. Fu, and J. Jia. PointGroup: Dual-set point grouping for 3D instance segmentation. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4866–4875, 2020. doi:10.1109/CVPR42600.2020.00492.

[4] C. Choy, J. Gwak, and S. Savarese. 4D spatio-temporal convnets: Minkowski convolutional neural networks. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3070–3079, 2019. doi:10.1109/CVPR.2019.00319.

[5] J. Behley, M. Garbade, A. Milioto, J. Quenzel, S. Behnke, C. Stachniss, and J. Gall. SemanticKITTI: A dataset for semantic scene understanding of lidar sequences. In *International Conference on Computer Vision (ICCV)*, pages 9296–9306, 2019. doi:10.1109/ICCV.2019.00939.

[6] A. Dai, A. X. Chang, M. Savva, M. Halber, T. Funkhouser, and M. Nießner. ScanNet: Richly-annotated 3D reconstructions of indoor scenes. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2432–2443, 2017. doi:10.1109/CVPR.2017.261.

[7] Z. Zhang, R. Girdhar, A. Joulin, and I. Misra. Self-supervised pretraining of 3D features on any point-cloud. In *International Conference on Computer Vision (ICCV)*, pages 10232–10243, 2021. doi:10.1109/ICCV48922.2021.01009.

[8] S. Xie, J. Gu, D. Guo, C. R. Qi, L. J. Guibas, and O. Litany. PointContrast: Unsupervised pretraining for 3D point cloud understanding. In *European Conference Computer Vision (ECCV)*, pages 574–591, Cham, 2020. Springer International Publishing. ISBN 978-3-030-58580-8.

[9] J. Hou, B. Graham, M. Nießner, and S. Xie. Exploring data-efficient 3D scene understanding with contrastive scene contexts. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 15582–15592, 2021. doi:10.1109/CVPR46437.2021.01533.

[10] A. v. d. Oord, Y. Li, and O. Vinyals. Representation learning with contrastive predictive coding. arXiv:1807.03748v2 [cs.LG], 2018. URL `https://arxiv.org/abs/1807.03748`.

[11] T. Chen, S. Kornblith, M. Norouzi, and G. Hinton. A simple framework for contrastive learning of visual representations. In *International Conference on Machine Learning (ICML)*, volume 119 of *Proceedings of Machine Learning Research*, pages 1597–1607. PMLR, 2020. URL `https://proceedings.mlr.press/v119/chen20j.html`.

[12] Z. Wu, Y. Xiong, S. X. Yu, and D. Lin. Unsupervised feature learning via non-parametric instance discrimination. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3733–3742, 2018. doi:10.1109/CVPR.2018.00393.

[13] K. He, H. Fan, Y. Wu, S. Xie, and R. Girshick. Momentum contrast for unsupervised visual representation learning. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 9726–9735, 2020. doi:10.1109/CVPR42600.2020.00975.

[14] J.-B. Grill, F. Strub, F. Altché, C. Tallec, P. Richemond, E. Buchatskaya, C. Doersch, B. Avila Pires, Z. Guo, M. Gheshlaghi Azar, B. Piot, k. kavukcuoglu, R. Munos, and M. Valko. Bootstrap your own latent - A new approach to self-supervised learning. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 33, pages 21271–21284. Curran Associates, Inc., 2020.

[15] X. Chen and K. He. Exploring simple siamese representation learning. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 15745–15753, 2021. doi:10.1109/CVPR46437.2021.01549.

[16] M. Caron, I. Misra, J. Mairal, P. Goyal, P. Bojanowski, and A. Joulin. Unsupervised learning of visual features by contrasting cluster assignments. In H. Larochelle, M. Ranzato, R. Hadsell, M. Balcan, and H. Lin, editors, *Advances in Neural Information Processing Systems (NeurIPS)*, volume 33, pages 9912–9924. Curran Associates, Inc., 2020. URL https://proceedings.neurips.cc/paper/2020/file/70feb62b69f16e0238f741fab228fec2-Paper.pdf.

[17] J. Li, P. Zhou, C. Xiong, and S. C. H. Hoi. Prototypical contrastive learning of unsupervised representations. In *International Conference on Learning Representations (ICLR)*. OpenReview.net, 2021. URL https://openreview.net/forum?id=KmykpuSrjcq.

[18] P. Khosla, P. Teterwak, C. Wang, A. Sarna, Y. Tian, P. Isola, A. Maschinot, C. Liu, and D. Krishnan. Supervised contrastive learning. In H. Larochelle, M. Ranzato, R. Hadsell, M. Balcan, and H. Lin, editors, *Advances in Neural Information Processing Systems (NeurIPS)*, volume 33, pages 18661–18673. Curran Associates, Inc., 2020. URL https://proceedings.neurips.cc/paper/2020/file/d89a66c7c80a29b1bdbab0f2a1a94af8-Paper.pdf.

