LABELMAKER3D: Automatic Semantic Label Generation from RGB-D Trajectories

Silvan Weder[1] Hermann Blum[1] Francis Engelmann[1,2,3] Marc Pollefeys[1]

[1]ETH Zurich [2]ETH AI Center [3]Google

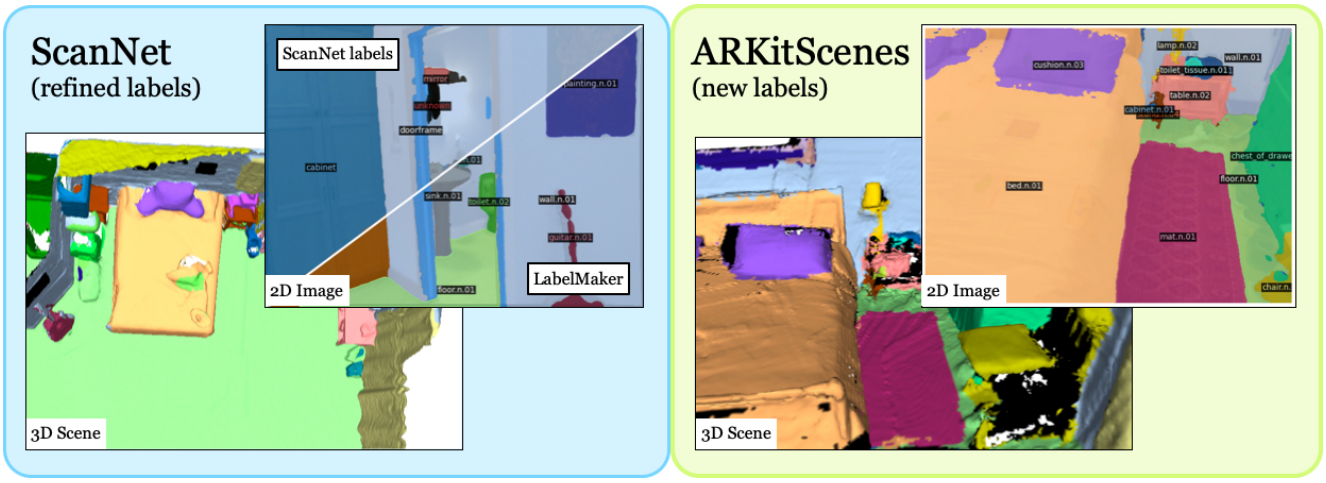

Figure 1. LabelMaker3D bundles a collection of state-of-the-art segmentation models with different sets of predicted classes in a neural field. LabelMaker3D can refine existing annotations and produce highly accurate 2D as well as 3D labels on ScanNet *(left)*. At the same time, it opens new possibilities to rapidly label large-scale datasets without human effort such as ARKitScenes *(Right)*.

Abstract

Semantic annotations are indispensable to train or evaluate perception models, yet very costly to acquire. This work introduces a fully automated 2D/3D labeling framework that, without any human intervention, can generate labels for RGB-D scans at equal (or better) level of accuracy than comparable manually annotated datasets such as ScanNet. Our approach is based on an ensemble of state-of-the-art segmentation models and 3D lifting through neural rendering. We demonstrate the effectiveness of our LabelMaker3D pipeline by generating significantly better labels for the ScanNet datasets and automatically labelling the previously unlabeled ARKitScenes dataset. Code and models are available at http://labelmaker3d.github.io.

1. Introduction

Semantic perception of the world around us is of central importance for many computer vision applications [23, 28, 33]. Without semantic perception, meaningful interactions with our environment are hardly possible. Thus, semantic scene perception has been a long-standing problem in computer vision and robotics [6, 12, 19, 23]. In recent years, most solutions have converged towards using deep neural networks. However, training and evaluating these networks is hard. As recent works such as SAM [10], language-based models [16, 20, 27], or InternImage [30] have shown, huge quantities of training data, orders of magnitude larger than any single existing research dataset, are necessary to achieve good generalization. On the other hand, generalization is necessary because the distribution of the deployment environment - *e.g.*, a particular user's home, in which a robotic application is to be deployed - is outside of the distribution of existing annotated training datasets. To evaluate generalization in or adapt to specific deployment environments, labeled data of these environments is necessary. From both training and deployment perspectives, the availability of labeled data is therefore a key problem. Unfortunately, the acquisition of this data is usually very expensive as semantic ground-truth annotation is a time-consuming manual process.

In this work, we particularly focus on 3D semantic segmentation. The available scale of 3D semantic segmentation data such as ScanNet [8] or Matterport3D [4] is far below the scale of 2D semantic segmentation datasets like ADE20k [38], COCO-stuff [3], or others [7, 25, 31]. Even tough tasks such as semantic segmentation or online semantic reconstruction gain maturity and are crucial for interactive applications, there is even less semantic data with paired camera trajectories and corresponding scene recon-

structions. ScanNet [8] is by far the largest in this domain with an abundance of scenes and a well-established benchmark. However, both camera images and labels are oftentimes noisy, making it hard to generalize from ScanNet to other datasets. ARKitScenes [1] shows the growing possibility to capture RGB-D trajectories at scale, and at the same time illustrates the cost of semantic annotations, featuring an incomplete list of bounding boxes.

To push the scale and accuracy of 3D semantic segmentation datasets, we present *LabelMaker3D*. LabelMaker3D automatically creates labels that are on the same level of accuracy as the established ScanNet benchmark, but without any human annotation. Further, we show that it can produce better labels than the original ScanNet labels when using the human annotations as an additional input.

The design of our method is motivated by two observations. The first observation is on recent advances in 2D semantic segmentation, where a leap in training data scale through combination of different tasks and datasets [30] or visual-language models [16] has boosted generalization. The second observation is in the field of neural radiance fields, where [17, 24, 36] have shown that NeRFs can be used to denoise semantic input labels and learn a multi-view consistent semantic label field. We leverage these two observations and motivate an automatic labelling pipeline with two main components at its heart. First, we leverage large 2D models, that combine the power of different tasks and input modalities, in order to predict different hypothesis for labels in 2D. These labels are aggregated using our consensus voter in order to obtain a single 2D prediction for every frame. Second, all 2D predictions are aggregated and made consistent using a neural radiance field. This neural radiance field can be used to render clean and consistent 2D label maps. Alternatively, the labels can be aggregated and mapped into 3D to obtain labeled pointclouds or meshes.

