Supplementary Material for UnScene3D: Unsupervised 3D Instance Segmentation for Indoor Scenes

1. Appendix

1.1. UnScene3D as Data Efficient Pretraining

We report additional qualitative details on the data efficient pretraining performance of UnScene3D in Table 1.

We also note that the 3D contrastive pre-training of CSC, similar to other 3D pre-training methods developed for non-transformer backbones [6, 10, 16, 18], was not beneficial for a transformer-based model. A similar observation was also reported in a recent pretraining method [7]. We thus also compare with CSC pretraining on their original 3D backbone (which demonstrated improvement over training from scratch on the same backbone). Our approach can improves notably over both alternatives.

1.2. The effect of noise robust losses.

We adopt DropLoss [15] for our self-training cycles, which is robust to sparse data and missing annotations. In particular, we use a weighted combination of cross-entropy and Dice [13] losses for bipartite-matching with pseudo annotations. We then drop losses for backpropagation which do not have at least τ_{drop} overlap with the annotations from the previous cycle. We evaluate the effect of different noise robust losses for self-training in Table 2. We compare our baseline losses with a 3D extension of the projection loss of [14], and our adaptation of DropLoss from [15]. Our approach does not penalize for missing pseudo masks, which enables more effective self-training to discover previously missed instances.

1.3. Additional Qualitative Results

We show more qualitative results from our method trained on ARKitScenes [1] in Figure 1 and on ScanNet [4] in Figure 2.

1.4. Pseudo Mask Generation Ablations

We also ablate the saliency threshold, oversegmentation parameters, and separation strategy in our pseudo mask generation. If not explicitly stated otherwise in Table 8, we use both 2D and 3D modality features for the pseudo mask generation.

What is the effect of the saliency threshold in pseudo mask generation? We threshold the saliency matrix A with $\tau_{cut} = 0.55$ for geometric-only features and $\tau_{cut} = 0.65$ for combined modalities. Table 3 shows that our approach maintains robust performance across a large range of τ_{cut} thresholds used to estimate salient areas for pseudo masks. In this table we report results using features from combined modalities, but similar behaviour can be observed for the other scenarios as well.

The effect of iterative mask densification. We designed a strategy to leverage a sparse set of relatively clean initial pseudo masks, which are progressively extended with confident self-predictions during later iterations. This leads to a 3x improvement over state of the art in the Average Precision Metric. We could also consider different mask refinement strategies using a mixture of segments, initial masks or self-trained instances. Tab. 4 ablates a mask refinement strategy of discarding previous masks and retaining current predictions. We also consider using Felzenswalb segments directly instead of feature-based pseudo labels. Both these strategies lead to lower performance due to the increased presence of noisy labels, which dominate the training signal.

Robustness to oversegmentation parameters. Table 5 shows that our approach maintains strong robustness to a wide range of oversegmentation parameters for our geometric segments (our used parameters denoted in bold).

Additional pseudo mask generation hyperparameters. Additionally, we also test the effect of other hyperparameters in out *NCut*-based pseudo mask generation module, including used distance metrics in the similarity matrix and different methods to separate unconnected patches in the predicted foregrounds. During the foreground separation in the Normalized Cut algorithm, we had an additional condition for the minimum number of foreground segments for the bipartitions. This conditions was able effectively filter out suboptimal partitioning of the full graph leading to separated parts from the full instances. Reducing the size of this parameter can directly lead to a more dense set of initial

Figure 1. Additional results on the ARKitScenes dataset [1], compared to geometric clustering and oversegmentation-based baselines.