[19] X. Wang, R. Zhang, C. Shen, T. Kong, and L. Li. Dense contrastive learning for self-supervised visual pre-training. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3023–3032, 2021. doi:10.1109/CVPR46437.2021.00304.

[20] Z. Xie, Y. Lin, Z. Zhang, Y. Cao, S. Lin, and H. Hu. Propagate yourself: Exploring pixel-level consistency for unsupervised visual representation learning. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 16679–16688, 2021. doi:10.1109/CVPR46437.2021.01641.

[21] W. Wang, T. Zhou, F. Yu, J. Dai, E. Konukoglu, and L. V. Gool. Exploring cross-image pixel contrast for semantic segmentation. In *International Conference on Computer Vision (ICCV)*, pages 7283–7293, 2021. doi:10.1109/ICCV48922.2021.00721.

[22] J. Hou, S. Xie, B. Graham, A. Dai, and M. Nießner. Pri3D: Can 3D priors help 2D representation learning? In *International Conference on Computer Vision (ICCV)*, pages 5673–5682, 2021. doi:10.1109/ICCV48922.2021.00564.

[23] Z. Li, Z. Chen, A. Li, L. Fang, Q. Jiang, X. Liu, J. Jiang, B. Zhou, and H. Zhao. SimIPU: Simple 2D image and 3D point cloud unsupervised pre-training for spatial-aware visual representations. *Association for the Advancement of Artificial Intelligence Conference (AAAI)*, 36(2): 1500–1508, 2022. doi:10.1609/aaai.v36i2.20040. URL https://ojs.aaai.org/index.php/AAAI/article/view/20040.

[24] M. Afham, I. Dissanayake, D. Dissanayake, A. Dharmasiri, K. Thilakarathna, and R. Rodrigo. CrossPoint: Self-supervised cross-modal contrastive learning for 3D point cloud understanding. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 9902–9912, 2022.

[25] C. Sautier, G. Puy, S. Gidaris, A. Boulch, A. Bursuc, and R. Marlet. Image-to-lidar self-supervised distillation for autonomous driving data. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 9891–9901, 2022.

[26] C. Godard, O. M. Aodha, M. Firman, and G. Brostow. Digging into self-supervised monocular depth estimation. In *International Conference on Computer Vision (ICCV)*, pages 3827–3837, 2019. doi:10.1109/ICCV.2019.00393.

[27] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. ImageNet: A large-scale hierarchical image database. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 248–255, 2009. doi:10.1109/CVPR.2009.5206848.

[28] I. Armeni, O. Sener, A. R. Zamir, H. Jiang, I. Brilakis, M. Fischer, and S. Savarese. 3D semantic parsing of large-scale indoor spaces. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1534–1543, 2016. doi:10.1109/CVPR.2016.170.

[29] S. Song, S. P. Lichtenberg, and J. Xiao. SUNRGBD: A RGBD scene understanding benchmark suite. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 567–576, 2015. doi:10.1109/CVPR.2015.7298655.

[30] A. Janoch, S. Karayev, Y. Jia, J. T. Barron, M. Fritz, K. Saenko, and T. Darrell. A category-level 3-D object dataset: Putting the kinect to work. In *International Conference on Computer Vision (ICCV)*, pages 1168–1174, 2011. doi:10.1109/ICCVW.2011.6130382.

[31] J. Xiao, A. Owens, and A. Torralba. SUN3D: A database of big spaces reconstructed using SfM and object labels. In *International Conference on Computer Vision (ICCV)*, pages 1625–1632, 2013. doi:10.1109/ICCV.2013.458.

[32] N. Silberman, D. Hoiem, P. Kohli, and R. Fergus. Indoor segmentation and support inference from rgbd images. In *European Conference Computer Vision (ECCV)*, pages 746–760, Berlin, Heidelberg, 2012. Springer Berlin Heidelberg. ISBN 978-3-642-33715-4.

[33] A. Geiger, P. Lenz, and R. Urtasun. Are we ready for autonomous driving? The KITTI vision benchmark suite. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3354–3361, 2012. doi:10.1109/CVPR.2012.6248074.

[34] C. Choy, J. Park, and V. Koltun. Fully convolutional geometric features. In *International Conference on Computer Vision (ICCV)*, pages 8957–8965, 2019. doi:10.1109/ICCV.2019.00905.