Anisotropic Convolutional Networks for 3D Semantic Scene Completion *

Jie Li[1] Kai Han[2] Peng Wang[3†] Yu Liu[4] Xia Yuan[1]
[1]Nanjing University of Science and Technology, China [2]University of Oxford, United Kingdom
[3]University of Wollongong, Australia [4]The University of Adelaide, Australia

Abstract

As a voxel-wise labeling task, semantic scene completion (SSC) tries to simultaneously infer the occupancy and semantic labels for a scene from a single depth and/or RGB image. The key challenge for SSC is how to effectively take advantage of the 3D context to model various objects or stuffs with severe variations in shapes, layouts and visibility. To handle such variations, we propose a novel module called anisotropic convolution, which properties with flexibility and power impossible for the competing methods such as standard 3D convolution and some of its variations. In contrast to the standard 3D convolution that is limited to a fixed 3D receptive field, our module is capable of modeling the dimensional anisotropy voxel-wisely. The basic idea is to enable anisotropic 3D receptive field by decomposing a 3D convolution into three consecutive 1D convolutions, and the kernel size for each such 1D convolution is adaptively determined on the fly. By stacking multiple such anisotropic convolution modules, the voxel-wise modeling capability can be further enhanced while maintaining a controllable amount of model parameters. Extensive experiments on two SSC benchmarks, NYU-Depth-v2 and NYUCAD, show the superior performance of the proposed method. Our code is available at https://waterljwant.github.io/SSC/.

1. Introduction

To behave in the 3D physical world, it requires an accurate understanding of both the 3D geometry as well as the semantics of the environment. Humans can easily infer such geometrical and semantic information of a scene from partial observations. An open topic in computer vision is to study how to enable machines such an ability, which is desirable in many applications such as navigation [4], grasping [20], 3D home design [1], to name a few.

Semantic scene completion (SSC) [16] is a computer vision task teaching the machine how to perceive the 3D world from the static depth and/or RGB image. The task has two coupled objectives: one is 3D scene completion, which aims at inferring the volumetric occupancy of the scene, and the other is 3D scene labeling, which requires to predict the semantic labels voxel-wisely. As the objects within the physical scene carry severe variations in shapes, layouts, and visibility due to occlusions, the main challenge thereon is how to model the 3D context to learn each voxel effectively.

Recently, promising progress has been achieved for SSC [16, 6, 8, 10, 13] by employing deep convolutional neural networks (CNNs). A direct solution is to use 3D convolutional neural network [16] to model the volumetric context, which consists of a stack of conventional 3D convolutional layers. This solution, however, suffers from apparent limitations. On the one hand, 3D convolution renders a fixed receptive field that does not cater to the variations of the objects. On the other hand, 3D convolution is resource demanding, which causes massive computational and memory consumption. 3D convolution variations [10, 21] are proposed to address such shortcomings. For example, a lightweight dimensional decomposition network is proposed in [10] to alleviate the resource consumption, but it still leaves the object variation issue unattended.

In this work, we propose a novel module, termed anisotropic convolution, to model object variation, which properties with flexibility and power impossible for competing methods. In contrast to standard 3D convolution and some of its variations that are limited to the fixed receptive field, the new module adapts to the dimensional anisotropy property voxel-wisely and enables receptive field with varying sizes, a.k.a anisotropic receptive field. The basic idea is to decompose a 3D convolution operation into three consecutive 1D convolutions and equip each such 1d convolution with a mixer of different kernel sizes. The combination weights of such kernels along each 1D convolution are learned voxel-wisely and thus anisotropic 3D context can essentially be modeled by consecutively performing such adaptive 1D convolutions. Although we use multiple ker-

*This work is supported by the National Natural Science Foundation of China under Grants 61773210 and 61603184 and the EPSRC Programme Grant Seebibyte EP/M013774/1.
† Corresponding author.

nels, *e.g*. 3, due to the dimensional decomposition scheme, our module is still parameter-economic comparing to the 3D counterpart. By stacking multiple such modules, a more flexible 3D context, as well as an effective mapping function from such context to the voxel output, can be obtained.

The contributions of this work are as follows:

- We present a novel anisotropic convolutional network (AIC-Net) for the task of semantic scene completion. It renders flexibility in modeling the object variations in a 3D scene by automatically choosing proper receptive fields for different voxels.
- We propose a novel module, termed anisotropic convolution (AIC) module, which adapts to the dimensional anisotropy property voxel-wisely and thus implicitly enables 3D kernels with varying sizes.
- The new module is much less computational demanding with higher parameter efficiency comparing to the standard 3D convolution units. It can be used as a plug-and-play module to replace the standard 3D convolution unit.

We thoroughly evaluate our model on two SSC benchmarks. Our method outperforms existing methods by a large margin, establishing the new state-of-the-art. Code will be made available.

2. Related Work

2.1. Semantic Scene Completion

SSCNet [16] proposed by Song *et al*. is the first work that tries to simultaneously predict the semantic labels and volumetric occupancy of a scene in an end-to-end network. The expensive cost of 3D CNN, however, limits the depth of the network, which hinders the accuracy achieved by SSCNet. Zhang *et al*. [21] introduced spatial group convolution (SGC) into SSC for accelerating the computation of 3D dense prediction task. However, its accuracy is slightly lower than that of SSCNet. By combining the 2D CNN and 3D CNN, Guo and Tong [8] proposed the view-volume network (VVNet) to efficiently reduce the computation cost and enhance the network depth. Li *et al*. [11] use both depth and voxels as the inputs of a hybrid network and consider the importance of elements at different positions [23] while training.