With a comparison to SOTA methods and datasets and an extensive ablation study, we showcase that our method automatically generates labels of similar quality than human annotators. We also demonstrate fully automatic labelling for ARKitScenes, for which no dense labels exist to date.

In summary, our contributions are:

- A curated mapping between the indoor label sets NYU40, ADE20k, ScanNet, Replica, and into the wordnet graph.
- A pipeline to automatically label RGB-D trajectories, as well as corresponding 3D point clouds, that achieves higher quality than the original labels of ScanNet.
- Generated labels in 3D meshes and 2D images for ScanNet [8] and ARKitScenes [1].

2. Related Work

Labelling in 2D. Cityscapes [7] is one of the most established 2D semantic segmentation datasets. The authors report an effort of more than 1.5h to annotate a single frame. Similar frame-by-frame manual annotations were provided in NYU Depth [25], ADE20k [38], or COCO-stuff [3]. While frame-by-frame annotations yield very high quality segmentation masks, they are expensive to obtain. Although the effort can be reduced through comfortable annotation tools [2, 13], it cannot be avoided that a human inspects every image and performs at least a couple of clicks.

Labelling in 3D. If scenes are annotated in 3D, their annotations can easily be rendered into any localized camera image in the same scene, therefore potentially reducing labeling effort. This approach was followed in Replica [26] and ScanNet [8]. iLabel [37] pioneered to use NeRFs for this type of rendering, additionally showing that NeRFs have an intrinsic capability to segment whole objects along texture boundaries from a few clicks. Similarly, [11, 34] also reduce the manual labelling effort to a few positive and negative clicks per object. Matterport [4] consists of large labeled 3D scans, but does not have corresponding 2D images and therefore can only be used for 3D methods.

Pretrained Models. It is a well-established approach in labelling to label parts of a dataset, train a model on that part, and use its predictions to bootstrap labels for the rest of the data. More recently, models pretrained on large amounts of data have been introduced to help labelling completely unseen datasets. SAM [10] showed impressive results of segmenting objects in images from close to zero clicks where only labels have to be assigned. The seconds step can even be bootstrapped through CLIP [21]. CLIP2Scene [5] takes a similar approach in 3D to train a pointcloud classifier on previously unlabeled data.

3. Method

We briefly discuss the relabelling of ScanNet scenes. Then, we discuss the translation between prediction spaces. Finally, we present our automatic labelling pipeline.

3.1. Relabeling ScanNet Scenes

To be able to evaluate the quality of LabelMaker3D, we want to compare it against existing human annotations. We choose the ScanNet dataset because its scale has a large potential for automatic processing. To be able to evaluate the quality of the existing labels and compare them with LabelMaker3D, we create high-quality annotations for a selection of scenes.

The original ScanNet [8] labels were created using free text user prompts. They consequently have duplicates or are ill-defined. This reflects the open-world approach of Dai et al. [8], but contradicts the use as benchmark labels, for which they them map to other class sets. As a set of annotation classes, we therefore did not directly annotate with

ScanNet classes, but use wordnet [18] synkeys[1]. In particular, we start from the mapping that ScanNet defined between their labels and wordnet and take the categories that occur at least three times in the dataset. This yields an initial list of 199 categories, already resolving many ambiguities. We then check the definitions of all of these categories in the wordnet database and correct the initial mapping, as well as merged categories that are still too ambiguous by their definitions in wordnet (e.g. `rug.n.01` "rug, carpet, carpeting; floor covering consisting of a piece of thick heavy fabric (usually with nap or pile)" and `mat.n.01` "a thick flat pad used as a floor covering"). The result are 186 categories that come with a text definition, a defined hierarchy, and all possible synonyms that describe the category.

We then annotate our selected ScanNet scenes with these 186 categories based on their wordnet definitions. We use [11] to annotate the fine meshes of the scenes with a minimum number of necessary clicks. Only the authors of this paper provided annotations, and each annotation was cross-checked by at least one other author. In case of doubt, individual objects were discussed together. On average, labeling of a scene took 5 hours.

3.2. Translation between Prediction Spaces

We employ different predictors that were trained on different data sets with different numbers and definitions of classes. This requires translating between different prediction spaces. We therefore build a mapping between the class definitions of NYU40, ADE20k, ScanNet20, ScanNet200, Replica, and the WordNet semantic language graph.

In this effort, we build on top of previous work, as the original ScanNet [8] already defined a mapping between ScanNet classes, NYU40 classes, Eigen13 classes, and wordnet synkeys [18]. Further, Lambert et al. [14] curated mappings between the taxonomies of semantic segmentation datasets, out of which mappings between NYU40, SUNRGBD, and ADE20k are most relevant for indoor perception. We took the union of both works as initial mapping, but found that many corrections were needed especially with regard to wordnet synkeys and many ADE20k were missing because [14] only considered 20 NYU categories. We then further added mappings to the Replica categories for the purpose of evaluation, since Replica is one of the most accurately annotated indoor semantic datasets.

When mapping between two class spaces, for any class in the source space there are three cases in the target space: *a)* there is no corresponding class in the target space, *b)* there is exactly one corresponding class in the target space. This may be an exact match, or a class to which multiple class ids from the source space are matched (*e.g.*, the source space may distinguish between office chair, chair, and stool but the target space just has one general chair class), *c)* there are multiple corresponding classes in the target space because the target space has a higher resolution than the source space (*e.g.*, a general chair class in the source space can be split up in the target space to distinguish between office chair, chair, or stool).

For (a) and (b), mappings are straightforward. We resolve (c) dependent on the use cases:

- *Evaluating a class with multiple correspondences.* A label of any of the correspondences is treated as a true positive. If none of the correspondences is the true class, all of them are counted as false positives.
- *Computing model consensus.* Predictions in the source space vote for all possible correspondences in the target space. The ambiguity between the possible correspondences is usually resolved through an additional predictor with a prediction space of higher resolution. If no resolution is achieved, we pick the first of the possible classes.

3.3. Base Models

We employ an ensemble of strong base models, each state-of-the-art in their respective task and data characteristic:

InternImage [30] is a supervised 2D RGB-only semantic segmentation model that at the time of writing has state-of-the-art performance on the Cityscapes and ADE20k benchmarks. It achieves this by performing large-scale joined pretraining on most available visual classification datasets. We use the ADE20k fine-tuned variant.

