Cascaded Context Pyramid for Full-Resolution 3D Semantic Scene Completion

Pingping Zhang[†]   Wei Liu[‡]   Yinjie Lei[§]   Huchuan Lu[†*]   Xiaoyun Yang[⋆]

[†]Dalian University of Technology,   [‡]University of Adelaide,   [§]Sichuan University,   [⋆]China Science IntelliCloud Technology Co.Ltd

jssxzhpp@mail.dlut.edu.cn   wei.liu02@adelaide.edu.au   yinjie@scu.edu.cn   lhchuan@dlut.edu.cn   xiaoyun.yang@intellicloud.ai

Abstract

Semantic Scene Completion (SSC) aims to simultaneously predict the volumetric occupancy and semantic category of a 3D scene. It helps intelligent devices to understand and interact with the surrounding scenes. Due to the high-memory requirement, current methods only produce low-resolution completion predictions, and generally lose the object details. Furthermore, they also ignore the multi-scale spatial contexts, which play a vital role for the 3D inference. To address these issues, in this work we propose a novel deep learning framework, named Cascaded Context Pyramid Network (CCPNet), to jointly infer the occupancy and semantic labels of a volumetric 3D scene from a single depth image. The proposed CCPNet improves the labeling coherence with a cascaded context pyramid. Meanwhile, based on the low-level features, it progressively restores the fine-structures of objects with Guided Residual Refinement (GRR) modules. Our proposed framework has three outstanding advantages: (1) it explicitly models the 3D spatial context for performance improvement; (2) full-resolution 3D volumes are produced with structure-preserving details; (3) light-weight models with low-memory requirements are captured with a good extensibility. Extensive experiments demonstrate that in spite of taking a single-view depth map, our proposed framework can generate high-quality SSC results, and outperforms state-of-the-art approaches on both the synthetic SUNCG and real NYU datasets.

1. Introduction

Human can perceive the real-world through 3D views with partial observations. For example, one can capture the geometry of rigid objects by only seeing the corresponding 2D images. Thus, understanding and reconstructing a 3D scene from its partial observations is a valuable technique for many computer vision and robotic applications, such as object localization, visual reasoning and indoor navigation. As an encouraging direction, Semantic Scene Completion

---

[*]Corresponding author.

(SSC) has draw more and more attentions in recent years. It aims to simultaneously predict the volumetric occupancy and semantic category of a 3D scene. Given a single depth image, several outstanding works [32, 10, 36] have been proposed for single-view SSC. By designing 3D Convolutional Neural Networks (CNNs), these methods can automatically predict the semantic labels or complete 3D shapes of the objects in the scene. However, it is not a trivial task to utilize 3D CNNs for the SSC task. Vanilla 3D CNNs are locked in the cubic growth of computational and memory requirements with the increase of voxel resolution. Thus, current methods inevitably limit the resolution of predictions and the depth of 3D CNNs, which leads to wrong labels and missing shape details in the completion results.

To achieve better SSC results, several works [8, 5, 23, 20] introduce the 2D semantic segmentation as an auxiliary, which takes an additional RGB image and applies complex 2D CNNs for semantic enhancement. These methods can fully exploit the high-resolution input, however, they ignore the 3D context information of the scene. Thus, only based on the 2D input image, they may not infer the invisible object parts of the complex scene. Recently, Song *et al.* [32] show that global 3D context helps the prediction of SSC. However, the 3D CNN used in their work simply adopts the dilated convolutions [34], and concatenates the multi-stage features for predictions. It only considers the global semantics, which result in low-resolution predictions, and lose the scene details. In this work, we find that both *local geometric details* and *multi-scale 3D contexts* of the scene play a vital role in the SSC task. The local geometric details help the SSC system to identify the fine-structured objects. The multi-scale 3D contexts can enhance the spatial coherence and infer the occluded objects from the scene layout. However, designing a framework that can efficiently integrate both characteristics is still a challenging task.

To address above problems, we propose a novel deep learning framework, named Cascaded Context Pyramid Network (CCPNet), for single depth image based SSC. The proposed CCPNet effectively learns both local geometry details and multi-scale 3D contexts from the training dataset. For semantic confusing objects, the CCPNet improves the

prediction coherence with an effective self-cascaded context pyramid. The self-cascaded pyramid helps the model to reduce the semantic gap of different contexts and capture the hierarchical dependencies among the objects and scenes [24]. In addition, we introduce a Guided Residual Refinement (GRR) module to progressively restore the fine-structures of complex objects. The GRR corrects the latent fitting using low-level features, and avoids the high computational cost and memory consumption of the 3D CNN. With this module, the CCPNet can output full-resolution completion results and show much better accuracy than vanilla 3D networks. Experimental results demonstrate that our approach outperforms other state-of-the-art methods on both synthetic and real datasets. With only a single depth map, our method generates high-quality SSC results with much better accuracy and faster inference.

In summary, **our contributions** are three folds:

- We propose a novel cascaded context pyramid network (CCPNet) for efficient 3D semantic scene completion. The CCPNet automatically integrates both local geometric details and multi-scale 3D contexts of the scene in a self-cascaded manner.

- We also propose an efficient guided residual refinement (GRR) module for restoring fine-structures of objects and full-resolution predictions. The GRR progressively refines the objects with low-level features and light-weight residual connections, improving both computational efficiency and completion accuracy.

- Extensive experiments on public synthetic and real benchmarks demonstrate that our proposed approach achieves superior performance over other state-of-the-art methods.

2. Related Work

In this section, we briefly review related work on analyzing and completing a 3D scene from depth images. For more details, we refer the readers to [17] for a survey of deep learning based 3D data processing.

Semantic Scene Analysis. In recent years, many deep learning based methods have been proposed for semantic scene analysis with a depth image or RGB-D image pair. In general, 2D image-based methods [26, 12, 33] treat the depth image as additional information, and adopt complex 2D CNNs for semantic scene analysis tasks, *e.g.*, salient object detection, semantic segmentation and scene completion. Meanwhile, several works [13, 11, 1] extract deep features from the depth image and the RGB image separately, then fuse them for multi-mode complementarity. Although effective, 2D image-based methods ignore the spatial occupancy of objects, and can not fully exploit the depth information. While 3D volume-based methods usually convert

the depth image into a volumetric representation, and exploit rich handcrafted 3D features [27, 30] or learned 3D C-NNs [31] for detecting 3D objects. Although existing methods can detect and segment visible 3D objects and scenes, they cannot infer the objects that are totally occluded. Instead, our method can predict the semantic labels and 3D shapes for both visible and invisible objects.

