# Improved Sliding Shapes for Instance Segmentation of Amodal 3D Object

**Jinhua Lin[1], Yu Yao[1] and Yanjie Wang[2]**
[1] Computer Application Technology, Changchun University of Technology
Changchun, Jilin 130000 - China
[e-mail: ljh3832@163.com, 927168801@qq.com]
[2] Machinery & Electronics Engineering, Chinese Academy of Sciences University
Changchun, Jilin 130033 - China
[e-mail: 282765569@qq.com]
*Corresponding author: Jinhua Lin

*Received November 21, 2017; revised February 22, 2018; accepted April 5, 2018; published November 30, 2018*

## Abstract

State-of-art instance segmentation networks are successful at generating 2D segmentation mask for region proposals with highest classification score, yet 3D object segmentation task is limited to geocentric embedding or detector of Sliding Shapes. To this end, we propose an amodal 3D instance segmentation network called A3IS-CNN, which extends the detector of Deep Sliding Shapes to amodal 3D instance segmentation by adding a new branch of 3D ConvNet called A3IS-branch. The A3IS-branch which takes 3D amodal ROI as input and 3D semantic instances as output is a fully convolution network(FCN) sharing convolutional layers with existing 3d RPN which takes 3D scene as input and 3D amodal proposals as output. For two branches share computation with each other, our 3D instance segmentation network adds only a small overhead of 0.25 fps to Deep Sliding Shapes, trading off accurate detection and point-to-point segmentation of instances. Experiments show that our 3D instance segmentation network achieves at least 10% to 50% improvement over the state-of-art network in running time, and outperforms the state-of-art 3D detectors by at least 16.1 AP.

***Keywords:*** instance segmentation, detector, fully convolution network, amodal proposals

This research was supported by National Natural Science Foundation of China (Grant No. 51705032), National High-tech R&D Program (Grant No. 2014AA7031010B) and Science-Technology Project of the thirteenth Five-Year Plan (Grant No. 2016345).

**http://doi.org/10.3837/tiis.2018.11.021** **ISSN : 1976-7277**

## 1. Introduction

Amodal 3D instance segmentation plays an important role in robotic vision system, it combines the work of object detection and semantic segmentation in 3D world. The state-of-the-art 3D object detection networks(e.g, class label and bounding box regression for 3D object) focus on geocentric embedding for depth images[1] or sliding shape for 3D object[2][3]. The geocentric embedding method encodes additional channels for RGB-D images and achieves semantic and instance segmentation for 2.5D scene. This method extracts depth and RGB features by R-CNN[4] which generates region proposals through Selective Search[5], yet the R-CNN is a computationally expensive network compared to the latest incarnation, Faster R-CNN[6]. The embedding method spends much time for proposals generation while its segmentation is only for 2.5d instances. Compared with Sliding Shape method which detects the 3D bounding box by extra CAD data, the Deep Sliding Shapes method[3] detects the bbox of amodal 3D object by multi-scale RPN(Region Proposal Network), the latter extracts geometric and RGB features from 3D and 2D convolution networks, this strategy speeds up the generation of amodal 3D proposals, but it is only for 3D detection task without instance segmentation.

On the other hand, the state-of-art 3D semantic segmentations focus on improving deep architectures by different 3D geometric input. The typical input is represented by 3D voxels[7], collection of views[8] or 3D point cloud[9]. [7][8][9] provide effective CNN architectures for 3D semantic segmentation(e.g, classify pixels/points without differentiation of instances), yet 3D instance segmentation denotes detection via masks which is both semantic and detection.

To this end, we introduce an amodal 3D instance segmentation network, called A3IS-CNN, extends Deep Sliding network by adding a new 3D FCN branch for predicting segmentation masks on each 3D amodal RoI(Region of Interest), being compatible with the existing branch for localization and regression of 3D bounding box(see **Fig. 1**). The 3D FCN branch shares a common set of convolution layers with 3D RPN, detecting semantic instances in a point-to-point manner. A3IS-CNN is supervised to learn 3D and color features from NYUv2 dataset[10] and ModelNet40[11] dataset, while the network is tested by the random amodal perception which is sampled from real scene by RGB-D sensor.

