Virtual Multi-view Fusion for 3D Semantic Segmentation

Abhijit Kundu(✉), Xiaoqi Yin, Alireza Fathi, David Ross, Brian Brewington, Thomas Funkhouser, and Caroline Pantofaru

Google Research, San Francisco, USA
abhijitkundu@google.com

Abstract. Semantic segmentation of 3D meshes is an important problem for 3D scene understanding. In this paper we revisit the classic multiview representation of 3D meshes and study several techniques that make them effective for 3D semantic segmentation of meshes. Given a 3D mesh reconstructed from RGBD sensors, our method effectively chooses different virtual views of the 3D mesh and renders multiple 2D channels for training an effective 2D semantic segmentation model. Features from multiple per view predictions are finally fused on 3D mesh vertices to predict mesh semantic segmentation labels. Using the large scale indoor 3D semantic segmentation benchmark of ScanNet, we show that our virtual views enable more effective training of 2D semantic segmentation networks than previous multiview approaches. When the 2D per pixel predictions are aggregated on 3D surfaces, our virtual multiview fusion method is able to achieve significantly better 3D semantic segmentation results compared to all prior multiview approaches and recent 3D convolution approaches.

Keywords: 3D semantic segmentation · Scene understanding

1 Introduction

Semantic segmentation of 3D scenes is a fundamental problem in computer vision. Given a 3D representation of a scene (e.g., a textured mesh of an indoor environment), the goal is to output a semantic label for every surface point. The output could be used for semantic mapping, site monitoring, training autonomous navigation, and several other applications.

State-of-the-art (SOTA) methods for 3D semantic segmentation currently use 3D sparse voxel convolution operators for processing input data. For example, MinkowskiNet [7] and SparseConvNet [11] each load the input data into a sparse 3D voxel grid and extract features with sparse 3D convolutions. These "place-centric" methods are designed to recognize 3D patterns and thus work well for

Electronic supplementary material The online version of this chapter (https://doi.org/10.1007/978-3-030-58586-0_31) contains supplementary material, which is available to authorized users.

A. Vedaldi et al. (Eds.): ECCV 2020, LNCS 12369, pp. 518–535, 2020.
https://doi.org/10.1007/978-3-030-58586-0_31

types of objects with distinctive 3D shapes (e.g., chairs), and not so well for others (e.g., wall pictures). They also take a considerable amount of memory, which limits spatial resolutions and/or batch sizes.

Alternatively, when posed RGB-D images are available, several researchers have tried using 2D networks designed for processing photographic RGB images to predict dense features and/or semantic labels and then aggregate them on visible 3D surfaces [15,41], and others project features onto visible surfaces and convolve them further in 3D [10,18,19,40]. Although these "view-centric" methods utilize massive image processing networks pretrained on large RGB image datasets, they do not achieve SOTA performance on standard 3D segmentation benchmarks due to the difficulties of occlusion, lighting variation, and camera pose misalignment in RGB-D scanning datasets. None of the view-based methods is currently in the top half of the current leaderboard for the 3D Semantic Label Challenge of the ScanNet benchmark.

In this paper, we propose a new view-based approach to 3D semantic segmentation that overcomes the problems with previous methods. The key idea is to use synthetic images rendered from "virtual views" of the 3D scene rather than restricting processing to the original photographic images acquired by a physical camera. This approach has several advantages that address the key problems encountered by previous view-centric method [3,21]. First, we select camera intrinsics for virtual views with unnaturally wide field-of-view to increase the context observed in each rendered image. Second, we select virtual viewpoints at locations with small variation in distances/angles to scene surfaces, relatively few occlusions between objects, and large surface coverage redundancy. Third, we render non-photorealistic images without view-dependent lighting efffects and occlusions by backfacing surfaces – i.e., virtual views can look into a scene from behind the walls, floors, and ceilings to provide views with relatively large context and little occlusion. Fourth, we aggregate pixel-wise predictions onto 3D surfaces according to exactly known camera parameters of virtual views, and thus do not encounter "bleeding" of semantic labels across occluding contours. Fifth, virtual views during training and inference can mimic multi-scale training and testing and avoid scale in-variance issues of 2D CNNs. We can generate as many virtual views as we want during both training and testing. During training, more virtual views provides robustness due to data augmentation. During testing, more views provides robustness due to vote redundancy. Finally, the 2D segmentation model in our multiview fusion approach can benefit from large image pre-training data like ImageNet and COCO, which are unavailable for pure 3D convolution approaches.

We have investigated the idea of using virtual views for semantic segmentation of 3D surfaces using a variety of ablation studies. We find that the broader design space of view selection enabled by virtual cameras can significantly boost the performance of multiview fusion as it allows us to include physically impossible but useful views (e.g., behind walls). For example, using virtual views with original camera parameters improves 3D mIoU by 3.1% compared with using original photographic images, using additional normal and coordinates channels

and higher field of view can further boost mIoU by 5.7%, and an additional gain of 2.1% can be achieved by carefully selecting virtual camera poses to best capture the 3D information in the scenes and optimize for training 2D CNNs.

Overall, our simple system is able to achieve state-of-the-art results on both 2D and 3D semantic labeling tasks in ScanNet Benchmark [9], and is significantly better than the best performing previous multi-view methods and very competitive with recent 3D methods based on convolutions of 3D point sets and meshes. In addition, we show that our proposed approach consistently outperforms 3D convolution and real multi-view fusion approaches when there are fewer scenes for training. Finally, we show that similar performance can be obtained with significantly fewer views in the inference stage. For example, multi-view fusion with ∼12 virtual views per scene will outperform that with all ∼1700 original views per scene.