Garbade *et al*. [6] proposed a two-stream approach that jointly leverages the depth and visual information. In specific, it first constructs an incomplete 3D semantic tensor for the inferred 2D semantic information, and then adopts a vanilla 3D CNN to infer the complete 3D semantic tensor. Liu *et al*. [13] also used RGB-D image as input and proposed a two-stage framework to sequentially carry out the 2D semantic segmentation and 3D semantic scene completion, which are connected via a 2D-3D re-projection layer. However, their two-stage method can suffer from the error accumulation, producing inferior results. Although significant improvements have been achieved, these methods are limited by the cost of 3D convolution and the fixed receptive fields. Li *et al*. [10] introduced a dimensional decomposition residual network (DDRNet) for the 3D SSC task. Although it achieves good accuracy with less parameters, it still leaves the limitation of using fixed receptive field unattended.

2.2. Going Beyond Fixed Receptive Field

Most existing models utilize fixed-size kernel to model fixed visual context, which are less robust and flexible when dealing with objects with various sizes.

Inception family [17, 19, 18] take receptive field with multiple sizes into account, and it implements this concept by launching multi-branch CNNs with different convolution kernels. The similar idea appears in atrous spatial pyramid pooling (ASPP) [2], multi-scale information was captured via using several parallel convolutions with different atrous(dilation) rates on the top of feature map. These strategies essentially embrace the idea of multi-scale fusion, and the same fusion strategy is uniformly applied to all the positions. Zhang *et al*. [22] choose a more suitable receptive field by weighting convolutions with different kernel sizes.

STN [9] designs a Spatial Transformer module to achieve invariance in terms of translation, rotation, and scale. However, it treats the whole image as a unit, rather than adjusts the receptive field pixel-wisely. Deformable CNN (DCNv1) [3] attempts to adaptively adjust the spatial distribution of receptive fields according to the scale and shape of the object. Specifically, it utilizes offset to control the spatial sampling. DCNv2 [25] increases the modeling power by stacking more deformable convolutional layers to improve its modelling ability and proposes to use a teacher network to guide the training process. However, DCNv2 still struggles to control the offset in order to focus on relevant pixels only.

Different from the above methods, the proposed AIC module is tailored for 3D tasks, in particular for SSC in this paper. It is capable of handling objects with variations in shapes, layouts and visibility by learning anisotropic receptive field voxel-wisely. At the same time, it achieves trade-off between semantic completion accuracy and computational cost.

3. Anisotropic Convolutional Networks

In this section, we introduce our ***ani****sotropic* ***c****onvolutional* ***net****works* (AIC-Net) for 3D semantic scene completion. At the core of AIC-Net is our proposed anisotropic convolutional (AIC) module. Given a single-view RGB-D image of a 3D scene, AIC-Net predicts a dense 3D voxel representation and maps each voxel in the

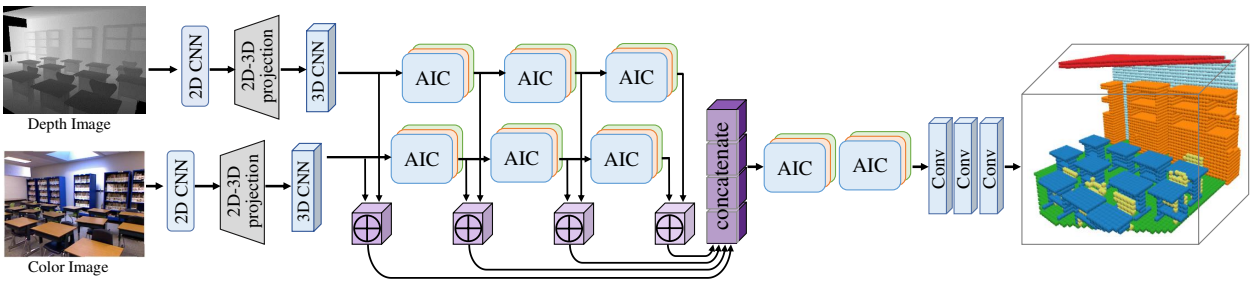


Figure 1. The overall network structure of AIC-Net. AIC-Net has two feature extractors in parallel to capture the features from RGB and depth images, respectively. The feature extractor contains a projection layer to map the 2D feature to 3D space. After that, we use stacked AICs to obtain information with adaptive receptive fields. The multi-scale features are concatenated and then fused through another two AICs followed by three voxel-wise convolutions to predict occupancy and object labels simultaneously.

view frustum to one of the labels $C = \{c_1, c_2, \cdots, c_{N+1}\}$, where N is the number of object classes, c_{N+1} represents the empty voxels, $\{c_1, c_2, \cdots, c_N\}$ represent the voxels occupied by objects of different categories.

Fig. 1 illustrates the overall architecture of our AIC-Net. It consists of a *hybrid feature extractor* for feature extraction from the depth map and RGB image, a *multi-stage feature aggregation module* with a stack of AIC modules to aggregate features obtained by the hybrid feature extractor, two extra AIC modules to fuse multi-stage information, followed by a sequence of voxel-wise 3D convolution layers to reconstruct the 3D semantic scene. The *hybrid feature extractor* contains two parallel branches to extract features for the depth map and the RGB image, respectively. Each branch contains a hybrid structure of 2D and 3D CNNs. The 2D and 3D CNNs are bridged by a 2D-3D projection layer, allowing the model to convert the 2D feature maps into 3D feature maps that are suitable for 3D semantic scene completion. The structure of our hybrid feature extractor follows that of DDRNet [10]. The *multi-stage feature aggregation module* consists of a sequence of AIC modules, each of which can voxel-wisely adjust the 3D context on the fly. The outputs of these AIC modules are concatenated together, and another two AIC modules fuse such multi-stage information. The 3D semantic scene can then be reconstructed by applying a sequence of voxel-wise 3D convolutional layers on the fused feature.