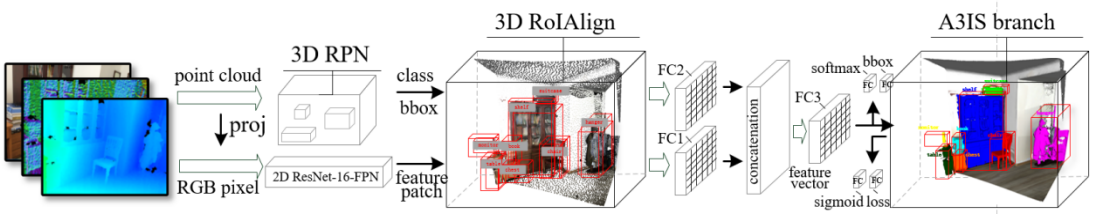


**Fig. 1.** Amodal 3D instance segmentation network. The network takes the 3D point cloud from deep maps as input, the 3D RPN extracts 3D amodal proposals at multi-scales with respect to different receptive fields of RGB-D sensor. For each 3D proposal, the 2D feature patches are feed back to the ResNet-16 to jointly learn the class labels and the bounding box regressions.

A3IS-CNN is a general extension of Deep Sliding Shapes, yet building the 3D FCN branch correctly is important for good result. Deep Sliding Shapes was not proposed for point-to-point segmentation between classes. It divides the space into coarse voxel grid for each proposal box. Inspired by Mask R-CNN[12] which fixes the misalignment through a 2D

RoIAlign layer, we propose a corresponding 3D layer called 3D RoIAlign, which exactly preserves the spatial locations of 3D proposals. 3D RoIAlign improves segmentation accuracy by 16.2AP compared with the state of art. 3D RoIAlign inherits the advantages of Mask R-CNN, it predicts the 3D mask of instances for each class independently, this independency guarantees robust segmentation for 3D amodal without artifacts on overlapping instances.

During testing, A3IS-CNN takes 19.8s per frame, in which, 5.87s is for new branch of instance segmentation. Training on NYUv2 dataset takes 5 to 6 days on a single NVIDIA 1070i 8-GPU. We believe that the fast train and test speeds will be achieved by higher configured GPU machine.

## 2. Related Work

**2D Object Detection and Segmentation** The Region-based Convolutional Network(R-CNN) extracts features from regional candidates to accelerate the speed of 2D object detection[4]. However, the overlap between candidates reduces the running time of R-CNN, the Fast R-CNN[13] was proposed to deal this problem. Fast R-CNN was build upon R-CNN and SPPnet[14] by maxpooling the proposal into a fixed size output. Then Faster R-CNN[15] improved the RoI pooling line by a full convolutional network called Region Proposal Network(RPN). Faster R-CNN is most efficient architecture for 2D object detection in the state-of-art. Driven by these popular networks, the methods for 2D classifications are proposed upon the fine tuning steps. Mask R-CNN is proposed to share the computation between detection network and segmentation network in 2D images, leading to the most improvement of accuracy and speed. In summary, the 2D object detection and segmentation methods[16,17,18,19] are prosperous for the baseline of Spatial pyramid pooling, RPN and parallel computation.

**3D Object Detection and Segmentation** RGB-D images are widely used for 3D object detection and segmentation[20,21,22,23,24]. The geometric embedding method[1] enriches the raw depth channel by three additional features including height above ground, angle with gravity and horizontal disparity. This enriched representation allows for robust localization of bounding box and accurate classification of pixels in proposals. However, this method obtaining regional candidates in 2.5 dimension is not compatible with 3D object segmentation[25,26,27], while the complexity of network is increased by enriched features of RGB-D images. Our network uses RGB-D images as input, transferring the depth data into 3D point cloud[28,29,30], this strategy leads to better accuracy for 3D object segmentation. However, the point cloud is usually computed in high cost time, to this end, we introduce a 3D RoIAlign layer(inspired by 2D RoIAlign in Mask R-CNN) to take a 3D key point cloud as input and 3D amodal proposals as output. This simple change improves the accuracy of existing 3D RPN while speeding up the detection time.

The core goal of our network is to achieve point-wise amodal 3D instance segmentation[31,32,33] without increasing the running time of existing 3D detection streamline(e.g, Deep Sliding Shapes network[3]). To this end, we extend the Deep Sliding Shapes by two steps. The first step, a new branch of CNN is added to the streamline of Deep Sliding Shapes to learn point-wise features of 3D amodal RoI(proposals), this new branch shares the same convolution layers with existing 3D RPN, this step allows for parallel computation between two branches without additional time for instance segmentation. The second step, a 3D RoIAlign layer is build to jointly learn the 3D geometric features with the 2D color features, and corrects the misalignment between 3D amodal RoI and the extracted features. This step guarantees accurate segmentation of 3D amodal instances.