3D Scene Completion. Semantic scene completion is a fundamental task in understanding 3D scenes. To achieve this goal, Zheng *et al.* [41] first complete the occluded objects with a set of pre-defined rules, and then refine the completion results by physical reasoning. Geiger and Wang [6] propose a high-order graphical model to jointly reason about the layout, objects and superpixels in the scene image. Their model leverages detailed 3D geometry of scenes, and explicitly enforces occlusion and visibility constraints. Then, Firman *et al.* [4] utilize the random forest to infer the occluded 3D object shapes from a single depth image. These methods are based on handcrafted features, and perform semantic scene segmentation and completion in two separate steps. Recently, Song *et al.* [32] propose the Semantic Scene Completion Network (SSCNet) to simultaneously predict the semantic labels and volumetric occupancy of the 3D objects from a single depth image. Although this method unifies the semantic segmentation and voxel completion, the expensive 3D CNN limits the input resolution and network depth. Thus the SSCNet only produces low-resolution predictions and generally lacks of object details. By combining the 2D CNN and 3D CNN, Guo and Tong [10] propose the View-Volume Network (VVNet) to efficiently reduce the computation cost and enhance the network depth. Garbade *et al.* [5] propose a two-stream approach that jointly leverages the depth and semantic information. They first construct an incomplete 3D semantic tensor for the inferred 2D semantic information, and then adopt a vanilla 3D CNN to infer the complete 3D semantic tensor. Liu *et al.* [23] propose a task-disentangled framework to sequentially carry out the 2D semantic segmentation, 2D-3D re-projection and 3D semantic scene completion. However, their multi-stage method may cause the error accumulation, producing mislabeling completion results. Similarly, Li *et al.* [20] introduce a Dimensional Decomposition Residual Network (DDRNet) for the 3D SSC task. Based on the factorized and dilated convolutions [2], they utilize the multi-scale feature fusion mechanism for depth and color images.

Although effective, current methods only consider the global semantics, which usually result in low-resolution predictions and lose the scene details. Different from previous works, we propose to integrate both local geometric details and multi-scale 3D contexts of the scene for the SSC task. To reduce the semantic gaps of multi-scale 3D contexts, we propose a self-cascaded context aggregation method to generate coherent labeling results. Meanwhile,
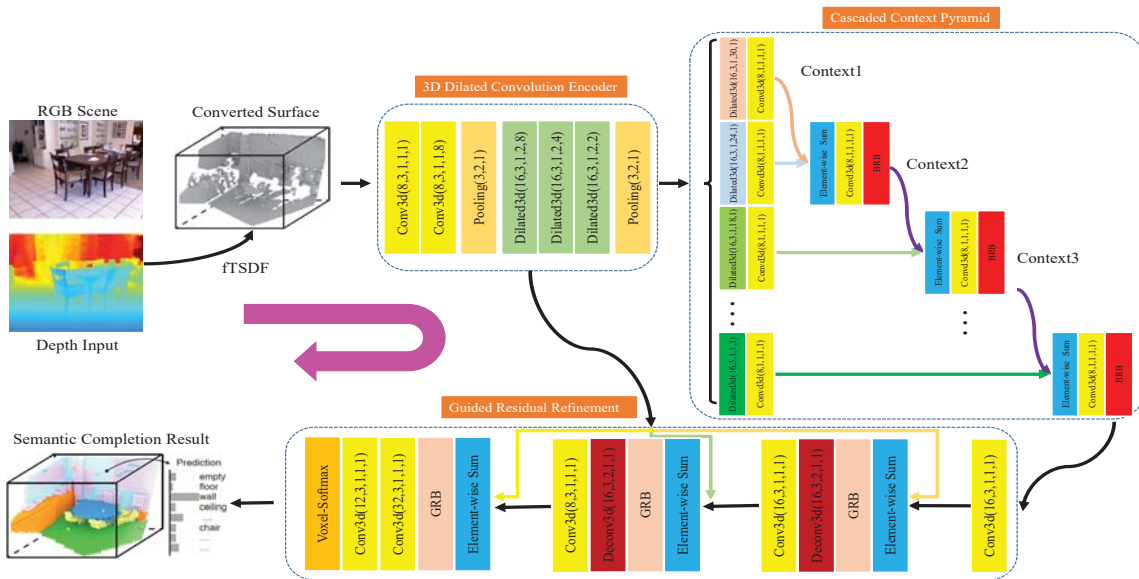
Figure 1. Illustration of our Cascaded Context Pyramid Network (CCPNet). Taking a single-view depth map as input, the CCPNet predicts the occupancy and object labels for each voxel in the view frustum. With light-weight operations, the CCPNet can produce full-resolution 3D completion results. The convolution parameters are shown as (number of filters, kernel size, stride, dilation, number of subvolumes).

the local geometric details are also incorporated to identify the fine-structured objects in a coarse-to-fine manner. We note that the proposed modules are general-purpose for 3D CNNs. Thus, they can be easily applied to other 3D tasks.

3. Cascaded Context Pyramid Network

Fig. 1 illustrates the overall architecture of our CCPNet. Given a single-view depth map of a 3D scene, the goal of our CCPNet is to map the voxels in the view frustum to one of the semantic labels $C = [c_0, c_1, ..., c_{N+1}]$, where N is number of semantic categories and c_0 stands for empty voxels. Our CCPNet is a self-cascaded pyramid structure to successively aggregate multi-scale 3D contexts and local geometry details for full-resolution scene completions. It consists of three key components, *i.e.*, 3D Dilated Convolution Encoder (DCE), Cascaded Context Pyramid (CCP), and Guided Residual Refinement (GRR). Functionally, the DCE adopts multiple dilated convolutions with separated kernels to extract 3D feature representations from single-view depth images. Then, the CCP performs the sequential global-to-local context aggregation to improve the labeling coherence. After the context aggregation, the GRR is introduced to refine the target objects using low-level features learned by the shallow layers. In the following subsections, we will describe these components in detail.

3.1. 3D Dilated Convolution Encoder

Input Tensor Generation. For the input of our front-end 3D DCE, we follow previous works [32, 5, 10] and rotate the 3D scene to align with the gravity and room orien-

tation based on the Manhattan assumption. We consider the absolute dimensions of the 3D space with 4.8 m horizontally, 2.88 m vertically, and 4.8 m in depth. Each 3D scene is encoded into a flipped Truncated Signed Distance Function (fTSDF) [32] with grid size 0.02 m, truncation value 0.24 m, resulting in a $240 \times 144 \times 240$ tensor as the network input. Our method produces the completion result with the same resolution as input. However, due to the fully convolutional structure and the light-weight network design, our method certainly can take larger depth images as input, even full-resolution depth maps (e.g., 427×561 from depth sensors). During the model training, we render depth maps from virtual viewpoints of 3D scenes and voxelize the full 3D scenes with object labels as ground truth.