In the rest of this section, we will introduce our AIC module (section 3.1), the multi-path kernel selection mechanism achieved by stacking our AIC modules (section 3.2), and the training loss for our model (section 3.3) in detail.

3.1. Anisotropic Convolution

Considering the variations in object shapes, layouts as well as the varying levels of occlusion in SSC, it will be beneficial to model different context information to infer the occupancy and semantics for different voxel positions. The anisotropic convolution (AIC) module is proposed to adapt to such variations, allowing the convolution to accommodate 3D geometric deformation. Fig. 2 shows the structure of our AIC module. Instead of using the 3D kernels ($k_1 \times k_2 \times k_3$) that are limited to the fixed 3D receptive field, we model the dimensional anisotropy property by enabling the kernel size for each 3D dimension to be learnable. To achieve this, we first decompose the 3D convolution operation as the combination of three 1D convolution operations along each dimension x, y, z. In each dimension, we can inject multiple (*e.g.* 3 in our implementation) kernels of different sizes to enable more flexible context modeling. For example, for dimension x, we can have three kernels as $(1 \times 1 \times k_1^x)$, $(1 \times 1 \times k_2^x)$, and $(1 \times 1 \times k_3^x)$. A set of selection weights, a.k.a. modulation factors, will be learned to select proper kernels along each of the three dimensions.

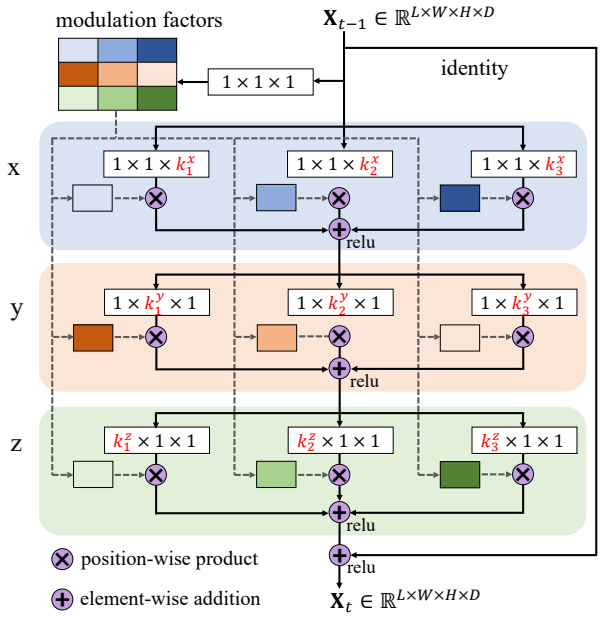


Figure 2. Anisotropic convolution. For each dimension, we set 3 parallel convolution with different kernel sizes as an example. The learned modulation factors for different kernels are denoted with different colors. The values of the modulation factors are positive and the values of each row sum up to 1.

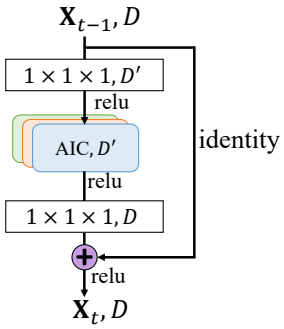


Figure 3. Bottleneck version AIC module. The first convolution reduces the number of channels from D to D' ($D' < D$) and the last convolution increases the channels back to D.

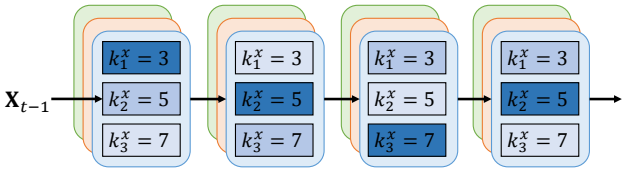


Figure 4. Illustration of multi-path kernel selection in one dimension. In this example, four AIC modules are stacked and for each module the kernel sizes for each dimension are $\{3, 5, 7\}$. The background darkness of the kernel indicates the value of the modulation factor, and thus reflects the selection tendency for this kernel. Stacking multiple AIC modules can increase the range of receptive fields exponentially.

Note that the kernel candidates for different dimensions are not necessary to be the same. When there are n, m, and l candidate kernels along x, y, and z dimensions respectively, the possible kernel combinations can grow exponentially as, $\{k_1^z, k_2^z, \cdots, k_l^z\} \times \{k_1^y, k_2^y, \cdots, k_m^y\} \times \{k_1^x, k_2^x, \cdots, k_n^x\}$. The AIC module can learn to select different kernels for each dimension, forming an anisotropic convolution to capture anisotropic 3D information.

Modulation factors To enable the model to determine the optimal combination of the candidate kernels and consequently adaptively controlling the context to model different voxels, we introduce a modulation module in the AIC module. As shown in Fig. 2, assume the input to an AIC module is a tensor $\mathbf{X}_{t-1} \in \mathbb{R}^{L \times W \times H \times D}$, where L, W, H denotes the length, width, height of the tensor, and D indicates the dimensionality of the feature. The output $\mathbf{X}_t \in \mathbb{R}^{L \times W \times H \times D}$ can be formulated as,

$$\mathbf{X}_t = \mathcal{F}^z(\mathcal{F}^y(\mathcal{F}^x(\mathbf{X}_{t-1}))) + \mathbf{X}_{t-1}, \tag{1}$$