## 3. Amodal 3D Instance Segmentation Network(A3IS-CNN)

A3IS-CNN is theoretically concise. Deep Sliding Shapes takes two outputs for each 3D amodal proposals, an objectness score and a 3D object bounding box. In our A3IS-CNN, a new branch is established to output the 3D semantic instances. A3IS-CNN is a reasonable and natural extension to Deep Sliding Shapes. However, the output of additional branch is different from the outputs of classification and bbox regression, needing much meticulous feature extraction of an 3D amodal RoI. In following sections, we will introduce the core parts of A3IS-CNN, including point-wise alignment, which is the core extending piece of Deep Sliding Shapes.

### 3.1 A3IS-CNN

At first, we briefly introduce the 3D detector of Deep Sliding Shapes. Deep Sliding Shapes consists of two sub-networks. The first network is a 3D Region Proposal Network(3D RPN) which learns objectness from 3D shapes. The second network extracts geometric and RGB features from each 3D amodal proposal and color images and jointly learns object classification and 3D bounding box regression. The two sub-networks effectively perform amodal 3D object detection with RGB-D images as input.

A3IS-CNN uses the same two sub-network and adds 3D instance segmentation sub-network(called A3IS branch) to Deep Sliding Shapes. In A3IS branch, in addition to generating classification label and 3D bbox loss, A3IS-CNN also outputs a binary 3D instance for each 3D amodal RoI. This architecture is different from most state-of-art network, where 3D classification result is build upon instance segmentation. Our network inherits the streamline of Deep Sliding Shapes that proposes 3D bounding box localization and fine-tuning in parallel.

In training of A3IS-CNN, we define the loss function for each 3D amodal proposals as follows:

$$L = L_{cls} + L_{reg} + L_{ins} \tag{1}$$

where the objectness score $L_{cls}$ and bounding box regression loss $L_{reg}$ are the same with the multi-task loss defined in Deep Sliding Shapes. The A3IS branch defines $L_{ins}$ as an average cross-entropy loss, which uses the sigmoid function as a neuron activation function. The sigmoid function is defined as:

$$L_{ins} = \sum_{i} [k \ln act + (1-k)\ln(1-act)] \tag{2}$$

where $k$ is the num of ground-truth class, $act$ is the actual value of binary instance. For each 3D amodal proposal which is related to ground-truth class $k$, $L_{ins}$ is only defined on the $k$ th class, other outputs of 3D instances is not contributed to $L_{ins}$. The integral loss function is expanded as follows:

$$L(p_i, p_i^*, t_i, t_i^*) = \sum_{i} L_{cls}(p_i, p_i^*) + \lambda \sum_{i} p_i^* L_{reg}(t_i, t_i^*) + \sum_{i} [k \ln act + (1-k)\ln(1-act)] \tag{3}$$

where $i$ is the index number of a 3D proposal, $p_i$ is the predicted probability of proposal $i$ being an 3D amodal object. $p_i^*$ is binary ground-truth label, '1' represents positive proposal, '0' represents negative proposal. $t_i$ is a vector representing the two diagonal coordinates of predicted bounding box, $t_i^*$ is the ground-truth bbox corresponding to a positive proposal. Our definition of multi-task loss is inspired by the Mask loss defined in [12]. $L_{ins}$ inherits the advantages of Mask loss, which allows for segmentation of instances for every class without

competition among classes. This formulation guarantees point-wise segmentation of instances without overlapping along boundaries.

A semantic instance corresponds to an input 3D amodal RoI. It is different from classification scores or bounding box loss that are generally transformed into vectors by 'FC' layers, our A3IS-CNN extracts the 3D geometric shapes and semantic features in point-by-point manner by convolution layers. A3IS branch predicts semantic instances from each 3D amodal RoI by a full convolution network(FCN)[34], the pipeline of A3IS branch is shown in **Fig. 2**. FCN allows for the 3D amodal object coming through the branch without reduction of dimension. Our point-to-point strategy guarantees our 3D semantic output to be accurately aligned with the per-point input of 3D amodal object. This alignment is achieved by a sub-network called 3D RoIAlign which will be shown in next section.

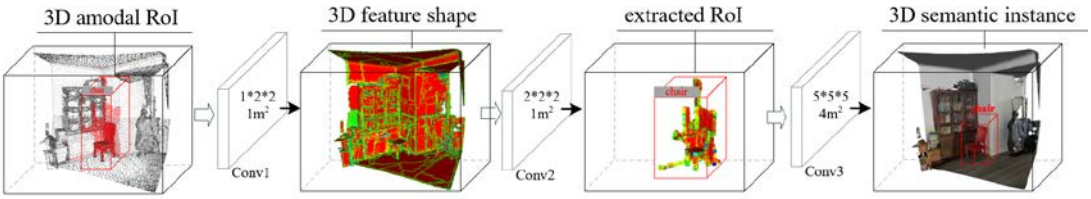


**Fig. 2.** Architecture of A3IS branch. The 3D amodal RoI is fixed by a multi-scale sliding box, the amodal object within the sliding box is to be segmented in semantic manner. Combined with the feature extraction step, the 3D amodal object is extracted as a colored TSDF.