Encoder Structure. Processing 3D data needs large memories and huge computations. To reduce the memory-requirement, we propose a light-weight encoder to extract the 3D feature representations of scenes, as shown in Fig. 1. As demonstrated in dense labeling tasks [40, 39, 2, 37], large contexts can provide valuable information for understanding the scenes. For the 3D scenes and depth images, spatial context is more useful due to the lack of high frequency signals. To effectively learn spatial contextual information, we make sure our encoder has a big enough receptive field. A direct method is using the 3D dilated convolution proposed in [34, 32], which can exponentially expand the receptive field without a loss of resolution or coverage. However, the computation of 3D dilated convolutions is rather huge, because we need to perform convolutions with large volumes. To address this problem, we propose
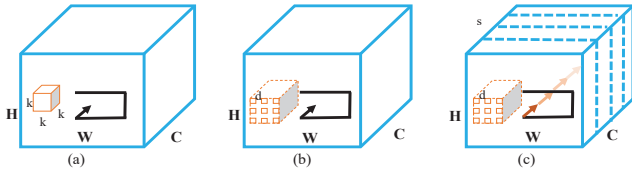
Figure 2. Comparison of (a) Vanilla 3D convolution [18], (b) 3D dilated convolution [32] and (c) Our proposed method.

the 3D dilated convolutions with *separated kernels*. More specifically, we first separate the input tensor into several subvolumes, then apply the 3D dilated kernels to each subvolume for the convolutions. The reasons are two-fold. On the one hand, our method can reduce the model parameters and computations, and inherit all characteristics of dilated convolutions. On the other hand, our method considers the characteristic of depth profiles, in which the depth values are continuous only in neighbour regions. Fig. 2 shows the differences of vanilla 3D convolution [18], 3D dilated convolution [32] and our proposed method. To build our 3D DCE, we stack the proposed 3D dilated convolution several times with 3D pooling. Besides, to avoid the extreme separation, we reduce the number of subvolumes along with the network depth. The detailed parameters are shown in Fig. 1.

3.2. Cascaded Context Pyramid

For scene completion, different objects have very different physical 3D sizes and visual orientations. This implies that the model needs to capture information at different contexts in order to recognize objects reliably. Besides, for confusing manmade objects in indoor scenes, obtaining coherent labeling results is not easily accessible, because they are of high intra-class variance and low inter-class variance. Therefore, it is insufficient to use only the single-scale and global information of the target objects [32, 24]. We need to introduce multi-scale context information, which characterizes the underlying dependencies between an object and its surroundings. However, it is very hard to retain the hierarchical dependencies in contexts of different scales, using common fusion strategies (*e.g.*, direct stack [2, 40]). To address this issue, we propose a novel self-cascaded context pyramid architecture, as shown in Fig. 3 (a). Different from previous methods, our method sequentially aggregates the global-to-local contexts while well retains the hierarchical dependencies, *i.e.*, the underlying inclusion and location relationship among the objects and scenes in different scales.

Architecture Details. To build the context pyramid, we perform 3D dilated convolutions on the last pooling layer of the 3D DCE to capture multi-scale contexts. By setting varied dilation rates (30, 24, 18, 12, 6 and 1 in the experiments) and feature reduction layers, a series of 3D feature maps with global-to-local contexts are generated. The large-scale context contains more semantics and wider visual cues, while the small-scale context retains object ge-
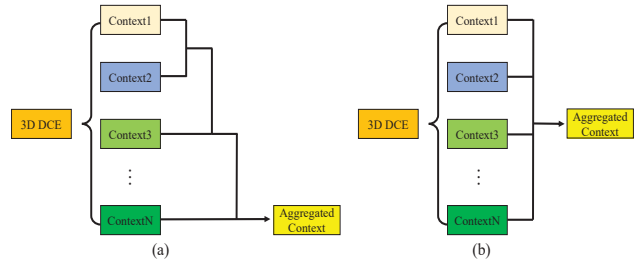

Figure 3. Comparison of different multi-scale context aggregation methods. (a) Our self-cascaded context aggregation approach, which reduces the semantic gaps of different scales. (b) Existing parallel concatenations, such as PSPNet [40], Deeplab variants [2]. "Context" denotes the dilated convolution for context extraction.

ometry details. Meanwhile, the obtained feature maps with multi-scale contexts can be aligned automatically due to their equal resolution. To well retain the hierarchical dependencies of multi-scale contexts, we *sequentially* aggregate them in a self-cascaded pyramid manner. Formally, it can be described as:

$$X_{sa} = \begin{cases} f(\cdots f(f(X_1 \oplus X_2) \oplus X_3) \oplus \cdots \oplus X_n), \\ d_1 > d_2 > d_3 > \cdots > d_n. \end{cases} \quad (1)$$

where X_n denotes the n-scale context, X_{sa} is the final aggregated context and d_n is the dilation rate for extracting the context X_n. $\oplus$ denotes the element-wise summation. f denotes the Basic Residual Block (BRB) [16], as shown in Fig. 4 (a). In our proposed method, we first aggregate the large-scale context with big dilation rates, then the context with small dilation rates. This aggregation rule is consistent with the human visual mechanism, *i.e.*, large-scale context could play a guiding role in integrating small-scale context.

We also notice that there are other outstanding structures for multi-scale contexts, such as PPM [40] and ASPP [2], as shown in Fig. 3 (b). In order to aggregate information with different contexts, they add a layer that *parallelly* concatenates the feature maps with different receptive fields:

$$X_{pa} = \begin{cases} g([X_1, X_2, X_3, \cdots, X_n]), \\ d_1 > d_2 > d_3 > \cdots > d_n. \end{cases} \quad (2)$$

where g denotes the aggregation function, which usually is an $1 \times 1 \times 1$ convolutional layer. $[\cdots]$ is the concatenation operation in channel-wise. However, our proposed self-cascaded pyramid architecture has several advantages: 1) Our self-cascaded strategy enhances the hierarchical dependencies in different context scales. Thus, it is more effective than the parallel strategies such as PSPNet [40], DeepLab variants [2], which directly fuse the multi-scale contexts with large semantic gaps; 2) Our method introduces more complicated nonlinear operations (Equ. 1), thus it has a stronger capacity to model the relationship of different contexts than simple convolution operations. 3) By adopting
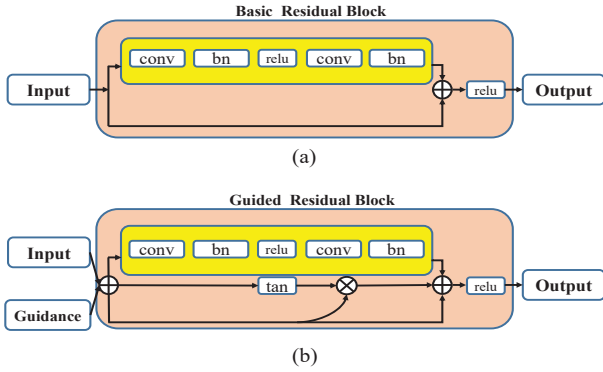
Figure 4. The used residual modules in our CCPNet. (a) The Basic Residual Block (BRB) [16]. (b) The proposed Guided Residual Block (GRB). In the GRB, we add a tangent function-based connection to amplify the fused features.

the summation, the sequential aggregation significantly reduces the parameters and computations. Experiments also verify the effectiveness of our proposed method.