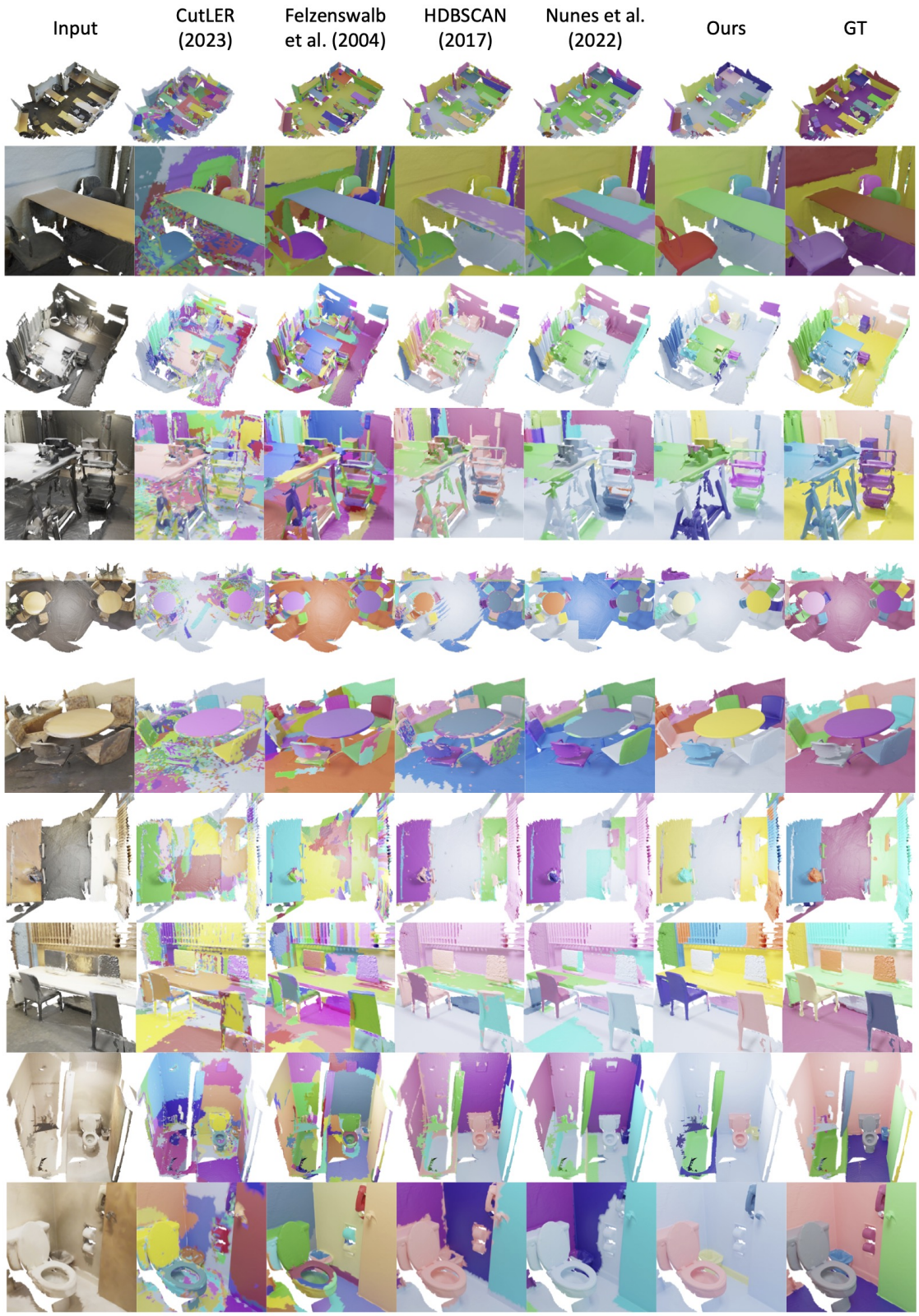

Figure 2. Additional results on the ScanNet dataset [4], compared to geometric clustering and oversegmentation-based baselines.

Model	Backbone	1%			5%			10%			20%			50%		
		AP@25	AP@50	AP	AP@25	AP@50	AP	AP@25	AP@50	AP	AP@25	AP@50	AP	AP@25	AP@50	AP
Scratch	Bottom-up	22.6	14.1	6.8	45.5	33.3	18.1	54.8	39.2	21.9	61.0	43.4	25.5	67.0	51.4	30.3
CSC [6]	Bottom-up	35.6	22.1	12.5	52.7	39.9	23.3	59.8	43.8	25.0	63.8	48.9	29.6	70.5	56.0	33.6
Scratch	Transformer	24.7	9.3	4.6	48.1	27.6	16.3	59.2	39.1	23.4	66.4	49.6	33.1	**78.9**	67.5	**49.8**
CSC	Transformer	17.0	6.8	3.8	44.2	22.7	13.1	55.2	32.3	19.1	62.0	41.2	26.0	73.7	58.2	40.0
Ours	Transformer	**43.5**	**28.4**	**15.8**	**63.2**	**46.8**	**28.3**	**70.3**	**55.7**	**36.7**	**72.4**	**60.7**	**41.5**	**78.9**	**68.0**	48.2

Table 1. Unsupervised class-agnostic pretraining with our method can also act as a powerful pretraining strategy, advancing over state of the art. We report pretraining with CSC [6] and UnScene3D, and evaluate the downstream weakly-supervised instance segmentation performance on ScanNet with percentage of limited annoated scenes used denoted in the top row. As we found that CSC degraded performance when using a transformer-based backbone, we also report the performance of training from scratch and CSC on their originally proposed backbone of a sparse UNet with bottom-up voting.