OVSeg [15] is an open-vocabulary semantic segmentation model based on CLIP [21], a visual-language representation model. OVSeg segments images by assigning region proposals to a set of given prompts and is therefore not limited to a fixed set of classes. In particular, we added such an open-vocabulary segmentation model not because they achieve the best performance on a given task but because of their generalization ability. We generate prompts from our set of wordnet synkeys by averaging over language prompts such as "A _ in a room.", but also using all possible synonyms according to wordnet.

CMX [35] is at the time of writing the state-of-the-art 2D semantic segmentation model for NYU Depth v2, a RGB+Depth indoor dataset. Its predictions also take the geometric cues from the depth into account.

Mask3D [23] is at the time of writing the state-of-the-art 3D instance segmentation model on ScanNet200 [22]. This method operates on an accumulated pointcloud of a scene instead of frames, therefore taking even better the geometry into account. It is trained on ScanNet. We render the 3D semantic instance predictions into the 2D training frames to map them into the same space as all other base models.

The four semantic models produce classifications in four different sets of classes. InternImage predicts 150 ADE20k

[1] Wordnet is a dictionary and synkeys are the names of its entries. I.e., a set of synonymous words has 1 synkey, but a word with different meanings as one synkey per definition.

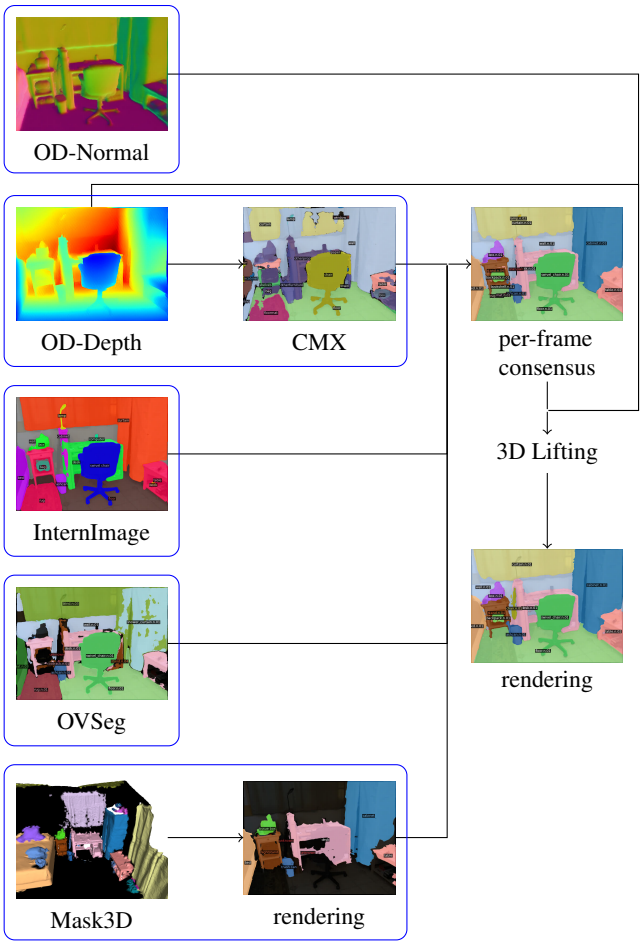

Figure 2. Pipeline Overview. All components in blue boxes run independently and in parallel.

classes, CMX predicts 40 NYU classes, Mask3D predicts 200 ScanNet classes, and our OVSeg prompts cover 186 wordnet classes. In addition to the semantic models, we use OmniData [9] to complement the depth sensor.

3.4. Model Consensus

As illustrated in Fig. 2, we run all models of Sec. 3.3 individually on every frame and then, per frame, merge their predictions together using the translation described in Sec. 3.2. We further use left-right flipping as test time augmentation, which means that each pixel receives votes for possible classes from:

- the standard RGB image and it's flipped version for the 2D segmentation models InternImage, CMX, and OVSeg
- 2 votes (to equalize the test-time augmentation of the RGB frame) from the Mask3D prediction rendered into the current frame
- in the variant where we also use available human annotations, 5 votes from the original ScanNet labels

For every pixel, we choose the class with the maximum number of votes. If no class has sufficient votes (parameterized as a threshold), we set the prediction to "*unknown*" and it will have no loss in the 3D lifting.

3.5. 3D Lifting

By computing a consensus over a diverse set of 2D predictors, we leverage the knowledge and scale of 2D semantic segmentation datasets. However, the per-frame predictions are noisy and often inconsistent, especially around image boundaries. These inconsistencies can be mitigated and the performance can even be improved, as previous work has shown [17, 24], by lifting the 2D predictions into 3D.

Therefore, we leverage the recent progress based on NeRFs to generate multi-view consistent 2D semantic segmentation labels in all frames. Based on the observation in previous works [17, 24] that accurate geometry is important to resolve inconsistencies between predictions of multiple frames instead of hallucinating geometry that would explain semantic predictions, we train an implicit surface model from sdfstudio [32] that has a more explicit surface definition compared to a NeRF yielding improved geometry compared to vanilla NeRF. Thus, we add a semantic head to the Neus-Acc model, train it on all views with losses from RGB reconstruction, sensor depth, monocular normal estimation, and our semantic consensus. Finally, we render the optimized semantics back into all camera frames.

To generate consistent 3D semantic segmentation labels, we follow an established and more direct approach. Given a pointcloud of the scene, we project the pointcloud into each consensus frame to find corresponding pixels and then take a majority vote over all pixels corresponding to a point.

4. Experiments

4.1. Implementation Details

For the 2D models, we use the corresponding available open-source code and adjust it to our pipeline. As described in Sec. 3.2, we generate votes from each 2D model into a common label space. We choose our defined 186 class wordnet label space as output. We choose the label with highest votes, but require a minimum of 3 out of 13 (with ScanNet annotations) resp. 4 out of 8 (automatic pipeline) votes. For 3D optimization, we build on top of SDFStudio [32], specifically the Neus-Acc [29] model, and add a semantic head and semantic rendering similar to [36].

4.2. Datasets

We run our proposed method on three different datasets to show its performance and validate our design choices.