3.3. Guided Residual Refinement

Besides semantic confusing categories, fine-structured objects also increase the difficulty for accurate labeling in 3D scenes. However, current methods usually produce low-resolution predictions, thus it is very hard to retain the fine-grained details of objects. To address this problem, we propose to reuse low-level features with the Guided Residual Refinement (GRR), as shown in the bottom of Fig. 1. Specifically, the rich low-level features are progressively reintroduced into the prediction stream by guided residual connections. As a result, the coarse feature maps can be refined and the low-level details can be restored for full-resolution predictions. The used Guided Residual Block (GRB) is shown in Fig. 4 (b), which can be formulated as:

$$\hat{X} = X \oplus G, \tag{3}$$

$$X_{rf} = ReLu(\hat{X} \oplus \hat{X}Tanh(\hat{X}) \oplus h(\hat{X})) \tag{4}$$

$$= ReLu(\hat{X}(I \oplus Tanh(\hat{X})) \oplus h(\hat{X})) \tag{5}$$

$$= ReLu(\hat{X}_G \oplus h(\hat{X})). \tag{6}$$

where X is the input semantic context feature and G is the guidance feature coming from a shallower layer. $\oplus$ denotes the element-wise summation and h is the standard non-linear transform in residual blocks. X_{rf} is the refined feature map. $ReLu(\cdot)$ and $Tanh(\cdot)$ are the rectified linear unit and hyperbolic tangent activation, respectively. To restore finer details with the shallower layer, we first integrate the input feature and the guidance (Equ. 3), then we introduce an auxiliary connection to the BRB [16]. More specifically, we use the hyperbolic tangent activation to amplify the integrated features (resulting in $\hat{X}_G$), as shown in Fig. 4 (b) and Equ. 4-6. It is very beneficial to fuse low-level features by the guided refinement strategy. On the one hand,

the feature maps of X and G represent different semantics at varied levels. Thus, due to their inherent semantic gaps, directly stacking all these features [14, 28, 3] may not be an efficient strategy. In the proposed method, the influence of semantic gaps is alleviated when a residual iteration strategy is adopted [7]. On the other hand, the feature amplification connection enhances the effect of low-level details and gradient propagations, which helps the effectively end-to-end training. There also exist effective refinement strategies for detail enhancement [25, 22, 38]. However, they are very different from ours. First, our strategy focuses on amplifying low-level features considering the 3D data properties, *e.g.*, high computation and memory requirements. In contrast, previous methods introduce complex refinement modules, which are hardly executable for the 3D data. Besides, we only choose specific shallow layers for the refinement, as shown in the bottom of Fig. 1. Other methods incorporate all the hierarchical layers that inevitably contain boundary noises [25, 22]. To build our model, several GRB modules are elaborately embedded in the prediction part, which can greatly prevent the fitting residual from accumulating. As a result, the proposed CCPNet effectively works in a coarse-to-fine labeling manner for full-resolution predictions.

3.4. Network Training

Given the training dataset (*i.e.*, the paired depth images and ground truth volumetric labels of 3D scenes), our proposed CCPNet can be trained in an end-to-end manner. We adopt the voxel-wise softmax loss function [32] for the network training. The loss can be expressed as:

$$L(p,y) = \sum_{i,j,k} w_{ijk} L_{sm}(p_{ijk}, y_{ijk}), \tag{7}$$

where L_{sm} is the softmax cross-entropy loss, y_{ijk} is the ground truth label, p_{ijk} is the predicted probability of the voxel at coordinates (i, j, k). The weight $w_{ijk} \in \{0, 1\}$ is used to balance the loss between different semantic categories. Due to the sparsity of 3D data, the ratio of empty vs. occupied voxels is extremely imbalanced. To address this problem, we follow [32] and randomly sample the training voxels with a 2:1 ratio to ensure that each mini-batch has a balanced set of empty and occupied examples.

4. Experiments

4.1. Experimental Settings

Datasets. The synthetic **SUNCG** dataset [31] consists of 45622 indoor scenes. Technically, the depth images and semantic scene volumes can be acquired by setting different camera orientations. Following previous useful works [32, 5, 10], we adopt the same training/test split for our network training and evaluation. More specifically, the training set contains about 150K depth images and the corresponding

Methods	scene completion			semantic scene completion											
	prec.	recall	IoU	ceil.	floor	wall	win.	chair	bed	sofa	table	tvs	furn.	objs.	avg.
SSCNet [32]	76.3	95.2	73.5	96.3	84.9	56.8	28.2	21.3	56.0	52.7	33.7	10.9	44.3	25.4	46.4
VVNet [10]	90.8	91.7	84.0	98.4	87.0	61.0	54.8	49.3	83.0	75.5	55.1	43.5	68.8	57.7	66.7
DCRF [36]	–	–	–	95.4	84.3	57.7	24.5	28.2	63.4	55.3	34.5	19.6	45.8	28.7	48.8
ESSCNet [35]	92.6	90.4	84.5	96.6	83.7	74.9	59.0	55.1	83.3	78.0	61.5	47.4	73.5	62.9	70.5
SATNet [23]	80.7	96.5	78.5	97.9	82.5	57.7	58.5	45.1	78.4	72.3	47.3	45.7	67.1	55.2	64.3
Ours	**98.2**	**96.8**	**91.4**	**99.2**	**89.3**	**76.2**	**63.3**	**58.2**	**86.1**	**82.6**	**65.6**	**53.2**	**76.8**	**65.2**	**74.2**

Table 1. The performances of different scene completion methods on the SUNCG dataset. The best results are in bold.