where $\mathcal{F}^u$ represents the anisotropic convolution along the $u \in \{x, y, z\}$ dimension. We adopt a residual structure to obtain the output by element-wisely summing up the input tensor and the output of three consecutive anisotropic 1D convolutions. Without losing generality, we represent $\mathcal{F}^x(\mathbf{X}_{t-1})$ as,

$$\mathbf{X}_t^x = \sum_{i=1}^{n} f^x(\mathbf{X}_{t-1}, \theta_i^x) \odot g^x(\mathbf{X}_{t-1}, \phi^x)[i], \tag{2}$$

where $f^x(\mathbf{X}_{t-1}, \theta_i^x)$ represents performing convolution to $\mathbf{X}_{t-1}$ using parameter θ_i^x which has kernel size $(1, 1, k_i^x)$ with $k_i^x \in \{k_1^x, k_2^x, \cdots, k_n^x\}$, n is the toal numner of candidate kernels for dimension x, and $\odot$ denotes element-wise multiplication. $g^x(\mathbf{X}_{t-1}, \phi^x)$ is a mapping function from the input tensor to the weights or modulation factors used to select the kernels along dimension x and ϕ^x denotes the parameters of the mapping function. We perform $softmax$ to $g^u(\cdot, \cdot)[i]$ in order that the weights for the kernels of each dimension $u \in \{x, y, z\}$ sum up to 1, that is,

$$\sum_{i=1}^{p \in \{n,m,l\}} g^u(\cdot, \phi^u)[i] = 1, \quad g^u(\cdot, \phi^u)[i] \geq 0. \tag{3}$$

In this sense, we adopt a soft constraint with a set of weights to determine the importance of different kernels. The two extreme cases are that the learned modulation factor is 1 or 0, indicating that the corresponding kernel will be the unique selected or be ignored. By using soft values, we can control the contributions of these kernels more flexibly.

In Fig. 2, we show an example of the AIC module with $m = n = l = 3$ and as seen, $g^u(\cdot, \cdot)$ is realized by a 1-layer 3D convolution with kernel $(1 \times 1 \times 1)$.

Bottleneck anisotropic convolution To further reduce the parameters of our AIC module, we propose a bottleneck based AIC module. As shown in Fig. 3, for each AIC module, a $(1 \times 1 \times 1)$ convolution is added both before and after the AIC operation. These two convolutions are responsible for reducing and restoring the feature channels, allowing the AIC module to have a more compact input. In the remainder of the paper, unless stated otherwise, AIC refers to the bottleneck based AIC.

3.2. Multi-path Kernel Selection

Despite the attractive properties in a single AIC module, here we show that greater flexibility can be achieved by stacking multiple AIC modules. Stacking multiple AIC modules forms multiple possible paths between layers implicitly and consequently enables an extensive range of receptive field variations in the model. Fig. 4 shows a stack of four AIC modules, and each module sets the kernel sizes to $\{3, 5, 7\}$ along all three dimensions. For one specific dimension, when each module tends to select the kernel size 7, a maximum receptive field of 25 will be obtained for this dimension. On the contrary, a minimum receptive field of 9 can be obtained for a dimension, if kernel size 3 dominates the selections of all four AIC modules in this dimension. In theory, the receptive field for this particular dimension

	scene completion			semantic scene completion											
Methods	prec.	recall	IoU	ceil.	floor	wall	win.	chair	bed	sofa	table	tvs	furn.	objs.	avg.
Lin *et al.* [12]	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 *et al.* [7]	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 [16]	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
EsscNet [21]	**71.9**	71.9	56.2	17.5	75.4	25.8	6.7	15.3	53.8	42.4	11.2	0	33.4	11.8	26.7
DDRNet [10]	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
VVNet [8]	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
AIC-Net	62.4	91.8	59.2	**23.2**	90.8	32.3	**14.8**	18.2	51.1	44.8	15.2	**22.4**	**38.3**	**15.7**	**33.3**

Table 1. Results on the NYU [15] dataset. Bold numbers represent the best scores.

	scene completion			semantic scene completion											
Methods	prec.	recall	IoU	ceil.	floor	wall	win.	chair	bed	sofa	table	tvs	furn.	objs.	avg.
Zheng *et al.* [24]	60.1	46.7	34.6	-	-	-	-	-	-	-	-	-	-	-	-
Firman *et al.* [5]	66.5	69.7	50.8	-	-	-	-	-	-	-	-	-	-	-	-
SSCNet [16]	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
TS3D [6]	80.2	91.0	74.2	33.8	**92.9**	46.8	**27.0**	27.9	**61.6**	51.6	27.6	**26.9**	44.5	22.0	42.1
DDRNet [10]	**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
VVNet [8]	86.4	92.0	80.3	-	-	-	-	-	-	-	-	-	-	-	-
AIC-Net	88.2	90.3	**80.5**	53.0	91.2	**57.2**	20.2	**44.6**	58.4	56.2	**36.2**	9.7	**47.1**	**30.4**	**45.8**

Table 2. Results on the NYUCAD dataset [24]. Bold numbers represent the best scores.

can freely vary in the range of $(9, 25)$. When considering three dimensions simultaneously, the number of 3D receptive fields supported by our AIC network will grow exponentially, which will provide flexibility and power for modeling object variations impossible for competing methods.

3.3. Training Loss

Our proposed AIC-Net can be trained in an end-to-end fashion. We adopt the voxel-wise cross-entropy loss function [16] for the network training. The loss function can be expressed as,

$$\mathcal{L} = \sum_{i,j,k} w_{ijk} \mathcal{L}_{sm}\left(p_{ijk}, y_{ijk}\right), \tag{4}$$

where $\mathcal{L}_{sm}$ is the cross-entropy loss, y_{ijk} is the ground truth label for coordinates (i, j, k), p_{ijk} is the predicted probability for the same voxel, and w_{ijk} is the weight to balance the semantic categories. We follow [16, 10] and use the same weights in our experiments.