The rest of the paper is organized as follows. We introduce the research landscape and related work in Sect. 2. We describe the proposed virtual multi-view fusion approach in detail in Sect. 3–Sect. 5. Experiment results and ablation studies of our proposed approach are presented in Sect. 6. Finally we conclude the paper with discussions of future directions in Sect. 7.

2 Related Work

There has been a large amount of previous work on semantic segmentation of 3D scenes. The following reviews only the most related work.

Multi-view Labeling. Motivated by the success of view-based methods for object classification [35], early work on semantic segmentation of RGB-D surface reconstructions relied on 2D networks trained to predict dense semantic labels for RGB images. Pixel-wise semantic labels were backprojected and aggregated onto 3D reconstructed surfaces via weighted averaging [15,41], CRFs [25], Bayesian fusion [24,41,46], or 3D convolutions [10,18,19]. These methods performed multiview aggregation only for the originally captured RGB-D photographic images, which suffer from limited fields-of-view, restricted viewpoint ranges, view-dependent lighting effects, and misalignments with reconstructed surface geometry, all of which reduce semantic segmentation performance. To overcome these problems, some recent work has proposed using synthetic images of real data in a multiview labeling pipeline [3,12,21], but they still use camera parameters typical of real images (e.g., small field of view), propose methods suitable only for outdoor environments (lidar point clouds of cities), and do not currently achieve state-of-the-art results.

3D Convolution. Recent work on 3D semantic segmentation has focused on methods that extract and classify features directly with 3D convolutions. Network architectures have been proposed to extract features from 3D point clouds [16,29–31,33,38], surface meshes [14,17], voxel grids [34], and octrees [32]. Current state-of-the-art methods are based on sparse 3D voxel convolutions [7,8,11], where submanifold sparse convolution operations are used to compute features

on sparse voxel grids. These methods utilize memory more efficiently than dense voxel grids, but are still limited in spatial resolution in comparison to 2D images and can train with supervision only on 3D datasets, which generally are very small in comparison to 2D image datasets.

Synthetic Data. Other work has investigated training 2D semantic segmentation networks using computer graphics renderings of 3D synthetic data [47]. The main advantage of this approach is that image datasets can be created with unlimited size by rendering novel views of a 3D scene [22,26]. However, the challenge is generally domain adaptation – networks trained on synthetic data and tested on real data usually do not perform well. Our method avoids this problem by training and testing on synthetic images rendered with the same process.

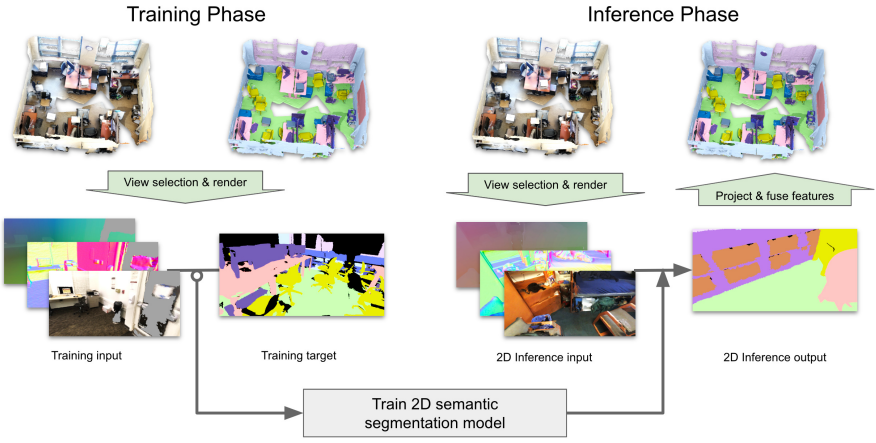


Fig. 1. Virtual multi-view fusion system overview.

3 Method Overview

The proposed multiview fusion approach is illustrated in Fig. 1. At a high level, it consists of the following steps.

Training Stage. During the training stage, we first select virtual views for each 3D scene, where for each virtual view we select camera intrinsics, camera extrinsics, which channels to render, and rendering parameters (e.g., depth range, backface culling). We then generate training data by rendering the selected virtual views for the selected channels and ground truth semantic labels. We train 2D semantic segmentation models using the rendered training data and use the model in the inference stage.

Inference Stage. At inference stage, we select and render virtual views using a similar approach as in the training stage, but without the ground truth semantic labels. We conduct 2D semantic segmentation on the rendered virtual views

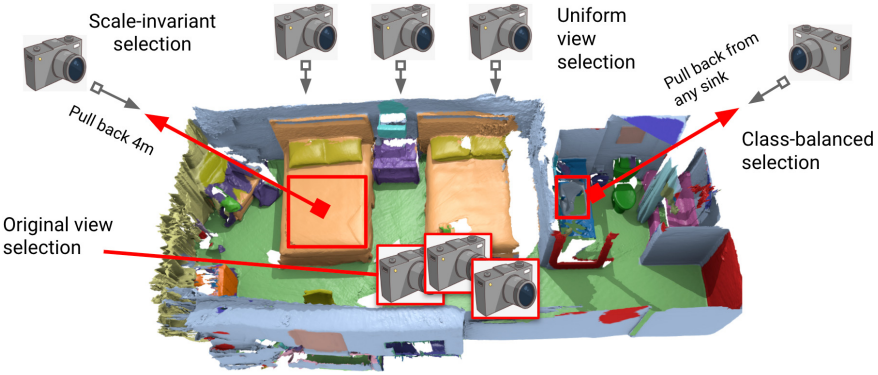


Fig. 2. Proposed virtual view selection approaches.

using the trained model, project the 2D semantic features to 3D, then derive the semantic category in 3D by fusing multiple projected 2D semantic features.