ScanNet [8] We randomly select 5 scenes from the ScanNet that cover all frequent room types. We carefully annotate high-resolution meshes of the scenes using [11] as de-

	2D						3D					
evaluation class set	**NYU** *(40 classes)*			**wordnet** *(186 classes)*			**NYU** *(40 classes)*			**wordnet** *(186 classes)*		
metric	mIoU	mAcc	tAcc	mIoU	mAcc	tAcc	mIoU	mAcc	tAcc	mIoU	mAcc	tAcc
ScanNet labels [8]	47.7	56.2	69.2	38.1	46.3	69.7	40.1	48.2	68.6	17.7	21.3	70.6
SemanticNerf* [36]	45.2	56.6	69.3	32.9	43.7	71.2	36.7	47.1	68.4	14.8	19.3	71.0
LabelMaker3D w/o ScanNet (automatic labels)	50.7	64.0	75.3	33.5	43.5	72.3	41.3	47.3	71.2	15.7	18.1	71.5
LabelMaker3D (Ours)	**53.4**	**65.0**	**77.5**	**39.1**	**49.3**	**77.2**	**44.1**	**53.4**	**76.1**	**18.2**	**22.0**	**76.7**

Table 1. Comparison of the label quality of the ScanNet labels, LabelMaker3D without any human input, and LabelMaker3D taking the ScanNet annotations as additional input. The results are measured over 5 scenes from ScanNet against newly annotated high-quality ground truth. Based on our translation of prediction spaces, we measure metrics over the medium-tail NYU40 set of categories and our full long-tail ground truth categories. For NYU40 classes, LabelMaker3D is capable of producing labels of higher quality than the ScanNet human annotations, without any human input. For more long-tail categories, the automatic mode does not reach the quality of ScanNet, but LabelMaker3D is able to considerably improve human annotations.

scribed in Sec. 3.1 in order to have a complete and accurate groundtruth to evaluate against.

Replica [26] We also evaluate our method on the Replica dataset. This is a semi-synthetic dataset, captured as a high accuracy mesh from real environments and then rendered into trajectories in [36]. We select the 3 'room' scenes and evaluate against the given annotation.

ARKitScenes [1] To showcase the automatic labelling pipeline on an existing dataset, we run it on selected scenes of the ARKitScenes dataset, where only sparse bounding box labels are available up to date. ARKit Scenes consists of trajectories captured with consumer smartphones which are registered to a professional 3D scanner.

4.3. Baselines

We mainly compare LabelMaker3D to the existing manually created annotations in ScanNet [8]. As an additional baseline, we report the result of fitting and rendering the ScanNet annotations with our adapted SemanticNeRF [36].

ScanNet [8]. For this baseline, we measure the quality of the annotations in ScanNet. To this end, we take the raw ScanNet labels and map them into our labelspace defined by wordnet. The mapping from ScanNet IDs to wordnet synkeys is to a large extent already provided in [8].

SemanticNeRF [36]. This baseline is inspired by [36] and adapted to our pipeline by integrating the semantic head into SDFStudio. Then, we run this version of SemanticNeRF on the ScanNet 2D semantic labels. Thus, we can measure the effect of multi-view aggregation and optimization on the groundtruth ScanNet labels. The hypothesised effect is that through the extra RGB and geometry information provided to the NeRF, segmentation boundaries may be smoother than those of the ScanNet 'supervoxels'.

4.4. Comparison to State-of-the-Art

In Tab. 1, we compare LabelMaker3D to the state-of-the-art baselines ScanNet and SemanticNeRF. We report mean intersection-over-union (mIoU), mean accuracy (mAcc), as well as total accuracy (tAcc). We evaluate the methods in 2D by comparing the renderings or labeled frames with renderings from the ground-truth 3D mesh and in 3D by mapping the 2D renderings onto the corresponding vertices in the 3D ground-truth mesh. Further, we measure the metrics over two different label sets. The NYU40 label set [25] consists of 40 semantic classes representing the common indoor classes in the short tail of the label distribution. The wordnet label set consists of 186 classes, therefore measuring performance also over the long tail of the label distribution.

We show that our proposed pipeline generates better labels than human-annotated ScanNet labels and their lifted version through SemanticNeRF [36]. Particularly, on the short tail of the distribution (NYU label set), our pipeline significantly improves over the human annotated labels. This is due to more accurate object boundaries as well as more consistent and complete labels. For the long tail of the label distribution, our method also outperforms all existing baselines indicating that different 2D expert votes and 3D aggregation boosts the quality of the annotated labels. Finally, we show that our fully automatic pipeline outperforms human annotations on NYU40 classes, highlighting the potential of LabelMaker3D to generate labels at scale.

Qualitative comparison with ScanNet [8] In Fig. 4, we compare qualitative results for ScanNet [8] with LabelMaker3D, and our groundtruth. To this end, we mapped the 2D renderings onto the high-resolution ground-truth mesh by projecting the mesh vertices into all labels using a visibility check. One can see that our pipeline produces consistently more complete and correct labels than the human annotations provided by ScanNet [8]. *E.g.*, our method consistently labels the kitchen countertop, the mats in the bathroom, and even the folded chair leaned against the desk.