### 3.2 3D RoIAlign

ReLU+Pool layer is a common operation for extracting features from input proposals. Deep Sliding Shapes uses this ReLU+Pool layer to extract 3D geometric features from 3D object, yet the color feature can not be aggregated automatically. To this end, Deep Sliding Shapes introduces 2D ConvNets(VGGnet) to extract color features from input images, then 3D points inside the bbox are projected to their corresponding 2D image patch. This projection method improves resolution of 3D detection compared with directly encoding color on 3D voxels[35,36,37], yet this projection introduces misalignment between the 3D proposals and the extracted color features. This misalignment may not impact the localization of bounding box, which is the core goal of detection network(Deep Sliding Shapes). Yet for our instance segmentation network, this misalignment has a large negative effect on predicting point-wise instances.

To this end, we propose a 3D RoIAlign layer, which eliminates the misalignment between 3D proposal and 2D color features, accurately aligning the color patch with the 3D amodal proposals. 3D RoIAlign layer is composed of two steps, the first step is quantization of proposals, the quantized proposals are then segmented into small 3D bins; the second step is max pooling of feature values which include geometric and color feature, the features of each bin are fused into an integral 3D amodal proposal with bounding box regression and classification labels. The architecture of 3D RoIAlign is shown in **Fig. 3**.

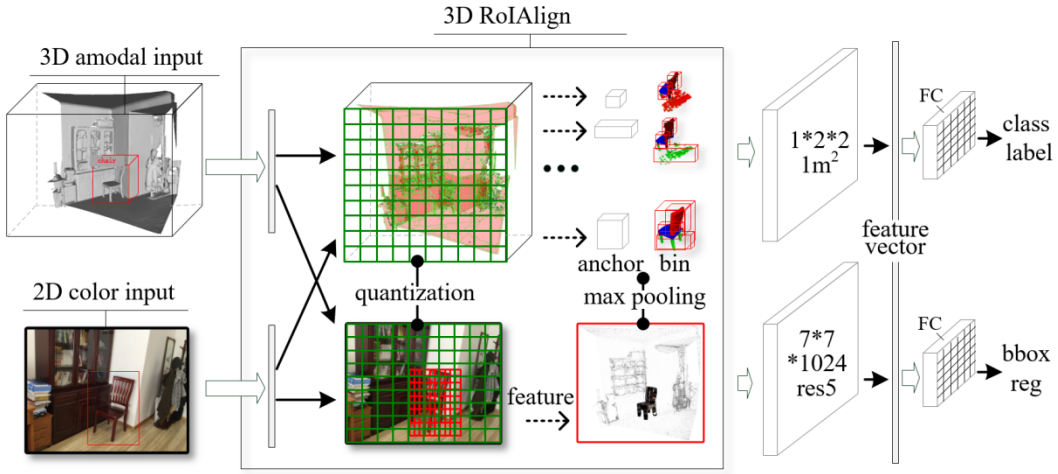


**Fig. 3.** The detection streamline of 3D RoIAlign. For each 3D amodal input, we yield the anchors and bins for each sliding boxes, and feed the 2D color patches(projection of the 3D proposals) to a ResNet-16. The class labels and bbox regressions are jointly learned from two combined branches which are presented as 3D RoIAlign.

In **Fig. 3**, the core of 3D RoIAlign is quantization step with respect to 3D proposals, we use Gaussian quantizer[38,39] to approximate each neuron's activation function. In order to discrete the feature values of the 3D amodal RoI, we define the quantization function as follows:

$$\begin{cases} Q(x) = q_i, & x \in (t_i, t_{i+1}] \\ q_{i+1} - q_i = \Delta, & \forall i \end{cases} \tag{4}$$

where $x$ is coordinate of 3D bin, $i$ is the index of anchor, $t_i$ is the $i$-th step of anchor, $q_i$ is quantization level, $\Delta$ is a constant quantization step. The quantization level of $q_i$ is used as the activation value of $x$, and $q_i \in \mathbf{R}^{c,l,w,h}$, $c$ is the number of filter channel, $l, w, h$ is the length, width, height of the 3D amodal RoI. The Gaussian quantizer is optimally defined in the mean error range as follows:

$$Q^*(x) = \arg\min_Q E_x[(Q(x) - x)^2] = \arg\min_Q \int p(x)(Q(x) - x)^2 dx \tag{5}$$

where $p(x)$ is probability density function. $Q^*(x)$ is a non-uniform function, by substituting the constraint of the formula (8) into the formula (9), an uniform solution of $Q^*(x)$ will be obtained.