4. Experiments

In this section, we start by introducing some key implementation details, followed by the description of the datasets as well as the evaluation metrics. Then we present some quantitative comparisons between the propose AIC-Net and some other existing works. Furthermore, qualitative comparisons are given through visualization. Finally, comprehensive ablation studies are performed to inspect some critical aspects of AIC-Net.

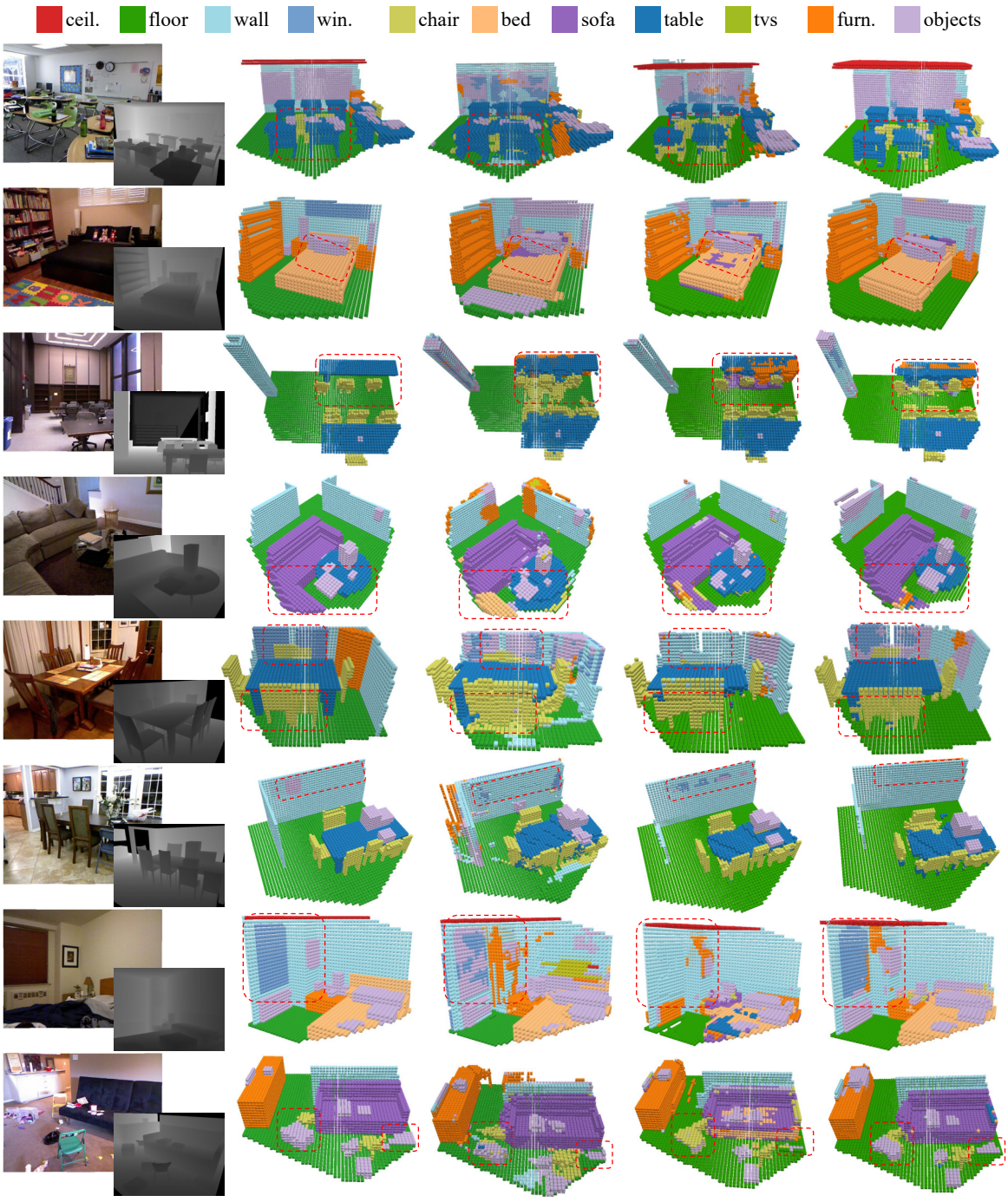
4.1. Implementation Details

In our AIC-Net, we stack three AIC modules for each branch in the multi-stage feature aggregation part, and two AIC modules are adopted to fuse these features. All the AIC modules used are the bottleneck version as shown in Fig. 3. For the three AIC modules in feature aggregation, the bottleneck layer is used to decrease the dimensionality of the features from $D = 64$ to $D' = 32$. For the AIC modules in feature fusion part, the dimensionalities of features before and after the bottleneck layer are $D = 256$ and $D' = 64$. Unless stated otherwise, we use three candidate kernels with kernel size $\{3, 5, 7\}$ for each dimension of all AIC modules. More details about the network structure can be found in the supplements.

Our model is trained by using SGD with a momentum of 0.9 and a weight decay of 10^{-4}. The initial learning rate is set to be 0.01, which decays by a factor of 10 every 15 epochs. The batch size is 4. We implement our model using PyTorch. All the experiments are conducted on a PC with 4 NVIDIA RTX2080TI GPUs.

Datasets. We evaluate the proposed AIC-Net on two SSC datasets. One dataset is the NYU-Depth-V2 [15], which is also known as the NYU dataset. The NYU dataset consists of 1,449 depth scenes captured by a Kinect sensor. Following SSCNet [16], we use the 3D annotations provided by [14] for semantic scene completion task. The second dataset is the NYUCAD dataset [5]. This dataset uses the depth maps generated from the projections of the 3D annotations to reduce the misalignment of depths and the annotations and thus can provide higher-quality depth maps.