4 Virtual View Selection

Virtual view selection is central to the proposed multiview fusion approach as it brings key advantages over multiview fusion with original image views. First, it allows us to freely select camera parameters that work best for 2D semantic segmentation tasks, and with any set of 2D data augmentation approaches. Second, it significantly broadens the set of views to choose from by relaxing the physical constraints of real cameras and allowing views from unrealistic but useful camera positions that significantly boost model performance, e.g. behind a wall. Third, it allows 2D views to capture additional channels that are difficult to capture with real cameras, e.g., normals and coordinates. Finally, by selecting and rendering virtual views, we have essentially eliminated any errors in the camera calibration and pose estimation, which are common in the 3D reconstruction process. Lastly, sampling views consistently at different scales resolves scale in-variance issues of traditional 2D CNNs.

Camera Intrinsics. A significant constraint of original image views is the FOV - images may have been taken very close to objects or walls, say, and lack the object features and context necessary for accurate classification. Instead, we use a pinhole camera model with significantly higher field of view (FOV) than the original cameras, providing larger context that leads to more accurate 2D semantic segmentation [27]. Figure 3 shows an example of original views compared with virtual views with high FOV.

Camera Extrinsics. We use a mixture of the following sampling strategies to select camera extrinsics as shown in Fig. 2 and Fig. 4.

Fig. 3. Original views vs. virtual views. High FOV provides larger context of the scene which helps 2D perception, e.g., the chair in the bottom right corner is partially represented in the original view but can easily segmented in the high FOV virtual view.

Fig. 4. Example virtual view selection on two ScanNet scenes. Green curve is the trajectory of the original camera poses; Blue cameras are the selected views with the proposed approach. Note that we only show a random subset of all selected views for illustration purposes. (Color figure online)

- Uniform sampling. We want to uniformly sample camera extrinsics to generate many novel views, independently from the specific structure of the 3D scene. Specifically, we use top-down views from uniformly sampled positions at the top of the 3D scene, as well as views that look through the center of the scene but with uniformly sampled positions in the 3D scene.
- Scale-invariant sampling. As 2D convolutional neural networks are generally not scale invariant, the model performance may suffer if the scales of views do not match the 3D scene. To overcome this limitation, we propose sampling views at a range of scales with respect to segments in the 3D scene. Specifically, we do an over-segmentation of the 3D scene, and for each segment, we position the cameras to look at the segment by pulling back to a certain range of distances along the normal direction. We do a depth check to avoid occlusions by foreground objects. If backface culling is disabled in the rendering stage (discussed in more detail below), we do a ray tracing and drop

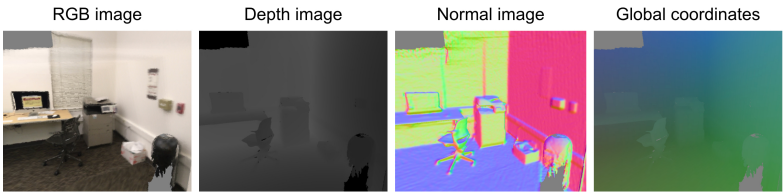


Fig. 5. Example virtual rendering of selected channels.

any views blocked by the backfaces. Note the over-segmentation of the 3D scene is unsupervised and does not use the ground truth semantic labels, so the scale-invariant sampling can be applied both in the training and inference stages.

- Class-balanced sampling. Class balancing has been extensively used as data augmentation approaches for 2D semantic segmentation. We conduct class balancing by selecting views that look at mesh segments of under-represented semantic categories, similar to the scale-invariant sampling approach. Note this sampling approach only applies to the training stage when the ground truth semantic labels are available.
- Original views sampling. We also sample from the original camera views as they represent how a human would choose camera views in the real 3D scene with real physical constraints. Also, the 3D scene is reconstructed from the original views, so including them can make sure we cover corner cases that would otherwise be difficult as random virtual views.

Channels for Rendering. To exploit all the 3D information available in the scene, we render the following channels: RGB color, normal, normalized global XYZ coordinates. The additional channels allow us to go beyond the limitations of the existing RGB-D sensors. While depth image also contains the same information, we think normalized global coordinate image makes the learning problem simpler as now just like the normal and color channel, coordinate values of the same 3D point is view invariant. Figure 5 shows example rendered views of the selected channels.

Rendering Parameters. We turn on backface culling in the rendering so that the backfaces do not block the camera views, further relaxing the physical constraints of the 3D scene and expanding the design space of the view selection. For example, as shown in Fig. 6, in an indoor scenario, we can select views from outside a room which typically include more context of the room and can potentially improve model performance; On the other hand, with backface culling turned off, we either are constrained ourselves to views inside the room therefore limited context, or suffer from high occlusion by the backfaces of the walls.