Since quantizer is distributed to each neuron efficiently, and this quantizer changes with the back-propagation pass, this leads to the accurate computation resutls for each 3D proposal.

### 3.3 Implementation details

Following the Deep Sliding Shapes, the parameters of A3IS-CNN are designed in the best way. Our A3IS-CNN keeps robust to these parameters being designed for 3D amodal detection and segmentation.

In our work, a 3D amodal RoI is regarded as positive, if its 3D IoU[40] is no less than 0.35, whereas the negative IoU is set to be less than 0.15. Our instance loss $L_{ins}$ is computed only for positive proposals. The task of our 3D instance segmentation is to close the gap between a 3D

amodal proposal and its related ground-truth value.

As in training of Deep Sliding Shapes, we compute the difference of centers $[c_x, c_y, c_z]$ and sizes $[s_1, s_2, s_3]$ between a positive anchor and its associated ground-truth box. Each Sliding window has $N = 19$ anchor boxes spanning 4 scales and 3 aspect ratios. The physical sizes of 3D amodal scene is ranged from -2.6 meters to 2.6 meters in length, -1.5 meters to 1 meters in height and 0.4 meters to 5.6 meters in width. The size of anchor box varies from 0.3 meters to 2 meters. We train on 2 GPUs with 4 effective batch for 40k iterations. Each GPU processes two amodal scene, each amodal scene has 256 anchors with positive and negative ratio of 1:1. Our A3IS branch shares the same architecture with 3D RPN, so they are trained jointly unless specified.

As for testing, after removing the empty anchors, the number of proposal is 112,764 for A3IS-CNN. Our 3D amodal detection and segmentation work are applied to these 3D proposals. We pick the top scoring 1000 boxes to input to the 3D instance segmentation branch(e.g A3IS branch). This is different from the training stage which shares the computation between the same convolution architecture(e.g A3IS branch and 3D RPN), yet we improve the accuracy of segmentation and extend a 3D detector(Deep Sliding Shapes) to a 3D instance segmentation network(A3IS-CNN).

## 4. Experimental Results

In this section, our A3IS-CNN is compared with the state of art, and ablation experiments is done to evaluate the performance of our network. We use the NYUv2 and PointNet dataset to train 3D RPN and A3IS-branch on NVIDIA i1070 GPU(8G), which takes about 5 to 7 days. As for testing, A3IS-branch spends 5.87fps to finish 3D instance segmentation which occupies 29.5% of the total running time, it is much faster than PointNet segmentation which takes 30 mins in CPU and 5 mins in GPU. The overall accuracy of A3IS-CNN is 52.5AP on average which is 16.1 more than Deep Sliding Shapes.

### 4.1 Overall Comparison

In **Table 1**, we compare A3IS-CNN to the state-of-the-art networks in 3D instance segmentation. Our A3IS-CNN outperforms other state-of-the-art methods, they are PointNet, geocentric embedding method(called for short: GeocNet) and Deep Sliding Shapes(called for short: DSS). PointNet designs a new CNN called PointNet to achieve 3D detection and segmentation on point cloud. GeocNet uses geocentric embedding strategy to enrich the depth images learning more features to improve precision of detection and segmentation. DSS uses deep sliding windows to achieve amodal 3D detection and segmentation. We use AP(averaged precision with a threshold of 0.35/0.15) to evaluate segmentation performance. The superscripts of AP represent object detection and segmentation in different scales. A3IS-CNN outperforms PointNet by 28.26 in AP and only 16.4 in $AP^r$, this indicates that using point cloud as input improves segmentation precision in larger invisible regions. A3IS-CNN uses point cloud as single input without color information reduces segmentation precision by 13.7 AP, this indicates that color input is important for our 3D RoIAlign layer which allows for accurate extraction of features from 3D amodal proposals. Our A3IS branch guarantees accurate segmentation of instances in 52.5 AP and 56.2 $AP^r$, this little higher points indicates the effectiveness of our branch for predicting invisible part of 3D amodal object.