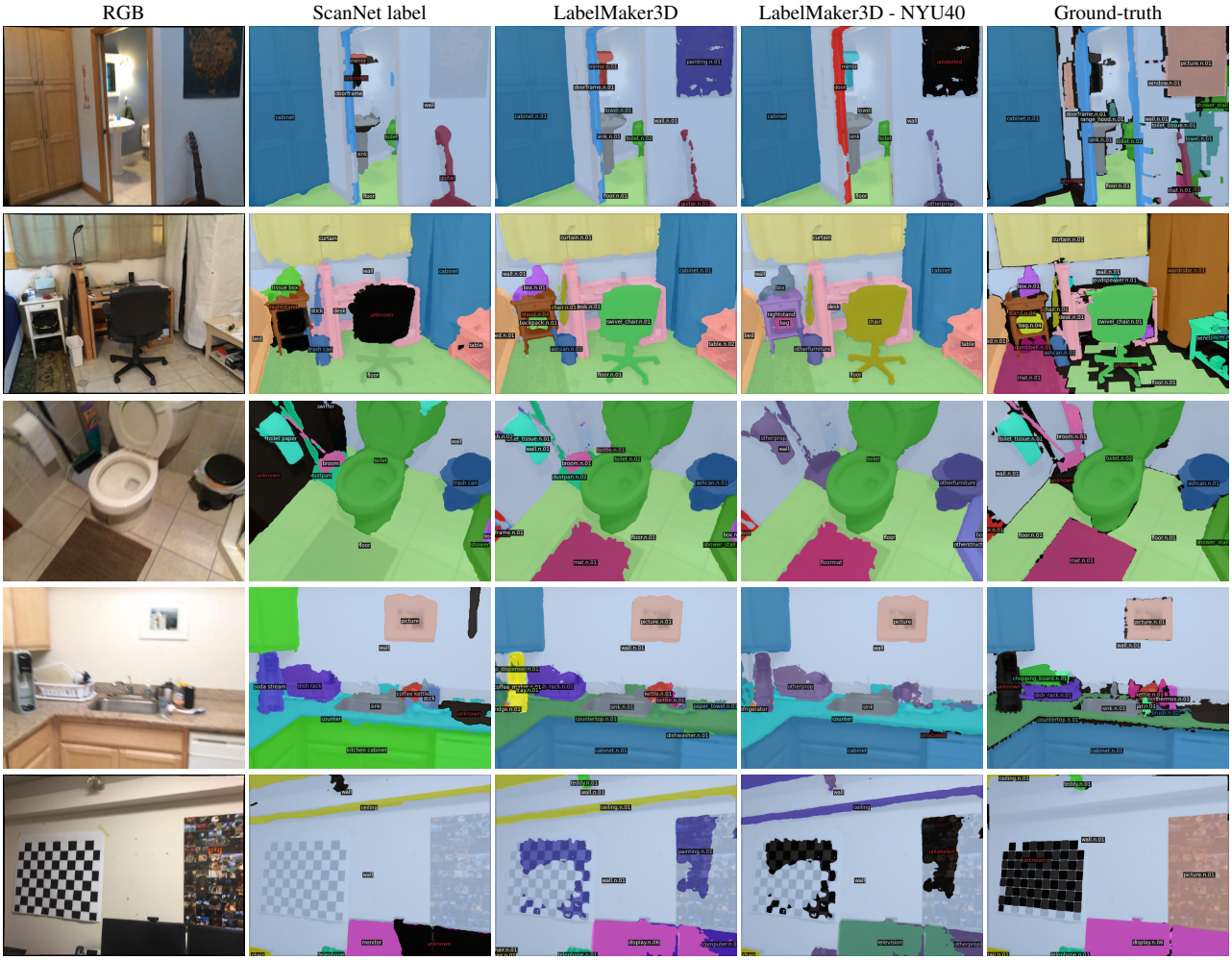

Figure 3. LabelMaker3D generates more accurate and more complete labels compared to the labels annotated by humans and provided by ScanNet. Particularly, unlabeled sections in ScanNet are correctly filled in and many wrong annotations such as missing rogs and pictures are corrected. The output labels can then be projected into differnet label spaces, such as our wordnet space or the NYU40 categories.

	ScanNet *(186 classes)*			**Replica** *(150 classes)*		
	mIoU	mAcc	tAcc	mIoU	mAcc	tAcc
OVSeg	15.3	24.4	43.7	20.7	26.5	69.4
InternImage	30.8	43.5	59.4	38.3	47.7	84.6
CMX	28.2	41.0	54.2	17.0	38.0	84.6
Mask3D	33.7	40.2	38.5	22.6	27.9	30.4
Consensus	38.9	48.3	77.0	39.1	46.2	84.3
LabelMaker3D (ours)	**39.1**	**49.3**	**77.2**	**42.1**	**51.0**	**86.7**

Table 2. Ablation of all base models in LabelMaker3D on our 5 labelled ScanNet [8] scenes and Replica [26]. InternImage is the strongest single base model, but the fusion with other predictions and 3D lifting increases the accuracy considerably beyond any of the state-of-the-art single models.

ScanNet Label Quality Because our experiments require new high-accuracy annotations of ScanNet scenes, we are able to estimate the quality of the default ScanNet labels. As Tab. 1 shows, but also any human who inspects the ScanNet labels knows, these are not perfect. We argue in Sec. 3.1 that this reflects the open-world approach of the dataset and annotation workflow, where – exactly as in any real application – semantics are ambiguous and not always clearly defined. We should also point out that even the detailed annotations we provide are not fully perfect. However, given the background that the ScanNet labels are also used as a benchmark to compare accuracy of semantic classifiers, our results indicate that a perfect prediction would reach accuracy values much lower than 100%. If two methods achieve higher mIoU on ScanNet than the ScanNet labels themselves, it is not possible to draw a clear conclusion about which method is better. This highlights the usefulness of improving the quality of the labels in datasets where some labels already exist.