	AP@25	AP@50	AP	AP Final
Initial Pseudo Masks	19.9	10.0	5.9	-
Baseline losses [12]	42.3	16.9	7.2	14.2
Projection loss [14]	35.7	12.1	4.7	7.2
DropLoss [15]	**52.9**	**23.2**	**10.4**	**15.9**

Table 2. A 3D projection loss struggles with under-determined associations, while DropLoss helps UnScene3D to discover parts of the scene that were missed by the source supervision. We report all metrics after a single iteration and the AP scores after 4 iterations of self-training.

τ_{cut}	AP@25	AP@50	AP
0.40	16.7	9.0	5.2
0.50	20.8	10.7	5.7
0.55	21.0	10.8	5.7
0.60	21.3	11.3	5.8
0.65	**19.9**	**10.0**	**5.9**
0.70	18.2	9.9	5.6
0.80	11.8	5.0	2.6

Table 3. Our pseudo mask generation quality, as measured by AP metrics, maintains robustness to a large range of τ thresholds that extract saliency. Note that this measures the quality of only the pseudo masks; our full approach with self-training produces significantly improved results. In this table we show results and parameters used by our method in bold and report pseudo mask performance generated from both modalities.

pseudo masks, with the cost of higher false positive rate. In Table 5 we report a sparser and denser version of the datasets with a minimum number of foregorund segments of 8 and 2 accordingly, and show the initial higher scores of the pseudo annotation doesn't necessarily propagate to better downstream self-trained performance.

Finally, we also ablate the effect of our physical connectivity-based foreground separation introduced in Section 3.1. In our main method we separate all set of connected components in the foreground, but only keep the component with the highest eigenvector activation (*Max*).

	AP@25	AP@50	AP
Felzenswalb Masks	35.5	20.6	10.3
Mask Refinement	43.7	24.4	12.4
Mask Addition (Ours)	58.6	32.0	16.0

Table 4. Instead of using masks from previous iteration directly it is the best to keep the initial masks fixed, and iteratively sample plausible predictions to enrich the pseudo dataset during self-training. This method strikes a balance between relatively clean, but sparse labels and increasing number of confident samples. Finally, even though Felzenswalb oversegmentation yields to higher precision, then our initial mask prediction algorithm, it also includes more background into the training, and this way plateauing at a lower self-training performance.

As an alternative we also test a method where we calculate the highest average activation in the connected component (*Avg.*), a method where we keep the component with the largest surface value (*Largest*) and finally, to test the effect of this module, without any kind of connectivity-based separation (*No Sep.*).

1.5. Comparison with methods from the 2D domain

To ensure a fair evaluation of methods operating on different input domains in Table 1. we followed the established procedure of well-known baselines [3, 5, 8]. This involves using depth information to project 2D predictions into 3D such that all methods are evaluated in the same 3D domain and aggregate multiple predictions through consensus by majority voting or accepting the maximum confidence scores for every voxel location. We also show results evaluated against 2D ScanNet images by projecting our method's predictions into 2D in Tab. 6, and comparing it to the current state of the art 2D unsupervised segmentation method [15] which demonstrates the usefulness of 3D reasoning.

We also compare to weakly-supervised instance segmentation method SAM3D [17], where powerful class-agnostic 2D masks are extracted by the powerful SAM model [9]. Here the projected 2D masks are merged into 3D masks iteratively with a bottom-up bidirectional merging approach to achieved cleaner and more view-independent 3D instances.