**Table 1.** Segmentation performance of several state-of-art networks.

| | input-CNN | AP | $AP^2$ | $AP^4$ | $AP^b$ | $AP^r$ | $AP^m$ |
|---|---|---|---|---|---|---|---|
| PointNet | Point cloud+PointNet | 24.24 | 28.5 | 25.5 | - | 39.8 | - |
| GeocNet | RGB-D+RCNN | 32.1 | 36.4 | 33.4 | 37.3 | 36.5 | 35.5 |
| DSS | RGB-D+Faster RCNN | 36.4 | 40.2 | 38.5 | 37.8 | 40.5 | 38.8 |
| A3IS-CNN | Point cloud+3D RPN | 26.8 | 30.5 | 27.8 | 28.5 | 29.5 | 29.8 |
| A3IS-CNN | RGB-D+3D RPN | 40.5 | 45.2 | 41.2 | 38.2 | 44.8 | 40.4 |
| A3IS-CNN | RGB-D+RPN+A3IS branch | **52.5** | 58.1 | 53.1 | 52.1 | **56.2** | 53.5 |

The segmentation results of A3IS-CNN are shown in **Fig. 4**. A3IS-CNN achieves robust segmentation of 3D instances even in overlapping regions. The comparison results are visualized in **Fig. 5**, DSS shows an overlapping error, this indicates that it is difficult for DSS to predict instances in point-wise level. A3IS-CNN segments 3D amodal instances in much more accurate manner.

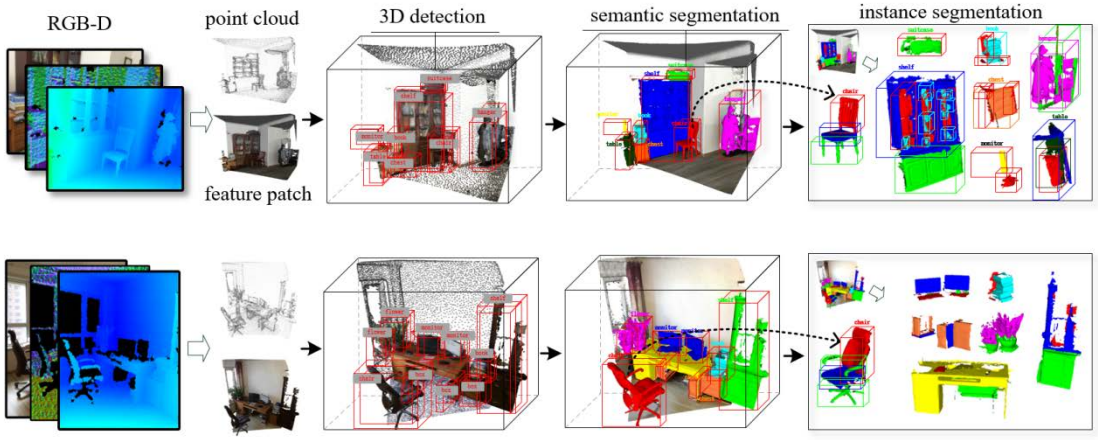


**Fig. 4.** More segmentation results of A3IS-CNN on 3D amodal scenes with 52.5 AP, running at 5.87 fps. For the detection results, we show the sliding boxes for the 3D distributions of the amodal proposals(red boxes with semantic labels). For the segmentation results, our A3IS-CNN can recognize and segment the amodal object in point-wise manner(the semantic instances are presented in different colors).

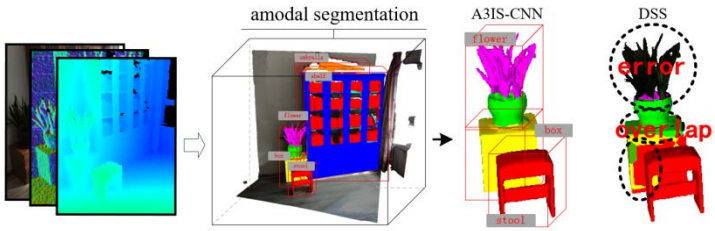


**Fig. 5.** Comparison of A3IS-CNN and DSS for segmentation of 3D amodal instantiations. A3IS-CNN detects the semantic instances without drift phenomenon, even for overlapping area. Yet the DSS is fail to detect complex amodal object, especially for the semantic instances within the same sliding box.

## 4.2 Ablation evaluation

In this section, ablation experiments are done to evaluate the performance of A3IS-CNN. The evaluation results are exhibited in **Table 2**. We will analyze the results as follows.