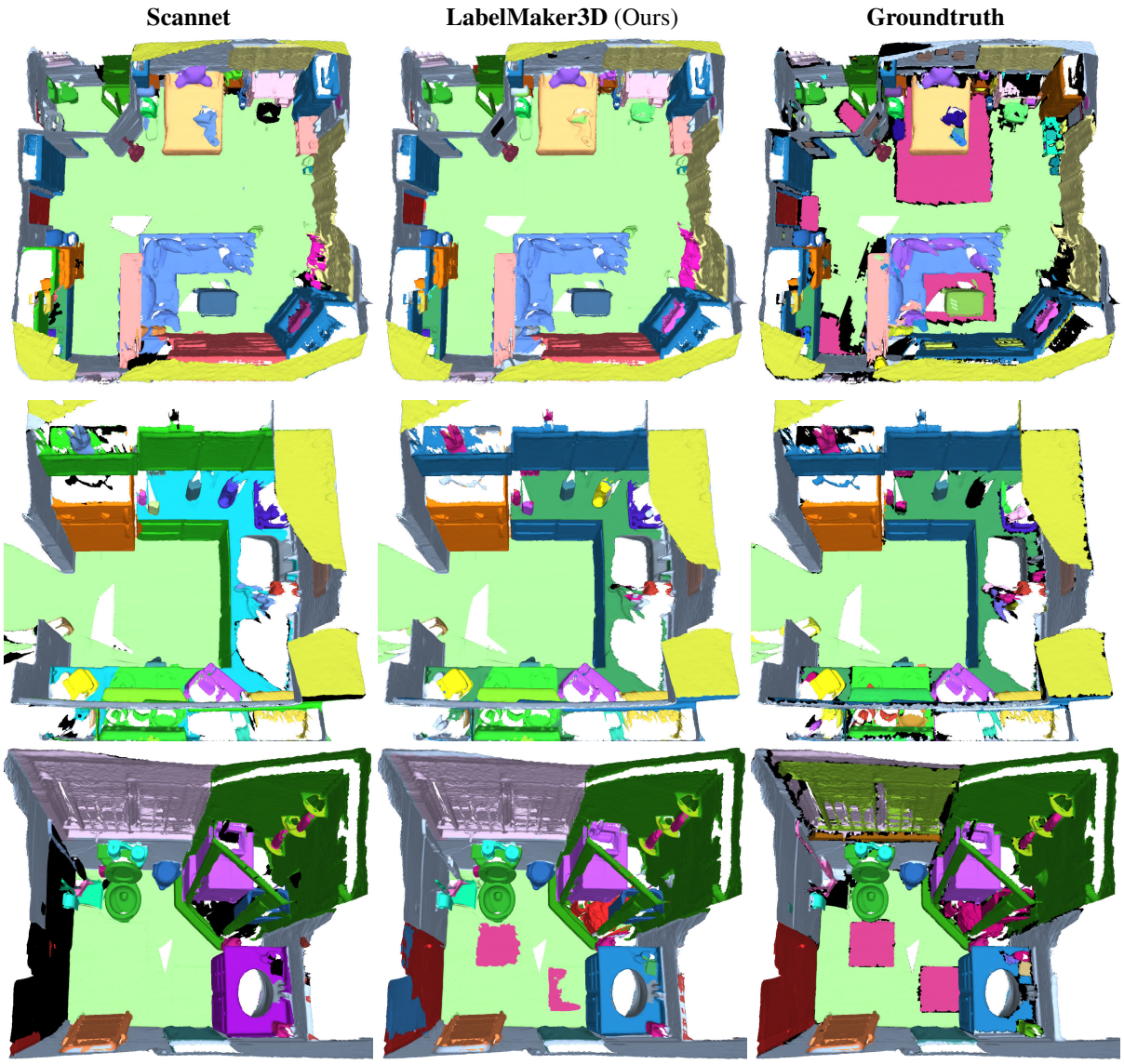

Figure 4. Dense 3D labels for ScanNetv2 [8]. We generate more consistent labels compared to human annotators and preserve rare classes (*e.g.*, swivel chair in front of the desk). Further, the labels are more complete (*e.g.*, wall in bathroom) and we can capture all object in the scene (*e.g.*, dustpan in bathroom).

4.5. Ablation Study

Does consensus voting make the model better? Tab. 2 shows the evaluation on the standard metrics (mIoU, mAcc, tAcc) in 2D for the ScanNet and the Replica datasets. We demonstrate that aggregating individual 2D predictions with our consensus voting mechanism improves upon the individual 2D models. Further, we also show that lifting the 2D consensus into 3D using our optimization pipeline further improves the results compared to the individual 2D models.

Which model is the most important? Tab. 2 shows that the performance of models differs noticeably. Compared to the others, InternImage and Mask3D have the strongest positive impact on the segmentation quality. Additionally and unsurprisingly, Tab. 1 shows that using ScanNet [8] labels as additional votes further improves performance.

Importance of 3D Lifting? We show in Tab. 2 the effect of 3D lifting to aggregate semantic labels and make them multi-view consistent. We compare LabelMaker3D with the aggregated consensus, as well as with individual models,and compute the 2D metrics on ScanNet and Replica. One can see that the 3D lifting significantly improves the performance by at least +1 mIoU.