Evaluation metrics. For semantic scene completion, we measure the intersection over union (IoU) between the predicted voxel labels and ground-truth labels for all object classes. Overall performance is also given by computing the average IoU over all classes. For scene completion, all voxels are to be categorized into either empty or occupied.



(a) RGB and Depth images (b) Ground truth (c) SSCNet (d) DDRNet (e) Ours

Figure 5. Qualitative results on NYUCAD. From left to right are input RGB-D image, the ground truth, results generated by SSCNet [16], DDRNet [10] and the proposed AIC-Net. (Best viewed in color.)

A voxel is counted as occupied if it belongs to any of the semantic classes. For scene completion, apart from IoU, precision and recall are also reported. Note that the IoU for semantic scene completion is commonly accepted as a more important metric in the SSC task.

4.2. Comparison with the State-of-the-Art

We compare our AIC-Net with the state-of-the-art methods on NYU and NYUCAD. The results are reported in Table 1 and Table 2, respectively. In Table 1, we can see that for the semantic scene completion our method signifi-

Methods	scene completion			semantic scene completion											
	prec.	recall	IoU	ceil.	floor	wall	win.	chair	bed	sofa	table	tvs	furn.	objs.	avg.
AIC-Net, k={3, 5, 7}	88.2	**90.3**	**80.5**	**53.0**	91.2	**57.2**	20.2	**44.6**	58.4	56.2	**36.2**	**9.7**	**47.1**	30.4	**45.8**
AIC-Net, k={5, 7}	**88.3**	89.5	79.9	51.0	91.3	56.8	18.6	41.3	**58.6**	**59.4**	34.6	4.8	46.7	**30.9**	44.9
AIC-Net, k={7}	86.3	**90.3**	79.1	50.7	**91.7**	54.5	**21.2**	38.0	55.5	57.1	33.2	7.9	44.9	29.4	44.0
AIC-Net, k={5}	87.8	88.2	78.4	49.6	91.3	55.3	15.7	38.7	**58.6**	52.8	30.9	0.	43.9	30.2	42.5

Table 3. The performance of AIC-Net under different kernel sets. We use the same kernel set $k = (k_1, k_2, \cdots, k_n)$ for each dimension. Results are reported on NYUCAD [24] dataset.

Methods	scene completion			semantic scene completion											
	prec.	recall	IoU	ceil.	floor	wall	win.	chair	bed	sofa	table	tvs	furn.	objs.	avg.
NYU															
AIC-Net-noMFs	**71.4**	79.0	**59.9**	22.3	**90.8**	32.0	14.4	14.5	47.5	41.3	12.6	16.8	32.8	12.7	30.7
AIC-Net	62.4	**91.8**	59.2	**23.2**	**90.8**	**32.3**	**14.8**	**18.2**	**51.1**	**44.8**	**15.2**	**22.4**	**38.3**	**15.7**	**33.3**
NYUCAD															
AIC-Net-noMF	87.2	**90.3**	79.6	51.1	**91.7**	57.0	18.5	39.3	51.4	51.8	30.7	1.3	45.0	30.1	42.5
AIC-Net	**88.2**	**90.3**	**80.5**	**53.0**	91.2	**57.2**	**20.2**	**44.6**	**58.4**	**56.2**	**36.2**	**9.7**	**47.1**	**30.4**	**45.8**

Table 4. The importance of the modulation factors. AIC-Net-noMFs denotes we set all the modulation factors to be 1. Results are reported on the NYU [15] and NYUCAD [24] datasets.

cantly outperforms other methods in overall accuracy. The proposed AIC-Net achieves 2.9% better than the cutting-edge approach DDRNet [10] in terms of the average IoU. For scene completion, our method is slightly outperformed by DDRNet [10]. The scene completion task requires to predict the volumetric occupancy, which is class-agnostic. Since our AIC-Net aims at modeling the object variation voxel-wisely, its advantage will fade in the binary completion task. In Table 2, our AIC-Net achieves the best semantic segmentation performance as well, and our average IoU outperforms the second-best approach by 3%. For scene completion, our method also observes superior performance, although the advantage is not as significant. Among the comparing methods, SSCNet [16] is built using standard 3D convolution. The inferior performance lies twofold. First, the fixed receptive field is not ideal for addressing object variations. Second, 3D convolution is resource demanding, which can limit the depth of the 3D network and consequently sacrifices the modeling capability.

Another interesting observation from these two tables is that our AIC-Net tends to obtain better performance on some categories that have more severe shape variations, *e.g.* *chair*, *table*, *objects*.

4.3. Qualitative Results

In Fig. 5, we show some visualization results to evaluate the effectiveness of our AIC-Net qualitatively. Generally, we can see that the proposed AIC-Net can handle diverse objects with various shapes and thus give more accurate semantic predictions and shape completion than SSCNet [16] and DDRNet [10]. Some challenging examples include "chairs" and "tables" in Row 1, Row 3, and Row 5, which require a model to adaptively adjust the receptive field voxel-wisely. For example, for some more delicate parts like "legs", a smaller receptive field can be more beneficial. It shows that our AIC-Net can identify such objects more clearly. While for some other objects like "windows" in Row 5 and Row 7, it expects to see the larger context. Both SSCNet and DDRNet fail in this case, but our method still successfully identifies them from other surrounding distractors. The "bed" in Row 2, the "wall" in Row 6, and the "sofa" in Row 4 also demonstrate the superiority of our approach. In Row 8, the "objects" marked by the red dashed rectangle are in a messy environment. Our AIC-Net is less vulnerable to the influence of surrounding objects and more accurately distinguishes the categories and shapes of these "objects".