Training vs. Inference Stage. We want to use similar view selection approaches for the training and inference stages to avoid creating a domain gap, e.g., if we sampled many top-down views in the training stage but used

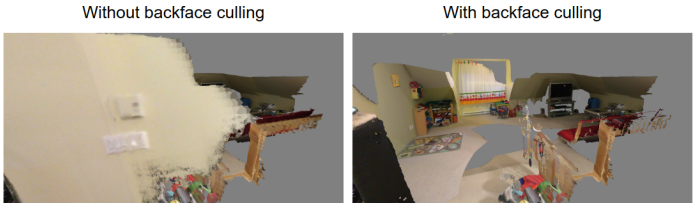


Fig. 6. Effect of backface culling. Backface culling allows the virtual camera to see more context from views that are not physically possible with real cameras.

lots of horizontal views in the inference stage. The main difference between the view selection strategies between the two stages is the class-balancing which can only be done in the training stage. Also, while the inference cost may matter in real-world applications, in this paper we consider offline 3D segmentation tasks and do not optimize the computation cost in either stage, so we can use as many virtual views as needed in either stage.

5 Multiview Fusion

5.1 2D Semantic Segmentation Model

With rendered virtual views as training data, we are now ready to train a 2D semantic segmentation models. We use a xcpetion65 [6] feature extractor and DeeplabV3+ [4] decoder. We initialize our model from pre-trained classification model checkpoints trained on ImageNet. When training a model with additional input channels like normal image and co-ordinate image we modify the first layer of the pre-training checkpoints by tiling the weights across the additional channels and normalize them across each spatial position such that the sum of weights along the channel dimension remains the same.

5.2 3D Fusion of 2D Semantic Features

During inference, we run the 2D semantic segmentation model on virtual views and obtain image features (e.g., unary probabilities for each pixel). To project the 2D image features to 3D, we use the following approach: We render a depth channel on the virtual views; For each 3D point, we project it back to each of the virtual views, and accumulate the image feature of the projected pixel only if the depth of the pixel matches the point-to-camera distance. This approach achieves better computational efficiency than the alternative approach of casting rays from each pixel to find the 3D point to aggregate. First, the number of 3D points in a scene are much less than the total number of pixels in all rendered images of the scene. Secondly, projecting a 3D point with a depth check is faster than operations involving ray casting.

Formally, let $\mathbf{X}_k \in \mathbb{R}^3$ be the 3D position of the kth point, $\mathbf{x}_{k,i} \in \mathbb{R}^2$ be the pixel coordinates by projecting the kth 3D point to virtual view $i \in \mathcal{I}$, $\mathbf{K}_i$ be its instrinsics matrix while $\mathbf{R}_i$ be the rotation, $\mathbf{t}_i$ the translation in the extrinsics, $\mathcal{A}_i$ be the set of valid pixel coordinates. Let $c_{k,i}$ be the distance between the position of camera i and kth 3D point. We have:

$$\mathbf{x}_{k,i} = \mathbf{K}_i(\mathbf{R}_i\mathbf{X}_k + \mathbf{t}_i) \tag{1}$$

$$c_{k,i} = \left\|\mathbf{X}_k - \mathbf{R}_i^{-1}\mathbf{t}_i\right\|_2 \tag{2}$$

Let $\mathcal{F}_k$ be the set of image features projected to the kth 3D point, $\mathbf{f}_i(\cdot)$ be the mapping from pixel coordinates in virtual image i to the image feature vector, $d_i(\cdot)$ be the mapping from pixel coordinates to the depth since we render depth channel. Then:

$$\mathcal{F}_k = \{\mathbf{f}_i(\mathbf{x}_{k,i}) \mid \mathbf{x}_{k,i} \in \mathcal{A}_i, |d_i(\mathbf{x}_{k,i}) - c_{k,i}| < \delta, \forall i \in \mathcal{I}\} \tag{3}$$

where $\delta > 0$ is the threshold for depth matching.

To fuse projected features $\mathcal{F}_k$ for 3D point k, we simply take the average of all features in $\mathcal{F}_k$ and obtain the fused feature. There simple fusion function was better than other alternatives like picking the category with maximum probability across all projected features.

6 Experiments

We ran a series of experiments to evaluate how well our proposed method for 3D semantic segmentation of RGB-D scans works compared to alternative approaches and to study how each component of our algorithm affects the results.

6.1 Evaluation on ScanNet Dataset

We evaluate our approach on ScanNet dataset [9], on the hidden test set for the task of both 3D mesh semantic segmentation and 2D image semantic segmentation. We also perform a detailed ablation study on the validation set of ScanNet in Sect. 6.3. Unlike our ablation studies, we use xception101 [6] as the 2D backbone and we additionally use ADE20K [48] for pre-training the 2D segmentation model. We compare our virtual multiview-fusion approach against state-of-the-art methods for 3D semantic segmentation, most of which utilize 3D convolutions of sparse voxels or point clouds. We also compare our 2D image segmentation results obtained by projecting back 3D labels obtained by our multiview fusion approach. Results are available in Table 1.

From these results, we see that our approach outperforms previous approaches based on convolutions of 3D point sets [16,30,38,43,44], and it achieves results comparable to the SOTA methods based on sparse voxel convolutions [7,11,13]. Our method achieves the best 2D segmentation results (74.5%). In Sect. 6.3, we also demonstrate improvement in single frame 2D semantic segmentation.