Ceil floor wall window Chair bed sofa table tvs furn objects
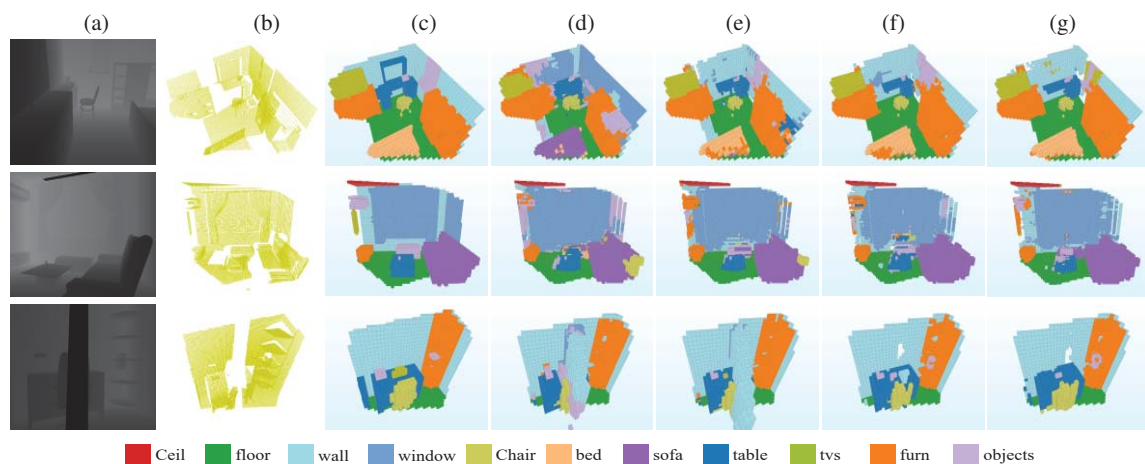
Figure 5. Completion results with different methods on the SUNCG dataset. From the left to right: (a) Input Depth; (b) fTSDF Surface; (c) Ground Truth; (d) SSCNet [32]; (e) VVNet [10]; (f) ESSCNet [35]; (g) Ours. It can be observed that, our results constantly contain more accurate and detailed structures compared to the baselines. The figure is best viewed in color with 200% zooming-in.

ground truth volumes. The test set consists of totally 470 pairs sampled from 170 non-overlap scenes.

The real **NYU** dataset [29] includes 1449 depth images captured by the Kinect depth sensor. The depth images are partitioned into 795 for training and 654 for test. Following previous works, we adopt the ground truth completion and segmentation from [9]. Some labeled volumes and their corresponding depth images are not well aligned in the NYU dataset. Thus, we also use the NYU CAD dataset [4], in which the depth map is rendered from the label volume. The NYU dataset is challenging due to the unavoidably measurement errors in the depth images collected by Kinect. As in [32, 10, 23], we also pre-train the network on the SUNCG dataset before fine-tuning it on the NYU dataset.

Evaluation Metrics. We mainly follow [32] and report the precision, recall, and Intersection over Union (IoU) of the compared methods. The IoU measures the overlapped ratio between intersection and union of the positive prediction volume and the ground truth volume. In this work, two tasks are considered: Scene Completion (SC) and Semantic Scene Completion (SSC). For the SC task, we treat all voxels as binary predictions, *i.e.*, occupied or non-occupied. The ground truth volume includes all the occluded voxels in the view frustum. For the SSC task, we report the IoU of each class, and average them to get the mean IoU.

Training Protocol. We implement our CCPNet in the modified Caffe toolbox [19] for 3D data processing. We perform experiments on a quad-core PC with an Intel i4790 CPU and one NVIDIA TITAN X GPU (12G memory). For the CCPNet, we initialize the weights by the "msra" method [15]. During the training, we use the standard SGD method with a batch size 4, momentum 0.9 and weight decay 0.0005. We set the base learning rate to 0.01. For the SUNCG dataset, we train the CCPNet with 200K iterations and change the learning rate to 0.001 after 150K iterations. To reduce the performance bias, we evaluate the results at every 5K steps after 180K iterations, and average them as the final results. For both the NYU Kinect and NYU CAD datasets, we follow previous works [32, 10, 5, 35, 23], and fine-tune the CPPNet pre-trained from the SUNCG dataset with 10K iterations. After that, we test the models at every 2K iterations and pick the best one as the final result.

4.2. Experimental Results

4.2.1 Comparison on the SUNCG dataset.

For the SUNCG dataset, we compare our proposed CCPNet with SSCNet [32], VVNet [10], DCRF [36], ESSCNet [35] and SATNet [23] for both SC and SSC tasks. As shown in Tab. 1, our approach achieves the best performance in both SC and SSC tasks. Compared to the SSCNet, the overall IoUs of our CCPNet significantly increase about 18% and 28% for SC and SSC tasks, respectively. In spite of taking a single depth map, our approach gets higher IoUs than the RGB-D based SATNet (Ours 91.4% vs. SATNet 78.5%).

		scene completion			semantic scene completion											
Methods	Trained on	prec.	recall	IoU	ceil.	floor	wall	win.	chair	bed	sofa	table	tvs	furn.	objs.	avg.
Lin et al. [21]	NYU	58.5	49.9	36.4	0.0	11.7	13.3	14.1	9.4	29.0	24.0	6.0	7.0	16.2	1.1	12.0
Geiger and Wang [6]	NYU	65.7	58.0	44.4	10.2	62.5	19.1	5.8	8.5	40.6	27.7	7.0	6.0	22.6	5.9	19.6
SSCNet [32]	NYU	57.0	94.5	55.1	15.1	94.7	24.4	0.0	12.6	32.1	35.0	13.0	7.8	27.1	10.1	24.7
SSCNet [32]	SUNCG	55.6	91.9	53.2	5.8	81.8	19.6	5.4	12.9	34.4	26.0	13.6	6.1	9.4	7.4	20.2
SSCNet [32]	NYU+SUNCG	59.3	92.9	56.6	15.1	94.6	24.7	10.8	17.3	53.2	45.9	15.9	13.9	31.1	12.6	30.5
CSSCNet [8]	NYU+SUNCG	62.5	82.3	54.3	–	–	–	–	–	–	–	–	–	–	–	27.5
VVNet [10]	NYU+SUNCG	69.8	83.1	61.1	19.3	94.8	28.0	12.2	19.6	57.0	50.5	17.6	11.9	35.6	15.3	32.9
DCRF [36]	NYU	–	–	–	18.1	92.6	27.1	10.8	18.8	54.3	47.9	17.1	15.1	34.7	13.0	31.8
TS3D,V2 [5]	NYU	65.7	87.9	60.4	8.9	94.0	26.4	16.1	14.2	53.5	45.8	16.4	13.0	32.9	12.7	30.4
TS3D,V3+ [5]	NYU	64.9	88.8	60.2	8.2	94.1	26.4	19.2	17.2	55.5	48.4	16.4	22.0	32.9	17.1	32.6
ESSCNet [35]	NYU	71.9	71.9	56.2	17.5	75.4	25.8	6.7	15.3	53.8	42.4	11.2	0.0	33.4	11.8	26.7
SATNet [23]	NYU+SUNCG	67.3	85.8	60.6	17.3	92.1	28.0	16.6	19.3	57.5	53.8	17.7	18.5	38.4	18.9	34.4
DDRNet [20]	NYU	71.5	80.8	61.0	21.1	92.2	33.5	6.8	14.8	48.3	42.3	13.2	13.9	35.3	13.2	30.4
Ours	NYU	74.2	90.8	63.5	23.5	96.3	35.7	20.2	25.8	61.4	56.1	18.1	28.1	37.8	20.1	38.5
Ours	NYU+SUNCG	**78.8**	**94.3**	**67.1**	**25.5**	**98.5**	**38.8**	**27.1**	**27.3**	**64.8**	**58.4**	**21.5**	**30.1**	**38.4**	**23.8**	**41.3**