**2D ConvNet for color feature extraction**. Color feature is an indispensable factor for accurate segmentation of 3D amodal scenes. In our A3IS-CNN, we use pre-trained 2D ConvNet to extract color features which are concatenated with 3D RPN to learn geometric and color information effectively. Point cloud data inside of the 3D amodal proposals are projected to the corresponding 2D bounding box and jointly aggregated into one feature patch. **Table 2(a)** presents A3IS-CNN with a set of different 2D detectors.

| architecture | AP | $AP^2$ | $AP^4$ |
|---|---|---|---|
| AlexNet | 35.2 | 41.4 | 34.8 |
| VGGNet | 39.9 | 45.5 | 39.8 |
| ResNet-8 | 43.2 | 49.2 | 44.1 |
| ResNet-16-C4 | 45.5 | 51.2 | 46.3 |
| ResNet-16-FPN | **50.5** | 56.4 | 51.2 |

**(a)**Comparison of various 2D ConvNet models with 50.5 box AP at best.

| | AP | $AP^2$ | $AP^4$ |
|---|---|---|---|
| RoIPool+sofmax | 38.5 | 40.6 | 37.8 |
| RoIPool+sigmoid | 39.0 | 42.9 | 38.2 |
| RoIAlign+softmax | 40.0 | 43.9 | 39.2 |
| RoIAlign+sigmoid | 44.5 | 50.8 | 46.2 |
| difference | **+4.5** | +6.9 | +7.0 |

**(b)**Comparison of A3IS-branch with sigmoid and softmax in different RoI layers.

**Table 2.** Evaluation results of A3IS-CNN for albation experiments.

**A3IS-branch for point-wise segmentation**. A3IS-CNN predicts locations of 3D bounding box and 3D instantiation respectively. A3IS-branch segments instances in associated classes with no need for corresponding to other classes. In **Table 2(b)**, we integrate a point-wise sigmoid and softmax into A3IS branch separately to evaluate the average precision(AP) of A3IS-CNN. Compared with softmax, A3IS-CNN with sigmoid shows a higher AP(at least 4.5 points) in multi-scale segmentation. In general, sigmoid corresponds to a binary loss which regrets the location of bbox point by point, yet softmax corresponds to a multinomial loss which regrets the location of bbox class by class. Point-wise sigmoid is more suitable for instance segmentation. To this end, we use sigmoid and binary loss to predict amodal instances. This means that if an instantiation is fit to a certain class, our instance segmentation is done in this class box without reference to other classes.

### 4.3 Error analysis for 3D RoIAlign

We evaluate the performance of 3D RoIAlign layer by drawing error curves for three FCNs under various quantizers including RoIPool, RoIWarp and our 3D RoIAlign, the evaluation results are shown in **Fig. 6**. This three FCNs are AlexNet, ResNet-16 and VGG-net. We integrate this three FCNs respectively into our A3IS-CNN as 2D detectors to extract color features for 3D RoIAlign. Compared with other two sampling methods, our 3D RoIAlign shows a finer performance in 4

0.5 AP which is 2.5 points higher than RoIPool(used in Deep Sliding Shapes). RoIWarp is better than RoIPool by about 1.0 higher AP, yet the misalignment is not eliminated substantively in this two 2D quantizers. Our proposed 3D RoIAlign layer inherits the advantages of 2D RoIAlign(proposed in Mask R-CNN) to effectively extract color features from input RGB-D images, on the other hand, we project the 3D geometric features within a

3D bounding box to the closed 2D bbox, then the alignment is done in this combined feature patch by our 3D RoIAlign layer.

Since our 3D quantizer directly samples features in each neuron point by point, and the error values of quantizer reduces obviously along with the deepening of back-propagation iteration(as can be seen in **Fig. 6**), this indicates that the computation of quantization is much accurate for each 3D proposal in our A3IS-CNN.