Table 1. Semantic segmentation results on ScanNet validation and test splits.

Method	3D mIoU (val split)	3D mIoU (test split)	2D mIoU (test split)
PointNet [30]	53.5	55.7	-
3DMV [10]	-	48.4	49.8
SparseConvNet [11]	69.3	72.5	-
PanopticFusion [28]	-	52.9	-
PointConv [43]	61.0	66.6	-
JointPointBased [5]	69.2	63.4	-
SSMA [39]	-	-	57.7
KPConv [38]	69.2	68.4	-
MinkowskiNet [7]	72.2	73.6	-
PointASNL [44]	63.5	66.6	-
OccuSeg [13]	-	**76.4**	-
JSENet [16]	-	69.9	-
Ours	**76.4**	74.6	**74.5**

Our approach performs significantly better than any previous multiview fusion methods [10,28] on ScanNet semantic labeling benchmark. The mean IoU of the previously best performing multiview method on the ScanNet test set is 52.9% [28], which is significantly less than our results of 74.6%. By using our virtual views, we are able to learn 2D semantic segmentation networks that provide more accurate and more consistent semantic labels when aggregated on 3D surfaces. The result is semantic segmentations of high accuracy and sharp boundaries, as shown in Fig. 7 (Table 2).

Table 2. Results on the Stanford 3D Indoor Spaces (S3DIS) dataset [1]. Following previous works we use Fold-1 split with Area5 as the test set.

Method	mIOU	ceiling	floor	wall	beam	column	window	door	chair	table	bookcase	sofa	board	clutter
PointNet [30]	41.09	88.8	97.3	69.8	0.1	3.9	46.3	10.8	52.6	58.9	40.3	5.9	26.4	33.2
SegCloud [37]	48.92	90.1	96.1	69.9	0.0	18.4	38.4	23.1	75.9	70.4	58.4	40.9	13.0	41.6
TangentConv [36]	52.80	90.5	97.7	74.0	0.0	20.7	39.0	31.3	77.5	69.4	57.3	38.5	48.8	39.8
3D RNN [45]	53.40	95.2	98.6	77.4	0.8	9.8	52.7	27.9	76.8	78.3	58.6	27.4	39.1	51.0
PointCNN [23]	57.26	92.3	98.2	79.4	0.0	17.6	22.7	62.1	80.6	74.4	66.7	31.7	62.1	56.7
SuperpointGraph [20]	58.04	89.4	96.9	78.1	0.0	42.8	48.9	61.6	84.7	75.4	69.8	52.6	2.1	52.2
PCCN [42]	58.27	90.3	96.2	75.9	0.3	6.0	69.5	63.5	66.9	65.6	47.3	68.9	59.1	46.2
PointASNL [44]	62.60	94.3	98.4	79.1	0.0	26.7	55.2	66.2	83.3	86.8	47.6	68.3	56.4	52.1
MinkowskiNet [7]	65.35	91.8	98.7	86.2	0.0	34.1	48.9	62.4	89.8	81.6	74.9	47.2	74.4	58.6
Ours	**65.38**	92.9	96.9	85.5	0.8	23.3	65.1	45.7	85.8	76.9	74.6	63.1	82.1	57.0

6.2 Evaluation on Stanford 3D Indoor Spaces (S3DIS)

We also evaluated our method on the Stanford Large-Scale 3D Indoor Spaces Dataset (S3DIS) [1,2] for the task of semantic 3D segmentation. The proposed

Fig. 7. Qualitiative 3D semantic segmentation results on ScanNet test set.

virtual multi-view fusion approach achieves 65.4% 3D mIoU, outperforming recent SOTA methods MinkowskiNet [7] (65.35%) and PointASNL [44] (62.60%). See Table 1 for quantitative evaluation. Figure 8 shows the output of our approach on Area5 scene from S3DIS dataset.

6.3 Ablation Studies

We investigate which aspects of our proposed method make the most difference we performed ablation study on the ScanNet [9]. To perform this experiment, we started with a baseline method that trains a model to compute 2D semantic segmentation for the original photographic images, uses it to predict semantics for all the original views in the validation set, and then aggregates the class probabilities on backprojected 3D surfaces using the simple averaging method

Fig. 8. Qualitiative 3D semantic segmentation results on Area5 of Stanford 3D Indoor Spaces (S3DIS) Dataset. Semantic label colors are overlayed on the textured mesh. *Ceiling* not shown for clarity.

described in Sect. 3. This mean class IoU of this baseline result is shown in the top row of Table 3. We then performed a series of tests where we included features of our virtual view algorithm one-by-one and measured the impact on performance. The second row shows the impact of using rendered images rather than photographic ones; the third shows the impact of adding additional image normal and coordinate channels captured during rendering; the fourth row shows the impact of rendering images with two times larger field-of-view; the fifth row shows the impact of our virtual viewpoints selection algorithm. We find that each of these ideas improves the 3D segmentation IoU performance significantly.

Specifically, with fixed camera extrinsics matching the original views, we compare the effect of virtual view renderings versus the original photographic images: using virtual views leads to 3.1% increase of 3D mIoU as it removes any potential errors in the 3D reconstruction and pose estimation process. Using additional channels of normal and global coordinates achieves another 2.9% performance boost in 3D mIoU as it allows the 2D semantic segmentation model to exploit the 3D information in the scene other than RGB. Increasing the FOV further improves the 3D mIoU by 1.8% since it allows the 2D model to use more context. Lastly, view sampling with backface culling achieves the best performance and an 2.2% improvement compared to the original views, showing that the camera poses can significantly affect the perception of 3D scenes.