Table 2. The performances of different scene completion methods on the NYU Kinect dataset. The best results are in bold.

		scene completion			semantic scene completion											
Methods	Trained on	prec.	recall	IoU	ceil.	floor	wall	win.	chair	bed	sofa	table	tvs	furn.	objs.	avg.
Zheng et al. [41]	NYU	60.1	46.7	34.6	–	–	–	–	–	–	–	–	–	–	–	–
Firman et al. [4]	NYU	66.5	69.7	50.8	–	–	–	–	–	–	–	–	–	–	–	–
SSCNet [32]	NYU	75.0	92.3	70.3	–	–	–	–	–	–	–	–	–	–	–	–
SSCNet [32]	NYU+SUNCG	75.4	**96.3**	73.2	32.5	92.6	40.2	8.9	33.9	57.0	59.5	28.3	8.1	44.8	25.1	40.0
VVNet [10]	NYU+SUNCG	86.4	92.0	80.3	–	–	–	–	–	–	–	–	–	–	–	–
DCRF [36]	NYU	–	–	–	35.5	92.6	52.4	10.7	40.0	60.0	62.5	34.0	9.4	49.2	26.5	43.0
TS3D,V2 [5]	NYU	81.2	93.6	76.9	33.9	93.4	47.0	26.4	27.9	61.7	51.7	27.6	27.3	44.4	21.8	42.1
TS3D,V3+ [5]	NYU	80.2	94.4	76.5	34.4	93.6	47.7	31.8	32.2	65.2	54.2	30.7	32.5	50.1	30.7	45.7
DDRNet [20]	NYU	88.7	88.5	79.4	54.1	91.5	56.4	14.9	37.0	55.7	51.0	28.8	9.2	44.1	27.8	42.8
Ours	NYU	91.3	92.6	82.4	56.2	94.6	58.7	35.1	44.8	68.6	65.3	37.6	35.5	53.1	35.2	53.2
Ours	NYU+SUNCG	**93.4**	91.2	**85.1**	**58.1**	**95.1**	**60.5**	**36.8**	**47.2**	**69.3**	**67.7**	**39.8**	**37.6**	**55.4**	**37.6**	**55.0**

Table 3. The performances of different scene completion methods on the NYU CAD dataset. The best results are in bold.

Our approach also perform better than the previous best ES-SCNet with a considerable margin. Tab. 1 also lists the IoU for each object category. Our approach also achieves the highest IoUs in each category. Thus, the quantitative results demonstrate that our approach is superior in 3D SSC. Fig. 5 illustrates the qualitative results on the SUNCG dataset. Although previous methods works well for many scenes, they usually fail in the objects which have complex structures and confusing semantics (the first and second rows). In contrast, our method leverages the low-level features and multi-scale contexts to overcome these difficulties.

4.2.2 Comparison on the NYU dataset.

For the NYU dataset, we compare our CCPNet with other outstanding methods. Tab. 2 and Tab. 3 illustrate the performances on the NYU Kinect and NYU CAD datasets, respectively. From the results, we can see that our CCP-Net also achieves the best performance. For the SC task, it outperforms the SSCNet (8.4% on NYU Kinect and 12.1% on NYU CAD) when only the NYU dataset is used as the training data. Meanwhile, even the SSCNet uses the additional SUNCG training dataset, our CCPNet still achieves a substantial improvement (7% on NYU Kinect and 9.2% on NYU CAD). We observe that the SSCNet achieves a rather high recall but a low precision for the SC task. Our model pre-trained with the SUNCG dataset achieves better performances, and outperforms previous best methods, *i.e.*, VVNet and SATNet, with a large margin.

For the SSC task, our approach achieves 41.3% on NYU Kinect and 55.0% on NYU CAD, and outperforms the SS-CNet [32] by 10.8% and 15%, respectively. With the same training data, our approach constantly performs better than existing best methods with a considerable margin. Tab. 2 and Tab. 3 also include the results of each category. In general, our method tends to predict more occluded voxels than previous methods, such as window, chair and furniture. Fig. 6 shows the qualitative results in which cluttered scene completions can be observed. Our method performs substantially better than other approaches.

4.2.3 Efficiency Analysis

Current methods usually depend on expensive 3D CNNs and feature concatenations, while our CCPNet utilizes a light-weight 3D dilated encoder and a self-cascaded pyramid. Thus, it significantly reduces memory requirement and computational cost for inference. Tab. 4 lists the parameters and computations of different methods. Our CCPNet achieves much better accuracy, and significantly reduces the model parameters, and speeds up for inference.

4.3. Ablation Studies

To verify the effect of our proposed modules, we also perform ablation experiments on the SUNCG dataset.