Generation Params.				Initial Pseudo Mask				1 Iteration of Self-Training			4 Iterations of Self-Training		
Segment Size	Metric	Separation	Min. # of Foreground	# of Instances	AP@25	AP@50	AP	AP@25	AP@50	AP	AP@25	AP@50	AP
30	Cos	Max	8	2169	21.9	11.5	6.3	53.7	26.2	12.4	55.4	30.3	15.3
50	**Cos**	**Max**	**8**	1414	**19.9**	**10.0**	**5.9**	**52.9**	**23.2**	**10.4**	**58.5**	**32.2**	**15.9**
100	Cos	Max	8	1090	17.4	8.0	4.2	33.1	10.2	3.9	39.6	13.7	5.3
200	Cos	Max	8	584	11.0	3.7	1.8	24.3	8.7	2.1	26.1	9.7	2.4
400	Cos	Max	8	319	6.4	2.5	1.1	19.1	3.9	1.2	19.9	3.2	1.0
50	L2	Max	8	1539	20.1	10.6	5.4	49.0	21.7	9.8	55.3	38.4	14.3
100	L2	Max	8	805	13.3	5.3	2.6	30.8	8.3	2.8	39.0	12.7	5.0
50	Cos	No Sep.	8	125	4.3	0.3	0.1	4.3	0.5	0.2	4.9	0.6	0.2
50	Cos	Largest	8	620	11.5	4.9	2.5	11.5	1.5	0.4	12.9	2.2	12.9
50	Cos	Avg.	8	1078	16.8	9.1	5.1	36.4	12.5	4.9	43.8	17.8	7.5
30	Cos	Max	2	2909	29.0	15.6	8.7	53.6	28.6	14.2	54.2	29.8	15.4
50	Cos	Max	2	2512	24.9	12.4	7.2	56.5	29.8	15.0	51.3	26.2	12.6
100	Cos	Max	2	2317	23.1	12.3	6.8	51.8	24.4	11.6	57.1	31.3	15.6
200	Cos	Max	2	2181	28.4	15.5	8.9	54.6	28.7	13.7	56.6	31.4	15.6
400	Cos	Max	2	1373	20.6	11.1	6.3	51.0	24.8	11.8	55.8	30.3	15.2
50	L2	Max	2	2496	28.6	15.8	9.0	55.8	29.6	14.6	54.8	30.3	15.3
100	L2	Max	2	1668	23.4	12.7	7.3	53.1	25.0	11.3	56.3	27.7	12.9
50	Cos	No Sep.	2	159	0.2	0.5	3.6	5.4	0.6	0.3	3.9	0.4	0.2
50	Cos	Largest	2	1026	14.1	7.2	3.9	11.5	1.8	0.5	14.5	2.5	0.7
50	Cos	Avg.	2	2053	23.3	12.0	6.8	52.5	27.4	12.7	54.9	29.9	14.9

Table 5. We denote the parameters used by our method in bold. We show that our method is robust to a wide range of numbers regarding segments sizes and different similarity metrics, and only degrades somewhat in performance when segments are constrained to be too large. We also show that the separation of physically distant foreground patches is important and it is beneficial to use the activation of the eigenvector for the best results. Finally, we show that denser initial mask predictions lead to quantitatively better initial pseudo annotations, and even better self-training performance after a single iteration, but underperforming in their final scores. This behaviour can be explained by the larger false positive ratio in the denser initial predictions, which is propagating through all iterations, but thanks to the noise robust losses and iterative refinement of predictions the sparse set of labels can be effectively used. In this table we report results using both modalities for the initial pseudo mask generation, and number predicted pseudo instances in the official validation split of the ScanNet dataset.

	AP@25 (2D)	AP@50 (2D)	AP (2D)
CutLER (2D)	7.8	2.8	0.7
Ours (projected)	60.0	38.1	21.1

Table 6. 2D evaluation on ScanNet images.

A qualitative comparison on ScanNet can be seen in Table 7, with qualitative comparisons in Figure 3.

	AP@25	AP@50	AP
SAM3D	37.2	11.8	3.7
SAM3D with GT Segments	47.6	24.1	10.8
Ours	**58.5**	**32.2**	**15.9**

Table 7. UnScene3D achieves significantly better performance on ScanNet than SAM3D through our strong multi-modal reasoning.

Figure 3. While SAM has powerful capabilities in crisp 2D mask generation, when aggregated on 3D, SAM3D tends to over-segment object instances.

SAM3D must resolve view inconsistencies and SAM's tendency to over-segment objects, which results in SAM3D splitting instances, while UnScene3D is able to achieve complete masks through multi-modal reasoning. We believe integrating SAM or other (weakly-) supervised 2D models into our pipeline to enable multi-modal reasoning is an interesting avenue for future work.

1.6. Additional Implementation Details

Here, we further explain the implementation details of our pseudo mask generation.

Pseudo code for masked NCut We show the pseudo code-style implementation for the masked normalized cut algorithm generating multiple instances as pseudo masks. The full algorithm can be seen in 1.