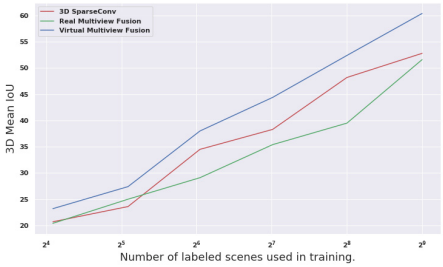
In addition, we compute and compare a) the *single-view* 2D image mIoU, which compares 2D ground truth with the prediction of a 2D semantic segmentation model from single image, and b) *multi-view* 2D image mIoU, which compares ground truth with the reprojected semantic labels from the 3D semantic segmentation after multiview fusion. In all cases, we observed consistent improvements of 2D image mIoU after multiview fusion by a margin of 5.3% to 8.4%. This shows the multiview fusion effectively aggregates the observations and resolves the inconsistency between different views. Note that the largest single-view to multi-view improvement (8.4%) is observed in the first row, i.e., on the original views, which confirms our hypothesis of potential errors and inconsistency in the 3D reconstruction and pose estimation process and the advantage of virtual views on removing these inconsistencies.

Table 3. Evaluation on 2D and 3D Semantic segmentation tasks on ScanNet validation set. Ablation study evaluating the impact of sequentially adding features from our proposed virtual view fusion algorithm. The top row shows results of the traditional semantic segmentation approach with multiview fusion – where all semantic predictions are made on the original captured input images. Subsequent rows show the impact of gradually replacing characteristics of the original views with virtual ones. The bottom row shows the performance of our overall method using virtual views.

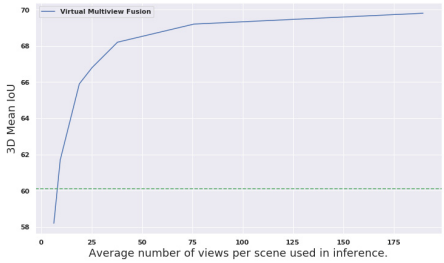
Image Type	Input Image Channels	Intrinsics	Extrinsics	2D Image IoU (Single View)	3D Mesh IoU (Multiview)	2D Image IoU (Multiview)
Real	RGB	Original	Original	60.1	60.1	68.5
Virtual	RGB	Original	Original	64.4	63.2	69.8
Virtual	RGB + Normal + Coordinates	Original	Original	66.1	66.1	70.8
Virtual	RGB + Normal + Coordinates	High FOV	Original	66.9	67.9	72.2
Virtual	RGB + Normal + Coordinates	High FOV	View sampling	67.0	70.1	74.9

Effect of Training Set Size. Our next experiment investigates the impact of the training set size on our algorithm. We hypothesize that generating large numbers of virtual views provides a form of data augmentation that improves generalizability of small training sets. To test this idea, we randomly sampled different numbers of scenes from the training set and trained our algorithm only on them. We compare performance of multiview fusion using a 2D model trained from virtual views rendered from those scenes versus from the original photographic images, as well as a 3D convolution method SparseConv (Fig. 9a). Note that we conduct the experiments on ScanNet low resolution meshes while for others we use high resolution ones. For virtual/real multiview fusion approaches, we use the same set of views for each scene across different experiments. We find that the virtual multiview fusion approach consistently outperforms 3D SparseConv and real multiview fusion even with a small number of scenes.

Effect of Number of Views at Inference. Next we investigate the impact of number of virtual views used in the inference stage on our algorithm. We run



(a) Effect of training data size on 3D segmentation IoU. Virtual multiview fusion model gives the better performance even when training data is small. Our hypothesis is that virtual view provides better data augmentation than simple 2D image level augmentations. Data augmentation is important with less training data



(b) Effect of number of views used at inference time on 3D segmentation. The dotted green line shows the best mIoU (60.1) obtained with multi-view fusion using all original views (≈ 1700 views per scene). Our virtual multiview fusion model achieves the same accuracy with just ≈ 12 views per scene.

Fig. 9. Impact of data size (number of views) during training and inference.

our virtual view selection algorithms on the ScanNet validation dataset, run a 2D model on them, and then do multiview fusion using only a random subset of the virtual views. As shown in Fig. 9b, the 3D mIoU increases with the number of virtual views with diminishing returns. The virtual multiview fusion approach is able to achieve good performance even with a significantly smaller inference set. For example, while we achieve 70.1% 3D mIoU with all virtual views (∼2000 views per scene), we can reach 61.7% mIoU even with ∼10 views per scene, and 68.2% with ∼40 views per scene. In addition, the result shows that using more views selected with the same approach as for training views does not negatively affect the multiview fusion performance, which is not obvious as the confident but wrong prediction of one single view can harm the overall performance.

7 Conclusion

In this paper, we propose a virtual multiview fusion approach to 3D semantic segmentation of textured meshes. This approach builds off a long history of representing and labeling meshes with images, but introduces several new ideas that significantly improve labeling performance: virtual views with additional channels, back-face culling, wide field-of-view, multiscale aware view sampling. As a result, it overcomes the 2D-3D misalignment, occlusion, narrow view, and scale invariance issues that have vexed most previous multiview fusion approaches.