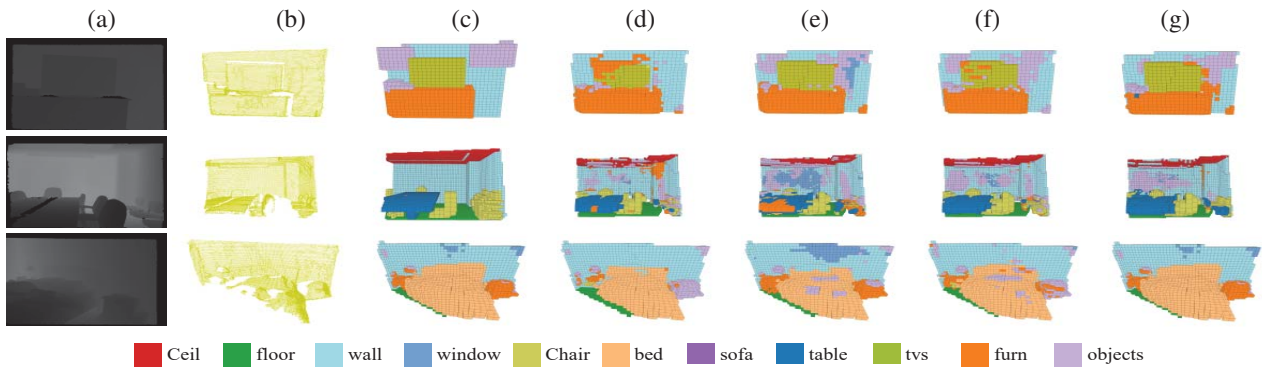**Separated Convolution Kernels.** Based on the SSC-Net [32], we replace the 3D dilated convolutions of SSCNet

Figure 6. Completion results with different methods on the NYU dataset. From the left to right: (a) Input Depth; (b) fTSDF Surface; (c) Ground Truth; (d) SSCNet [32]; (e) DDRNet [20]; (f) VVNet [10]; (g) Ours. The figure is best viewed in color with 200% zooming-in.

Methods	Params/k	FLOPs/G	Speed/ms	SC-IoU	SSC-IoU
SSCNet [32]	930	163.8	578	55.1	24.7
VVNet [10]	685	119.2	74	61.1	32.9
ESSCNet [35]	160	22.0	121	56.2	26.7
SATNet [23]	1200	187.5	1300	60.6	34.4
DDRNet [20]	195	27.2	658	61.0	30.4
Ours	**89**	**11.8**	**57**	**67.1**	**41.3**

Table 4. Comparison of efficiency with different methods.

Methods	SC-IoU	SSC-IoU	Params/k	FLOPs/G
SSCNet [32]	73.5	46.4	930	163.8
SSCNet [32]+SK	**76.8**	**52.5**	**532**	**100.3**

Table 5. Quantitative results on separated convolution kernels.

Methods	SC-IoU	SSC-IoU	Params/k	FLOPs/G
CCPNet	**91.4**	**74.2**	89	11.8
CCPNet (CCP→ PPM)	87.3	71.6	120	87.2
CCPNet (CCP→ ASPP)	89.1	72.3	145	140.2
CCPNet (GRB→BRB)	83.3	65.8	89	9.2
CCPNet (GRB w/o Ampli)	88.7	73.6	89	11.5
CCPNet (GRB w/o Guidance)	84.3	67.4	89	11.2
CCPNet-Quarter	86.5	69.1	**76**	**6.5**
CCPNet-Half	88.4	73.1	81	10.4

Table 6. Ablation results of components on the SUNCG dataset.

with our proposed separated kernels. For simplification, we set the number of subvolumes to 4. Tab. 5 shows the quantitative performances. For SC and SSC tasks, our method has fewer parameters and computations, while provides 3.3% and 6.1% IoU improvements compared to the SSCNet.

Cascaded Context Pyramid. To verify the effect of our CCP, we replace the CCP with the outstanding PPM [40] and ASPP [2] modules, and keep other modules unchanged. The first three rows of Tab. 6 show the quantitative results. With the PPM and ASPP, the IoUs of the CCPNet decrease 4.1% and 2.3% for the SC task, respectively. For the SSC task, it has a similar trend, which proves that our CCP is more effective. Note that the PPM and ASPP need more memories and parameters for the context aggregation.

Guided Residual Refinement. To evaluate the effect of our GRR, we compare the performances with different refinements. As shown in the 4-th row of Tab. 6, with the BRBs, the CCPNet shows worse results, decreasing 8.1% and 8.4% for SC and SSC, respectively. However, when introducing the guidance (the 5-th row), the model shows significant improvements for both SC and SSC tasks. Only with the feature amplification (the 6-th row), we observe a considerable improvement compared to the BRBs. A possible reason is that it is not enough for the detail recovery when only amplifying on the 3D context information. However, with the whole GRB, our approach shows best results.

Full-Resolution Prediction. To evaluate the benefits of

full-resolutions, we also re-implement our approach with the quarter and half resolution. To achieve this goal, we remove the corresponding layers after the deconvolution operations in Fig. 1. The last two rows of Tab. 6 show the performances. From the results, we can see that the low-resolution-based model shows worse performances. The main reason is that it cannot preserve the geometric details. However, our model still performs better than most state-of-the-art methods. This further demonstrates the effectiveness of our proposed modules. With full-resolution outputs, our model can fully exploit the geometric details, improving the IoUs by 4.9% and 5.1% respectively.

5. Conclusion

In this work, we propose a novel deep learning framework, named CCPNet, for full-resolution 3D SSC. The CCPNet is a self-cascaded pyramid structure to successively aggregate multi-scale 3D contexts and local geometry details. Extensive experiments on both synthetic and real benchmarks demonstrate that our CCPNet significantly improves the semantic completion accuracy, reduces the computational cost, and offers high-quality completion results with full-resolution. In the future work, we will explore color information for semantic and boundary enhancement.

References

[1] Amir Atapour-Abarghouei and Toby P Breckon. Depthcomp: real-time depth image completion based on prior semantic scene segmentation. In *BMVC*, pages 1–13, 2017.

[2] Liang-Chieh Chen, George Papandreou, Iasonas Kokkinos, Kevin Murphy, and Alan L Yuille. Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. *TPAMI*, 40(4):834–848, 2018.

[3] Özgün Çiçek, Ahmed Abdulkadir, Soeren S Lienkamp, Thomas Brox, and Olaf Ronneberger. 3d u-net: learning dense volumetric segmentation from sparse annotation. In *MICCAI*, pages 424–432, 2016.

[4] Michael Firman, Oisin Mac Aodha, Simon Julier, and Gabriel J Brostow. Structured prediction of unobserved voxels from a single depth image. In *CVPR*, pages 5431–5440, 2016.

[5] Martin Garbade, Yueh-Tung Chen, Johann Sawatzky, and Juergen Gall. Two stream 3d semantic scene completion. In *CVPR Workshops*, pages 1–10, 2019.

[6] Andreas Geiger and Chaohui Wang. Joint 3d object and layout inference from a single rgb-d image. In *GCPR*, pages 183–195, 2015.

[7] Klaus Greff, Rupesh K Srivastava, and Jürgen Schmidhuber. Highway and residual networks learn unrolled iterative estimation. *arXiv:1612.07771*, 2016.

[8] Andre Bernardes Soares Guedes, Teofilo Emidio de Campos, and Adrian Hilton. Semantic scene completion combining colour and depth: preliminary experiments. In *ICCV Workshop*, pages –, 2017.