3D Adaptation of FreeMask We also evaluate an alternative pseudo mask segmentation algorithm besides the masked *NCut* method. In the 2D domain FreeSOLO [14] also followed a two stage pipeline first generating the pseudo annotations, and then refine those predictions through a series of self-training cycles. We followed their intuition to take a self-supervised pretrained backbone and extract it's deep features at multiple levels of the decoder.

Algorithm 1: Masked NCut on 3D segments

Data: $\mathcal{S} = \{s_i, \dots, s_N\}$, $\mathcal{F} \in \mathcal{R}^{NxD}$,
$\mathcal{C} = \{(s_1, s_k), (s_1, s_l), \dots\}$
Result: $\mathcal{M} = \{m_j, \dots, m_M\}$

$\mathcal{M} \leftarrow \{\}$
while $j \leq max\_inst\_num$ **do**
  $\mathcal{F}' \leftarrow \mathcal{F}$
  $\mathcal{F}'[\mathcal{M}] \leftarrow 0.$ `// Mask out previous insts.`
  $\mathcal{W} \leftarrow \mathcal{F} \times \mathcal{F}^T$ `// Feature similarity`
  `// Saliency with connected graph`
  $\mathcal{W}_{i,k} = \begin{cases} 1. & \text{if } \mathcal{W}_{i,k} \geq \tau_{cut} \\ \epsilon & \text{if } \mathcal{W}_{i,k} < \tau_{cut} \end{cases}$
  $\mathcal{D}_{i,i} = \sum_k W_{i,k}$
  `// Get` 2^{nd} `smallest eigenvector`
  $\lambda, \mathbf{v} \leftarrow eigh(\mathcal{D} - \mathcal{W}, \mathcal{D}, -2)$
  $m_i = \begin{cases} 1 & \text{if } v_i \geq mean(\mathbf{v}) \\ 0 & \text{if } v_i < mean(\mathbf{v}) \end{cases}$
  `// Invert bipartition if too large`
  if $sum(\mathbf{m}) > D/2$ **then**
    $\mathbf{m} = 1 - \mathbf{m}$
    $\mathbf{v} = -1. * \mathbf{v}$
  `// Separate unconnected components`
  $v_{max} = max(\mathbf{v})$
  $\tilde{\mathbf{m}} = sep(\mathbf{v}, v_{max}, \mathcal{C})$
  $M \leftarrow M \cup \{\tilde{\mathbf{m}}\}$

While in standard pretrained UNet-style models early features represent global context, final features and local semantic meaning, intermediate features can act as an useful proxy to extract self-similar regions in the input samples. In our implementation we used the same backbone features of [2, 6] for the same 2D-3D setup and extracted the penultimate layer features for the self-similarity calculation. Then sampled the feature space with the Furthest Point Sampling [11] strategy to get a more limited set of anchor points, later used to extract self-similar regions. For every seed point we took similarity scores with the other features of the full scene and thresholded it to extract salient regions. Finally, we used the efficient Non Maximum Suppression implementation from [14] to sort the predicted salient areas and filter out overlapping regions. We also used average similarity score combined with the salient region area to get *maskness scores* for every salient region, directly following the original implementation. We report comparative results of the masked *NCut* algorithm and our FreeMask 3D adaptation after self-training in Table 3. of the main paper and in Table 8 of the initial pseudo mask scores.

We also note here that while there is a difference in the initial pseudo mask qualities for the different methods, the downstream performance is way more significant. This can explained by the nature of the pseudo masks. *NCut* provides a clean and sparse set of annotation, which is easy to densify for following iterations. On the other hand, the more dense, but noisy FreeMask predictions remain in the training for the duration of the whole training, hindering the performance of the self-trained model with noisy supervision.

	Modality	AP@25	AP@50	AP
FreeMask	3D	13.7	7,2	3.7
Ours	3D	13.8	4.7	2.0
FreeMask	2D	15.3	6.6	2.9
Ours	2D	15.6	7.2	3.6
FreeMask	both	17.9	7.5	3.7
Ours	both	19.9	10.0	5.9

Table 8. We compare pseudo mask generation from 3D-only features (3D), color-only features (2D), and both color and geometry (both) signal, as well as with pseudo annotation generation algorithm FreeMask. We compare the quality of the initial pseudo mask dataset using our masked *NCut* algorithm and the adaptation of FreeMask [14] to 3D. We see that the normalized cut-based method is superior for both modalities.