The surprising conclusion from this paper is that multiview fusion algorithms are a viable alternative to 3D convolution for semantic segmentation of 3D textured meshes. Although early work on this task considered multiview fusion, the general approach has been abandoned in recent years in favor of 3D convolutions

of point clouds and sparse voxel grids. This paper shows that the simple method of carefully selecting and rendering virtual views enables multiview fusion to outperform almost all recent 3D convolution networks. It is also complementary to more recent 3D approaches. We believe this will encourage more researchers to build on top of this.

References

1. Armeni, I., Sax, A., Zamir, A.R., Savarese, S.: Joint 2D–3D-Semantic Data for Indoor Scene Understanding. ArXiv e-prints, February 2017
2. Armeni, I., et al.: 3D semantic parsing of large-scale indoor spaces. In: Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition (2016)
3. Boulch, A., Guerry, J., Le Saux, B., Audebert, N.: Snapnet: 3D point cloud semantic labeling with 2D deep segmentation networks. Comput. Graph. **71**, 189–198 (2018)
4. Chen, L.-C., Zhu, Y., Papandreou, G., Schroff, F., Adam, H.: Encoder-decoder with atrous separable convolution for semantic image segmentation. In: Ferrari, V., Hebert, M., Sminchisescu, C., Weiss, Y. (eds.) ECCV 2018. LNCS, vol. 11211, pp. 833–851. Springer, Cham (2018). https://doi.org/10.1007/978-3-030-01234-2_49
5. Chiang, H., Lin, Y., Liu, Y., Hsu, W.H.: A unified point-based framework for 3D segmentation. In: 2019 International Conference on 3D Vision (3DV), pp. 155–163, September 2019
6. Chollet, F.: Xception: deep learning with depthwise separable convolutions. In: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2017)
7. Choy, C., Gwak, J., Savarese, S.: 4D spatio-temporal convnets: Minkowski convolutional neural networks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 3075–3084 (2019)
8. Choy, C., Park, J., Koltun, V.: Fully convolutional geometric features. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 8958–8966 (2019)
9. Dai, A., Chang, A.X., Savva, M., Halber, M., Funkhouser, T., Nießner, M.: Scannet: richly-annotated 3D reconstructions of indoor scenes. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 5828–5839 (2017)
10. Dai, A., Nießner, M.: 3DMV: joint 3D-multi-view prediction for 3D semantic scene segmentation. In: Ferrari, V., Hebert, M., Sminchisescu, C., Weiss, Y. (eds.) ECCV 2018. LNCS, vol. 11214, pp. 458–474. Springer, Cham (2018). https://doi.org/10.1007/978-3-030-01249-6_28
11. Graham, B., Engelcke, M., van der Maaten, L.: 3D semantic segmentation with submanifold sparse convolutional networks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 9224–9232 (2018)
12. Guerry, J., Boulch, A., Le Saux, B., Moras, J., Plyer, A., Filliat, D.: Snapnet-R: consistent 3D multi-view semantic labeling for robotics. In: Proceedings of the IEEE International Conference on Computer Vision Workshops, pp. 669–678 (2017)
13. Han, L., Zheng, T., Xu, L., Fang, L.: Occuseg: occupancy-aware 3D instance segmentation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 2940–2949 (2020)

14. Hanocka, R., Hertz, A., Fish, N., Giryes, R., Fleishman, S., Cohen-Or, D.: MeshCNN: a network with an edge. ACM Trans. Graph. (TOG) **38**(4), 1–12 (2019)
15. Hermans, A., Floros, G., Leibe, B.: Dense 3D semantic mapping of indoor scenes from RGB-D images. In: 2014 IEEE International Conference on Robotics and Automation (ICRA), pp. 2631–2638. IEEE (2014)
16. Hu, Z., Zhen, M., Bai, X., Fu, H., Tai, C.l.: JSENet: joint semantic segmentation and edge detection network for 3D point clouds. In: ECCV (2020)
17. Huang, J., Zhang, H., Yi, L., Funkhouser, T., Nießner, M., Guibas, L.J.: Texturenet: consistent local parametrizations for learning from high-resolution signals on meshes. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 4440–4449 (2019)
18. Jaritz, M., Gu, J., Su, H.: Multi-view pointnet for 3D scene understanding. In: Proceedings of the IEEE International Conference on Computer Vision Workshops (2019)
19. Lai, K., Bo, L., Fox, D.: Unsupervised feature learning for 3D scene labeling. In: 2014 IEEE International Conference on Robotics and Automation (ICRA), pp. 3050–3057. IEEE (2014)
20. Landrieu, L., Simonovsky, M.: Large-scale point cloud semantic segmentation with superpoint graphs. In: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR), June 2018
21. Lawin, F.J., Danelljan, M., Tosteberg, P., Bhat, G., Khan, F.S., Felsberg, M.: Deep projective 3D semantic segmentation. In: Felsberg, M., Heyden, A., Krüger, N. (eds.) CAIP 2017. LNCS, vol. 10424, pp. 95–107. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-64689-3_8
22. Li, W., et al.: Interiornet: mega-scale multi-sensor photo-realistic indoor scenes dataset. arXiv preprint arXiv:1809.00716 (2018)
23. Li, Y., Bu, R., Sun, M., Wu, W., Di, X., Chen, B.: PointCNN: convolution on x-transformed points. In: Advances in Neural Information Processing Systems, pp. 820–830 (2018)
24. Ma, L., Stückler, J., Kerl, C., Cremers, D.: Multi-view deep learning for consistent semantic mapping with RGB-D cameras. In: 2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), pp. 598–605. IEEE (2017)
25. McCormac, J., Handa, A., Davison, A., Leutenegger, S.: Semanticfusion: dense 3D semantic mapping with convolutional neural networks. In: 2017 IEEE International Conference on Robotics and automation (ICRA), pp. 4628–4635. IEEE (2017)
26. McCormac, J., Handa, A., Leutenegger, S., Davison, A.J.: Scenenet RGB-D: can 5m synthetic images beat generic imagenet pre-training on indoor segmentation? In: Proceedings of the IEEE International Conference on Computer Vision, pp. 2678–2687 (2017)
27. Mottaghi, R., et al.: The role of context for object detection and semantic segmentation in the wild. In: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR), June 2014
28. Narita, G., Seno, T., Ishikawa, T., Kaji, Y.: Panopticfusion: online volumetric semantic mapping at the level of stuff and things. arXiv preprint arXiv:1903.01177 (2019)
29. Pham, Q.H., Nguyen, T., Hua, B.S., Roig, G., Yeung, S.K.: JSIS3D: joint semantic-instance segmentation of 3D point clouds with multi-task pointwise networks and multi-value conditional random fields. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 8827–8836 (2019)