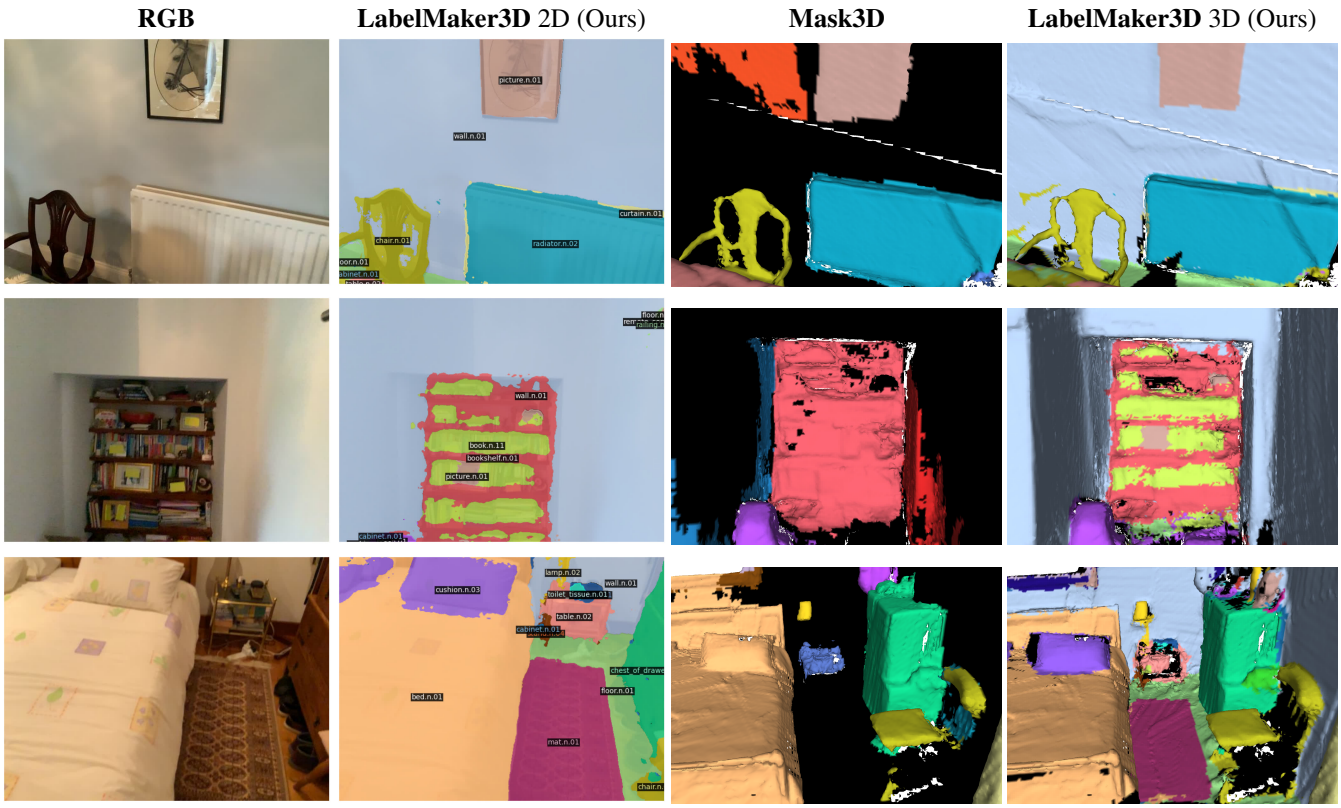

Figure 5. Automatic dense labelling of ARKitScenes. We demonstrate the applicability to label RGB-D datasets that do not have dense labels available. Compared to state-of-the-art Mask3D [23], we generate dense annotations for all classes in the scene. Further, we segment on a higher level of detail (see picture and books in bookshelf, or objects on the cabinet/nightstand). Thus, our labelling pipeline can readily be used on non-label dataset to provide training data for segmentation methods.

4.6. Experiments on ARKitScenes

To demonstrate the applicability of our labelling pipeline to new datasets, for which no dense labels exist, we run our pipeline on a set of scenes from the ARKitScenes [1] dataset. To this end, we process the smartphone trajectories using the low resolution depth maps as sensor depth and the corresponding VGA-resolution images as RGB input. We established these correspondences by synchronizing the depth and RGB timestamps. In Fig. 5, we show qualitative results for 2 scenes of the data set. One can see that the produced labels are more complete and accurate than for Mask3D, a state-of-the-art 3D instance segmentation method. Thus, we demonstrate the feasibility of automatically labeling huge datasets with zero human intervention.

5. Limitations

LabelMaker3D is still limited to a fixed set of classes. Extending it to output language embeddings instead of classes would make it more flexible and potentially help to resolve ambiguities. The 3D lifting with SDFStudio has numerous hyper-parameters, and this work possibly did not yet find the optimal settings. In terms of accuracy, the pipeline can be further profit from newly developed models as research progresses, which will improve the output quality. An interesting next step would be to implement a feedback loop where LabelMaker3D is used to produce a vast amount of automatically labeled training data, on which an additional model is trained as a distillation of the model zoo.

6. Conclusion

We present a fully automatic labeling pipeline that generates semantic annotations of similar quality to human annotations, with zero manual human labeling effort. The method also improves the accuracy and consistency of existing annotations. We quantitatively validate the performance of our pipeline on the ScanNet and Replica datasets. On ScanNet, it outperforms the existing human annotations, and on Replica it improves over all baseline methods. Finally, we showcase the applicability to large-scale 3D datasets and label images and point clouds of ARKitScenes.

Acknowledgments Francis Engelmann is a postdoctoral research fellow at the ETH AI Center. This project is partially funded by the ETH Career Seed Awards “Towards Open-World 3D Scene Understanding” and “ScanNetter”.

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. 2, 5, 8

[2] Amaury Bréhéret. Pixel Annotation Tool. https://github.com/abreheret/PixelAnnotationTool, 2017. 2

[3] Holger Caesar, Jasper Uijlings, and Vittorio Ferrari. Cocostuff: Thing and stuff classes in context. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1209–1218, 2018. 1, 2

[4] Angel Chang, Angela Dai, Thomas Funkhouser, Maciej Halber, Matthias Nießner, Manolis Savva, Shuran Song, Andy Zeng, and Yinda Zhang. Matterport3D: Learning from RGB-D Data in Indoor Environments. In *International Conference on 3D Vision (3DV)*, 2017. 1, 2

[5] Runnan Chen, Youquan Liu, Lingdong Kong, Xinge Zhu, Yuexin Ma, Yikang Li, Yuenan Hou, Yu Qiao, and Wenping Wang. Clip2scene: Towards label-efficient 3d scene understanding by clip. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023. 2

[6] Julian Chibane, Francis Engelmann, Tuan Anh Tran, and Gerard Pons-Moll. Box2Mask: Weakly Supervised 3D Semantic Instance Segmentation using Bounding Boxes. In *European Conference on Computer Vision (ECCV)*, 2022. 1

[7] M Cordts, M Omran, S Ramos, T Rehfeld, M Enzweiler, R Benenson, U Franke, S Roth, and B Schiele. The Cityscapes Dataset for Semantic Urban Scene Understanding. pages 3213–3223, 2016. 1, 2

[8] Angela Dai, Angel X Chang, Manolis Savva, Maciej Halber, Thomas Funkhouser, and Matthias Nießner. ScanNet: Richly-Annotated 3D Reconstructions of Indoor Scenes. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017. 1, 2, 3, 4, 5, 6, 7

[9] Ainaz Eftekhar, Alexander Sax, Jitendra Malik, and Amir Zamir. Omnidata: A scalable pipeline for making multi-task mid-level vision datasets from 3d scans. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 10786–10796, 2021. 4

[10] 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. 1, 2

[11] Theodora Kontogianni, Ekin Celikkan, Siyu Tang, and Konrad Schindler. Interactive object segmentation in 3d point clouds. In *International Conference on Robotics and Automation (ICRA)*, 2023. 2, 3, 4

[12] Lars Kreuzberg, Idil Esen Zulfikar, Sabarinath Mahadevan, Francis Engelmann, and Bastian Leibe. 4D-STOP: Panoptic Segmentation of 4D LiDAR using Spatio-temporal Object Proposal Generation and Aggregation. In *European Conference on Computer Vision (ECCV) Workshops*, 2022. 1

[13] labelme github contributors. labelme: Image polygonal annotation with python. 2

[14] John Lambert, Zhuang Liu, Ozan Sener, James Hays, and Vladlen Koltun. MSeg: A Composite Dataset for Multi-Domain Semantic Segmentation. pages 2879–2888, 2020. 3

[15] Feng Liang, Bichen Wu, Xiaoliang Dai, Kunpeng Li, Yinan Zhao, Hang Zhang, Peizhao Zhang, Peter Vajda, and Diana Marculescu. Open-vocabulary semantic segmentation with mask-adapted clip. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7061–7070, 2023. 3