References

[1] Gilad Baruch, Zhuoyuan Chen, Afshin Dehghan, Tal Dimry, Yuri Feigin, Peter Fu, Thomas Gebauer, Brandon Joffe, Daniel Kurz, Arik Schwartz, and Elad Shulman. ARKitscenes - a diverse real-world dataset for 3d indoor scene understanding using mobile RGB-d data. In *Thirty-fifth Conference on Neural Information Processing Systems Datasets and Benchmarks Track (Round 1)*, 2021. 1, 2

[2] Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging properties in self-supervised vision transformers. In *Proceedings of the International Conference on Computer Vision (ICCV)*, 2021. 6

[3] Angela Dai and Matthias Nießner. 3dmv: Joint 3d-multi-view prediction for 3d semantic scene segmentation. In *European Conference on Computer Vision*, 2018. 4

[4] Angela Dai, Angel X. Chang, Manolis Savva, Maciej Halber, Thomas Funkhouser, and Matthias Nießner. Scannet: Richly-annotated 3d reconstructions of indoor scenes. In *Proc. Computer Vision and Pattern Recognition (CVPR), IEEE*, 2017. 1, 3

[5] Ji Hou, Angela Dai, and Matthias Nießner. 3d-sis: 3d semantic instance segmentation of rgb-d scans. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 4421–4430, 2019. 4

[6] Ji Hou, Benjamin Graham, Matthias Nießner, and Saining Xie. Exploring data-efficient 3d scene understanding with contrastive scene contexts. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15587–15597, 2021. 1, 4, 6

[7] Ji Hou, Xiaoliang Dai, Zijian He, Angela Dai, and Matthias Nießner. Mask3d: Pre-training 2d vision transformers by

learning masked 3d priors. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13510–13519, 2023. 1

[8] Maximilian Jaritz, Jiayuan Gu, and Hao Su. Multi-view pointnet for 3d scene understanding. In *ICCV Workshop 2019*, 2019. 4

[9] Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C Berg, Wan-Yen Lo, et al. Segment anything. *arXiv preprint arXiv:2304.02643*, 2023. 4

[10] Lucas Nunes, Rodrigo Marcuzzi, Xieyuanli Chen, Jens Behley, and Cyrill Stachniss. Segcontrast: 3d point cloud feature representation learning through self-supervised segment discrimination. *IEEE Robotics and Automation Letters*, 7(2):2116–2123, 2022. 1

[11] Charles Ruizhongtai Qi, Li Yi, Hao Su, and Leonidas J Guibas. Pointnet++: Deep hierarchical feature learning on point sets in a metric space. *Advances in neural information processing systems*, 30, 2017. 6

[12] Jonas Schult, Francis Engelmann, Alexander Hermans, Or Litany, Siyu Tang, and Bastian Leibe. Mask3D for 3D Semantic Instance Segmentation. In *International Conference on Robotics and Automation (ICRA)*, 2023. 4

[13] Carole H Sudre, Wenqi Li, Tom Vercauteren, Sebastien Ourselin, and M Jorge Cardoso. Generalised dice overlap as a deep learning loss function for highly unbalanced segmentations. In *Deep Learning in Medical Image Analysis and Multimodal Learning for Clinical Decision Support: Third International Workshop, DLMIA 2017, and 7th International Workshop, ML-CDS 2017, Held in Conjunction with MICCAI 2017, Québec City, QC, Canada, September 14, Proceedings 3*, pages 240–248. Springer, 2017. 1

[14] Xinlong Wang, Zhiding Yu, Shalini De Mello, Jan Kautz, Anima Anandkumar, Chunhua Shen, and Jose M Alvarez. Freesolo: Learning to segment objects without annotations. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14176–14186, 2022. 1, 4, 5, 6

[15] Xudong Wang, Rohit Girdhar, Stella X Yu, and Ishan Misra. Cut and learn for unsupervised object detection and instance segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3124–3134, 2023. 1, 4

[16] Saining Xie, Jiatao Gu, Demi Guo, Charles R Qi, Leonidas Guibas, and Or Litany. Pointcontrast: Unsupervised pretraining for 3d point cloud understanding. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part III 16*, pages 574–591. Springer, 2020. 1

[17] Yunhan Yang, Xiaoyang Wu, Tong He, Hengshuang Zhao, and Xihui Liu. Sam3d: Segment anything in 3d scenes. *arXiv preprint arXiv:2306.03908*, 2023. 4

[18] Zaiwei Zhang, Rohit Girdhar, Armand Joulin, and Ishan Misra. Self-supervised pretraining of 3d features on any point-cloud. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 10252–10263, 2021. 1