30. Qi, C.R., Su, H., Mo, K., Guibas, L.J.: Pointnet: deep learning on point sets for 3D classification and segmentation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 652–660 (2017)
31. Qi, C.R., Yi, L., Su, H., Guibas, L.J.: Pointnet++: deep hierarchical feature learning on point sets in a metric space. In: Advances in Neural Information Processing Systems, pp. 5099–5108 (2017)
32. Riegler, G., Osman Ulusoy, A., Geiger, A.: OctNet: Learning deep 3D representations at high resolutions. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 3577–3586 (2017)
33. Shi, S., et al.: PV-RCNN: point-voxel feature set abstraction for 3D object detection. arXiv preprint arXiv:1912.13192 (2019)
34. Song, S., Yu, F., Zeng, A., Chang, A.X., Savva, M., Funkhouser, T.: Semantic scene completion from a single depth image. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 1746–1754 (2017)
35. Su, H., Maji, S., Kalogerakis, E., Learned-Miller, E.: Multi-view convolutional neural networks for 3D shape recognition. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 945–953 (2015)
36. Tatarchenko, M., Park, J., Koltun, V., Zhou, Q.Y.: Tangent convolutions for dense prediction in 3D. In: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR), June 2018
37. Tchapmi, L., Choy, C., Armeni, I., Gwak, J., Savarese, S.: Segcloud: semantic segmentation of 3D point clouds. In: 2017 International Conference on 3D Vision (3DV), pp. 537–547. IEEE (2017)
38. Thomas, H., Qi, C.R., Deschaud, J.E., Marcotegui, B., Goulette, F., Guibas, L.J.: KPConv: flexible and deformable convolution for point clouds. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 6411–6420 (2019)
39. Valada, A., Mohan, R., Burgard, W.: Self-supervised model adaptation for multimodal semantic segmentation. Int. J. Comput. Vis. 1–47 (2019)
40. Valentin, J., et al.: SemanticPaint: Interactive 3D labeling and learning at your fingertips. In: ACM Transactions on Graphics. ACM (2015)
41. Vineet, V., et al.: Incremental dense semantic stereo fusion for large-scale semantic scene reconstruction. In: 2015 IEEE International Conference on Robotics and Automation (ICRA), pp. 75–82. IEEE (2015)
42. Wang, S., Suo, S., Ma, W.C., Pokrovsky, A., Urtasun, R.: Deep parametric continuous convolutional neural networks. In: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR), June 2018
43. Wu, W., Qi, Z., Fuxin, L.: PointConv: deep convolutional networks on 3D point clouds. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 9621–9630 (2019)
44. Yan, X., Zheng, C., Li, Z., Wang, S., Cui, S.: PointASNL: robust point clouds processing using nonlocal neural networks with adaptive sampling. arXiv preprint arXiv:2003.00492 (2020)
45. Ye, X., Li, J., Huang, H., Du, L., Zhang, X.: 3D recurrent neural networks with context fusion for point cloud semantic segmentation. In: Ferrari, V., Hebert, M., Sminchisescu, C., Weiss, Y. (eds.) ECCV 2018. LNCS, vol. 11211, pp. 415–430. Springer, Cham (2018). https://doi.org/10.1007/978-3-030-01234-2_25
46. Zhang, C., Liu, Z., Liu, G., Huang, D.: Large-scale 3D semantic mapping using monocular vision. In: 2019 IEEE 4th International Conference on Image, Vision and Computing (ICIVC), pp. 71–76. IEEE (2019)

47. Zhang, Y., et al.: Physically-based rendering for indoor scene understanding using convolutional neural networks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 5287–5295 (2017)
48. Zhou, B., Zhao, H., Puig, X., Fidler, S., Barriuso, A., Torralba, A.: Scene parsing through ADE20K dataset. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 633–641 (2017)