[16] Feng Liang, Bichen Wu, Xiaoliang Dai, Kunpeng Li, Yinan Zhao, Hang Zhang, Peizhao Zhang, Peter Vajda, and Diana Marculescu. Open-Vocabulary Semantic Segmentation with Mask-adapted CLIP. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023. 1, 2

[17] Zhizheng Liu, Francesco Milano, Jonas Frey, Roland Siegwart, Hermann Blum, and Cesar Cadena. Unsupervised continual semantic adaptation through neural rendering. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023. 2, 4

[18] George A Miller. Wordnet: a lexical database for english. *Communications of the ACM*, 1995. 3

[19] Alexey Nekrasov, Jonas Schult, Or Litany, Bastian Leibe, and Francis Engelmann. Mix3d: Out-of-context Data Augmentation for 3D Scenes. In *International Conference on 3D Vision (3DV)*, 2021. 1

[20] Songyou Peng, Kyle Genova, Chiyu "Max" Jiang, Andrea Tagliasacchi, Marc Pollefeys, and Thomas Funkhouser. OpenScene: 3D Scene Understanding with Open Vocabularies. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023. 1

[21] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning Transferable Visual Models From Natural Language Supervision. In *International Conference on Machine Learning (ICML)*, 2021. 2, 3

[22] David Rozenberszki, Or Litany, and Angela Dai. Language-Grounded Indoor 3D Semantic Segmentation in the Wild. In *European Conference on Computer Vision (ECCV)*, 2022. 3

[23] 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. 1, 3, 8

[24] Yawar Siddiqui, Lorenzo Porzi, Samuel Rota Bulò, Norman Müller, Matthias Nießner, Angela Dai, and Peter Kontschieder. Panoptic lifting for 3d scene understanding with neural fields. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023. 2, 4

[25] Nathan Silberman, Derek Hoiem, Pushmeet Kohli, and Rob Fergus. Indoor Segmentation and Support Inference from RGBD Images. pages 746–760. Springer Berlin Heidelberg, 2012. 1, 2, 5

[26] Julian Straub, Thomas Whelan, Lingni Ma, Yufan Chen, Erik Wijmans, Simon Green, Jakob J. Engel, Raul Mur-Artal,

Carl Ren, Shobhit Verma, Anton Clarkson, Mingfei Yan, Brian Budge, Yajie Yan, Xiaqing Pan, June Yon, Yuyang Zou, Kimberly Leon, Nigel Carter, Jesus Briales, Tyler Gillingham, Elias Mueggler, Luis Pesqueira, Manolis Savva, Dhruv Batra, Hauke M. Strasdat, Renzo De Nardi, Michael Goesele, Steven Lovegrove, and Richard Newcombe. The Replica dataset: A digital replica of indoor spaces. *arXiv*, 2019. 2, 5, 6

[27] Ayça Takmaz, Elisabetta Fedele, Robert W Sumner, Marc Pollefeys, Federico Tombari, and Francis Engelmann. OpenMask3D: Open-Vocabulary 3D Instance Segmentation. *Neural Information Processing Systems (NeurIPS)*, 2023. 1

[28] Ayça Takmaz, Jonas Schult, Irem Kaftan, Mertcan Akçay, Bastian Leibe, Robert Sumner, Francis Engelmann, and Siyu Tang. 3D Segmentation of Humans in Point Clouds with Synthetic Data. In *International Conference on Computer Vision (ICCV)*, 2023. 1

[29] Peng Wang, Lingjie Liu, Yuan Liu, Christian Theobalt, Taku Komura, and Wenping Wang. Neus: Learning neural implicit surfaces by volume rendering for multi-view reconstruction. *arXiv preprint arXiv:2106.10689*, 2021. 4

[30] Wenhai Wang, Jifeng Dai, Zhe Chen, Zhenhang Huang, Zhiqi Li, Xizhou Zhu, Xiaowei Hu, Tong Lu, Lewei Lu, Hongsheng Li, et al. Internimage: Exploring large-scale vision foundation models with deformable convolutions. *arXiv preprint arXiv:2211.05778*, 2022. 1, 2, 3

[31] Fisher Yu, Haofeng Chen, Xin Wang, Wenqi Xian, Yingying Chen, Fangchen Liu, Vashisht Madhavan, and Trevor Darrell. Bdd100k: A diverse driving dataset for heterogeneous multitask learning. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020. 1

[32] Zehao Yu, Anpei Chen, Bozidar Antic, Songyou Peng Peng, Apratim Bhattacharyya, Michael Niemeyer, Siyu Tang, Torsten Sattler, and Andreas Geiger. Sdfstudio: A unified framework for surface reconstruction, 2022. 4

[33] Yuanwen Yue, Theodora Kontogianni, Konrad Schindler, and Francis Engelmann. Connecting the Dots: Floorplan Reconstruction Using Two-Level Queries. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023. 1

[34] Yuanwen Yue, Sabarinath Mahadevan, Jonas Schult, Francis Engelmann, Bastian Leibe, Konrad Schindler, and Theodora Kontogianni. AGILE3D: Attention Guided Interactive Multi-object 3D Segmentation. *arXiv preprint arXiv:2306.00977*, 2023. 2

[35] Jiaming Zhang, Huayao Liu, Kailun Yang, Xinxin Hu, Ruiping Liu, and Rainer Stiefelhagen. Cmx: Cross-modal fusion for rgb-x semantic segmentation with transformers. *IEEE Transactions on Intelligent Transportation Systems*, 2023. 3

[36] Shuaifeng Zhi, Tristan Laidlow, Stefan Leutenegger, and Andrew J Davison. In-place scene labelling and understanding with implicit scene representation. In *International Conference on Computer Vision (ICCV)*, 2021. 2, 4, 5

[37] Shuaifeng Zhi, Edgar Sucar, Andre Mouton, Iain Haughton, Tristan Laidlow, and Andrew J Davison. ilabel: Interactive neural scene labelling. *arXiv preprint arXiv:2111.14637*, 2021. 2

[38] Bolei Zhou, Hang Zhao, Xavier Puig, Sanja Fidler, Adela Barriuso, and Antonio Torralba. Scene parsing through ade20k dataset. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 633–641, 2017. 1, 2