4.4. Ablation Study

In this section, we dive into the AIC-Net to investigate its key aspects in detail. Specifically, we try to answer the following questions. 1). Is it beneficial to use multiple candidate kernels along each dimension of the AIC module? 2). Is the performance improvement simply coming from multiple kernels? 3). Will that work if the AIC module is used as a plug-and-play module? 4). The trade-off between SSC performance and cost.

The effectiveness of using multiple kernels In our AIC module, we use multiple candidate kernels in each dimension x, y, z, and use the learned modulation factors to choose proper kernels along each of these dimensions. Since we expect our AIC-Net to be able to deal with objects of varying shapes, the kernels in AIC should be sufficiently distinct. In our experiments, we set the kernel set to be $\{3, 5, 7\}$ across all three dimensions. The first question needs to be clarified is that will it be enough to use only the maximum kernel, *i.e.* 7 in our network? Then, are three kernels better than two? From the results of Table 3, we can see, either two kernels $\{5, 7\}$ or three kernels $\{3, 5, 7\}$ can

	scene completion			semantic scene completion											
method	prec.	recall	IoU	ceil.	floor	wall	win.	chair	bed	sofa	table	tvs	furn.	objs.	avg.
DDRNet-DDR-ASPP [10]	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
DDRNet-AIC-ASPP	87.9	89.1	79.4	48.0	90.9	56.1	20.1	41.6	56.6	55.0	33.1	12.6	45.3	29.0	44.4
DDRNet-DDR-AIC	88.0	89.6	79.7	49.0	91.4	57.6	19.7	40.5	52.3	52.9	32.5	6.1	44.6	30.7	43.4
DDRNet-AIC-AIC	87.5	89.3	79.1	51.7	91.5	56.4	16.5	44.1	56.3	56.4	35.4	12.3	46.1	30.4	45.2

Table 5. AIC module as plug-and-play modules. The components of DDRNet [10] are replaced by the AIC modules. Results are reported on NYUCAD [24] dataset.

Methods	Params/k	FLOPs/G	SC-IoU	SSC-IoU
SSCNet [16]	930.0	163.8	73.2	40.0
DDRNet [10]	195.0	27.2	79.4	42.8
3D conv, k=(3, 3, 3)	440.1	61.0	-	-
3D conv, k=(5, 5, 5)	1443.6	191.1	-	-
3D conv, k=(7, 7, 7)	3675.9	480.4	-	-
AIC-Net*, k={3, 5, 7}	628.7	85.5	79.1	45.2
AIC-Net, k={3, 5, 7}	847.0	113.7	80.5	45.8
AIC-Net, k={5, 7}	716.0	96.77	79.9	44.9

Table 6. Params, FLOPs and Performance of our approach compared with other methods. 3D conv, $k = (k_1, k_2, k_3)$ denotes we replace our AIC module with a 3D convolution unit with 3D kernel (k_1, k_2, k_3). AIC-Net* denotes a AIC-Net with one AIC module in feature fusion part, while by default we use two AIC modules.

outperform kernel 7. Since the maximum receptive field for all these three options is 7, the results demonstrate the benefits of using multiple kernels. At the same time, three kernels outperform two kernels by about 1% because it renders more flexibility in modeling the context.

Is it necessary to use modulation factors? In the above paragraph, we show the benefit of using multiple kernels along each dimension. However, another question arises that is the improvement simply coming from multiple kernels? In other words, is that necessary to learn modulation factors to adaptively select the kernels voxel-wisely? From Table 4, we can see when we discard the modulation factors in AIC modules, the performance of AIC-Net observes obvious degradation on both NYU and NYUCAD datasets. These results show that the superior performance of AIC-Net relies on modeling the dimensional anisotropy property by adaptively selecting proper kernels along each dimension. To further inspect the anisotropic nature of the learned kernels, we observed the statistical values of the modulation factors and found that: 1.) the selected kernel sizes are basically consistent with the object sizes; 2.) the modulation values for different voxels vary a lot within one scene; 3.) the modulation values among the three separable dimensions have significant variation. This indicates the learned "3D receptive field" are anisotropic and adaptive.

AIC module used as a plug-and-play module Due to its ability to model the anisotropic context, our AIC module is expected to be able to benefit other networks when it is used as a plug-and-play module. To validate this, we choose the DDRNet [10] as the test-bed, and use the AIC module to replace its building blocks, DDR and ASPP. DDR block models 3D convolution in a lightweight manner with the fixed receptive field. ASPP is a feature fusion scheme commonly used in semantic segmentation to take advantage of the multi-scale context. Table 5 shows the comparison. When we use AIC to replace the DDR module in DDRNet [10], the SSC-IoU is improved by 1.6%. When we replace ASPP by our AIC module, we still observe a 0.6% improvement in semantic segmentation. Finally, when we replace both DDR and ASPP by AIC, the result can be further boosted.

Trade-off in performance and cost Since we decompose the 3D convolution into three consecutive 1D convolutions, the model parameters and computation grow linearly with the number of candidate kernels in each dimension. While for standard 3D convolution, the parameters and computation will have cubic growth. Table 6 presents some comparisons in terms of both efficiency and accuracy. For the *3D conv,* $k = (k_1, k_2, k_3)$ in the table, it means we use this particular 3D convolution to replace our AIC module. As can be seen, when the 3D kernel size is $(5, 5, 5)$, it will result in 3 times of parameters and FLOPs comparing to our AIC-Net. When the kernel size is increased to $(7, 7, 7)$, the parameter and computation scale will be 8 times more than ours. DDRNet is a lightweight structure, which consumes the least parameters and has the lowest computation complexity, but it observes a glaring performance gap comparing to our method. Thus, our AIC-Net achieves a better trade-off between performance and cost.