[9] Ruiqi Guo, Chuhang Zou, and Derek Hoiem. Predicting complete 3d models of indoor scenes. *arXiv:1504.02437*, 2015.

[10] Yu-Xiao Guo and Xin Tong. View-volume network for semantic scene completion from a single depth image. In *IJCAI*, pages 726–732, 2018.

[11] Saurabh Gupta, Pablo Arbeláez, Ross Girshick, and Jitendra Malik. Indoor scene understanding with rgb-d images: Bottom-up segmentation, object detection and semantic segmentation. *IJCV*, 112(2):133–149, 2015.

[12] Saurabh Gupta, Pablo Arbelaez, and Jitendra Malik. Perceptual organization and recognition of indoor scenes from rgb-d images. In *CVPR*, pages 564–571, 2013.

[13] Saurabh Gupta, Ross Girshick, Pablo Arbeláez, and Jitendra Malik. Learning rich features from rgb-d images for object detection and segmentation. In *ECCV*, pages 345–360, 2014.

[14] Bharath Hariharan, Pablo Arbeláez, Ross Girshick, and Jitendra Malik. Hypercolumns for object segmentation and fine-grained localization. In *CVPR*, pages 447–456, 2015.

[15] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Delving deep into rectifiers: surpassing human-level performance on imagenet classification. arxiv e-prints 2015. In *ICCV*, pages 1026–1034, 2015.

[16] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, pages 770–778, 2016.

[17] Anastasia Ioannidou, Elisavet Chatzilari, Spiros Nikolopoulos, and Ioannis Kompatsiaris. Deep learning advances in computer vision with 3d data: A survey. *ACM Computing Surveys (CSUR)*, 50(2):20, 2017.

[18] Shuiwang Ji, Wei Xu, Ming Yang, and Kai Yu. 3d convolutional neural networks for human action recognition. *TPAMI*, 35(1):221–231, 2013.

[19] Yangqing Jia, Evan Shelhamer, Jeff Donahue, Sergey Karayev, Jonathan Long, Ross Girshick, Sergio Guadarrama, and Trevor Darrell. Caffe: Convolutional architecture for fast feature embedding. In *ACM Multimedia*, pages 675–678, 2014.

[20] Jie Li, Yu Liu, Dong Gong, Qinfeng Shi, Xia Yuan, Chunxia Zhao, and Ian Reid. Rgbd based dimensional decomposition residual network for 3d semantic scene completion. In *CVPR*, pages 7693–7702, 2019.

[21] Dahua Lin, Sanja Fidler, and Raquel Urtasun. Holistic scene understanding for 3d object detection with rgbd cameras. In *ICCV*, pages 1417–1424, 2013.

[22] Guosheng Lin, Anton Milan, Chunhua Shen, and Ian Reid. Refinenet: Multi-path refinement networks for high-resolution semantic segmentation. In *CVPR*, pages 1925–1934, 2017.

[23] Shice Liu, Yu Hu, Yiming Zeng, Qiankun Tang, Beibei Jin, Yinhe Han, and Xiaowei Li. See and think: Disentangling semantic scene completion. In *NIPS*, pages 261–272, 2018.

[24] Yongcheng Liu, Bin Fan, Lingfeng Wang, Jun Bai, Shiming Xiang, and Chunhong Pan. Semantic labeling in very high resolution images via a self-cascaded convolutional neural network. *ISPRS JPRS*, 145:78–95, 2018.

[25] Pedro O Pinheiro, Tsung-Yi Lin, Ronan Collobert, and Piotr Dollár. Learning to refine object segments. In *ECCV*, pages 75–91, 2016.

[26] Xiaofeng Ren, Liefeng Bo, and Dieter Fox. Rgb-(d) scene labeling: Features and algorithms. In *CVPR*, pages 2759–2766, 2012.

[27] Zhile Ren and Erik B Sudderth. Three-dimensional object detection and layout prediction using clouds of oriented gradients. In *CVPR*, pages 1525–1533, 2016.

[28] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *MICCAI*, pages 234–241, 2015.

[29] Nathan Silberman, Derek Hoiem, Pushmeet Kohli, and Rob Fergus. Indoor segmentation and support inference from rgb-d images. In *ECCV*, pages 746–760, 2012.

[30] Shuran Song and Jianxiong Xiao. Sliding shapes for 3d object detection in depth images. In *ECCV*, pages 634–651, 2014.

[31] Shuran Song and Jianxiong Xiao. Deep sliding shapes for amodal 3d object detection in rgb-d images. In *CVPR*, pages 808–816, 2016.

[32] Shuran Song, Fisher Yu, Andy Zeng, Angel X Chang, Manolis Savva, and Thomas Funkhouser. Semantic scene completion from a single depth image. In *CVPR*, pages 1746–1754, 2017.

[33] Weiyue Wang and Ulrich Neumann. Depth-aware cnn for rgb-d segmentation. In *ECCV*, pages 135–150, 2018.

[34] Fisher Yu and Vladlen Koltun. Multi-scale context aggregation by dilated convolutions. *ICLR*, 2016.

[35] Jiahui Zhang, Hao Zhao, Anbang Yao, Yurong Chen, Li Zhang, and Hongen Liao. Efficient semantic scene completion network with spatial group convolution. In *ECCV*, pages 733–749, 2018.

[36] Liang Zhang, Le Wang, Xiangdong Zhang, Peiyi Shen, Mohammed Bennamoun, Guangming Zhu, Syed Afaq Ali Shah, and Juan Song. Semantic scene completion with dense crf from a single depth image. *Neurocomputing*, 318:182–195, 2018.

[37] Pingping Zhang, Wei Liu, Hongyu Wang, Yinjie Lei, and Huchuan Lu. Deep gated attention networks for large-scale street-level scene segmentation. *PR*, 88:702–714, 2019.

[38] Pingping Zhang, Dong Wang, Huchuan Lu, Hongyu Wang, and Xiang Ruan. Amulet: Aggregating multi-level convolutional features for salient object detection. In *ICCV*, pages 202–211, 2017.

[39] Pingping Zhang, Luyao Wang, Dong Wang, Huchuan Lu, and Chunhua Shen. Agile amulet: Real-time salient object detection with contextual attention. *arXiv:1802.06960*, 2018.

[40] Hengshuang Zhao, Jianping Shi, Xiaojuan Qi, Xiaogang Wang, and Jiaya Jia. Pyramid scene parsing network. In *CVPR*, pages 2881–2890, 2017.

[41] Bo Zheng, Yibiao Zhao, Joey C Yu, Katsushi Ikeuchi, and Song-Chun Zhu. Beyond point clouds: Scene understanding by reasoning geometry and physics. In *CVPR*, pages 3127–3134, 2013.