5. Conclusion

In this paper, we proposed a novel AIC-Net, to handle the object variations in the semantic scene completion (SSC) task. At the core of AIC-Net is our proposed AIC module, which can learn anisotropic convolutions by adaptively choosing the convolution kernels along all three dimensions voxel-wisely. By stacking multiple such AIC modules, it allows us more flexibly to control the receptive field for each voxel. This AIC module can be freely inserted into existing networks as a plug-and-play module to effectively model the 3D context in a parameter-economic manner. Thorough experiments were conducted on two SSC datasets, and the AIC-Net outperforms existing methods by a large margin, establishing the new state-of-the-art.

References

[1] Planner5d. https://planner5d.com/. 1

[2] Liang-Chieh Chen, George Papandreou, Florian Schroff, and Hartwig Adam. Rethinking atrous convolution for semantic image segmentation. *arXiv preprint arXiv:1706.05587*, 2017. 2

[3] Jifeng Dai, Haozhi Qi, Yuwen Xiong, Yi Li, Guodong Zhang, Han Hu, and Yichen Wei. Deformable convolutional networks. In *Proceedings of the IEEE international conference on computer vision*, pages 764–773, 2017. 2

[4] Anh-Dzung Doan, Yasir Latif, Tat-Jun Chin, Yu Liu, Thanh-Toan Do, and Ian Reid. Scalable place recognition under appearance change for autonomous driving. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 9319–9328, 2019. 1

[5] 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

[6] Martin Garbade, Johann Sawatzky, Alexander Richard, and Juergen Gall. Two stream 3d semantic scene completion. *arXiv:1804.03550*, 2018. 1, 2, 5

[7] Andreas Geiger and Chaohui Wang. Joint 3d object and layout inference from a single rgb-d image. In *GCPR*, pages 183–195, 2015. 5

[8] Yuxiao Guo and Xin Tong. View-volume network for semantic scene completion from a single depth image. In *Proc. IJCAI*, pages 726–732, 7 2018. 1, 2, 5

[9] Max Jaderberg, Karen Simonyan, Andrew Zisserman, et al. Spatial transformer networks. In *Advances in neural information processing systems*, pages 2017–2025, 2015. 2

[10] 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. 1, 2, 3, 5, 6, 7, 8

[11] Jie Li, Yu Liu, Xia Yuan, Chunxia Zhao, Roland Siegwart, Ian Reid, and Cesar Cadena. Depth based semantic scene completion with position importance aware loss. *IEEE Robotics and Automation Letters*, 5(1):219–226, 2019. 2

[12] Dahua Lin, Sanja Fidler, and Raquel Urtasun. Holistic scene understanding for 3d object detection with rgbd cameras. In *ICCV*, pages 1417–1424, 2013. 5

[13] Shice Liu, Yu Hu, Yiming Zeng, Qiankun Tang, Beibei Jin, Yinhe Han, and Xiaowei Li. See and think: Disentangling semantic scene completion. In *Advances in Neural Information Processing Systems*, pages 263–274, 2018. 1, 2

[14] Jason Rock, Tanmay Gupta, Justin Thorsen, JunYoung Gwak, Daeyun Shin, and Derek Hoiem. Completing 3d object shape from one depth image. In *CVPR*, pages 2484–2493. IEEE, 2015. 5

[15] Nathan Silberman, Derek Hoiem, Pushmeet Kohli, and Rob Fergus. Indoor segmentation and support inference from rgbd images. In *ECCV*, pages 746–760. Springer, 2012. 5, 7

[16] 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 190–198, 2017. 1, 2, 5, 6, 7, 8

[17] Christian Szegedy, Sergey Ioffe, Vincent Vanhoucke, and Alexander A Alemi. Inception-v4, inception-resnet and the impact of residual connections on learning. In *Thirty-First AAAI Conference on Artificial Intelligence*, 2017. 2

[18] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. Going deeper with convolutions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1–9, 2015. 2

[19] Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jon Shlens, and Zbigniew Wojna. Rethinking the inception architecture for computer vision. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2818–2826, 2016. 2

[20] Jacob Varley, Chad DeChant, Adam Richardson, Joaquín Ruales, and Peter Allen. Shape completion enabled robotic grasping. In *Int. Conf. IROS*, pages 2442–2447, 2017. 1

[21] Jiahui Zhang, Hao Zhao, Anbang YaoE, Yurong Chen, Li Zhang, and Hongen LiaoE. Efficient semantic scene completion network with spatial group convolution. In *ECCV*, pages 733–749, 2018. 1, 2, 5

[22] Lei Zhang, Zhiqiang Lang, Peng Wang, Wei Wei, Shengcai Liao, Ling Shao, and Yanning Zhang. Pixel-wise deep function-mixture network for spectral super-resolution. In *AAAI Conference on Artificial Intelligence*, 2020. 2

[23] Lei Zhang, Peng Wang, Chunhua Shen, Lingqiao Liu, Wei Wei, Yanning Zhang, and Anton van den Hengel. Adaptive importance learning for improving lightweight image super-resolution network. *International Journal of Computer Vision*, pages 1 – 21, 2019. 2

[24] 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. 5, 7, 8

[25] Xizhou Zhu, Han Hu, Stephen Lin, and Jifeng Dai. Deformable convnets v2: More deformable, better results. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 9308–9316, 2019. 2