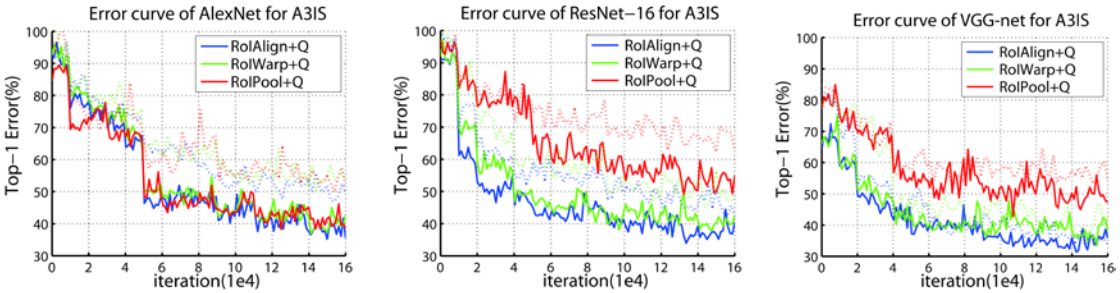


**Fig. 6.** Performance comparison of our 3D RoIAlign layer with Gaussian quantization. We respectively take three popular CNNs as 2D convnet branch in out network to evaluate the robustness of our A3IS-CNN. Dotted lines indicate the inference results, along with the deepening of iterations, the error curves converge to lower percentages, this indicate that A3IS-CNN performs convolutions in robust manner regardless the type of 2D convnet branches.

## 5. Conclusion

We propose a 3D instance segmentation network for point-wise detection of 3D amodal scenes. A 3D ConvNet branch is presented to segment 3D instantiations independently which is trained jointly with the existing 3D RPN. Our network achieves at least 50faster improvement over the state-of-art instance segmentation network, and outperforms the state-of-art 3D detectors by at least 16.1 AP.

## References

[1] S. Gupta, R. Girshick, P. Arbeláez and J. Malik, "Learning Rich Features from RGB-D Images for Object Detection and Segmentation," in *Proc. of the 13th European Conference on Computer Vision*, pp. 345-360, September 6-12, 2014. Article (CrossRef Link)

[2] S. Gupta, P. Arbeláez, R. Girshick and J. Malik, "Aligning 3d models to rgb-d images of cluttered scenes," in *Proc. of the 28th IEEE Conference on Computer Vision and Pattern Recognition*, pp. 4731-4740, June 7-12, 2015. Article (CrossRef Link)

[3] S. Song, J. Xiao, "Deep Sliding Shapes for Amodal 3D Object Detection in RGB-D Images," in *Proc. of the 29th IEEE Conference on Computer Vision and Pattern Recognition*, pp. 808-816, June 27-30, 2016. Article (CrossRef Link)

[4] Ross Girshick, Jeff Donahue, Trevor Darrell and Jitendra Malik, "Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation," in *Proc. of the 27th IEEE Conference on Computer Vision and Pattern Recognition*, pp. 580-587, June 23-28, 2014. Article (CrossRef Link)

[5] J. R. Uijlings, K. E. Sande, T. Gevers and A. W. Smeulders, "Selective Search for Object Recognition," *International Journal of Computer Vision*, vol. 104, no. 2, pp. 154-171, September, 2013.Article (CrossRef Link)

[6] S. Ren, K. He, R. Girshick and J. Sun, "Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 39, no. 6, pp:1137-1149, June, 2017. Article (CrossRef Link)
[7] D. Maturana and S. Scherer, "VoxNet: A 3D Convolutional Neural Network for real-time object recognition," in *Proc. of IEEE Conf. on Intelligent Robots and Systems*, pp.250-257, September 28-October 2, 2015. Article (CrossRef Link)
[8] H. Su, S. Maji, E. Kalogerakis and E. Learnedmiller, "Multi-view Convolutional Neural Networks for 3D Shape Recognition," in *Proc. of the 28th IEEE Conference on Computer Vision and Pattern Recognition*, pp. 945-953, December 7-13, 2015. Article (CrossRef Link)
[9] Charles Ruizhongtai Qi, Hao Su and Kaichun Mo, "PointNet: Deep Learning on Point Sets for 3D Classification and Segmentation," in *Proc. of the 29th IEEE Conference on Computer Vision and Pattern Recognition*, pp. 201-210, June 27-30, 2016. Article (CrossRef Link)
[10] N. Silberman, D. Hoiem, P. Kohli and R. Fergus, "Indoor Segmentation and Support Inference from RGBD Images," in *Proc. of the 11th European Conference on Computer Vision*, pp. 746-760, September 6-12, 2012. Article (CrossRef Link)
[11] Z. Wu, S. Song, A. Khosla and F. Yu. "3D ShapeNets: A deep representation for volumetric shapes," in *Proc. of the 27th IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1912-1920, June 23-28, 2014. Article (CrossRef Link)
[12] K. He, G. Gkioxari, P. Dollár and R. Girshick, "Mask R-CNN," in *Proc. of the 17th International Conference on Computer Vision*, pp. 746-760, October 22-29, 2017. Article (CrossRef Link)
[13] Girshick R, "Fast R-CNN," in *Proc. of the 15th International Conference on Computer Vision*, pp. 1440-1448, December 7-13, 2015. Article (CrossRef Link)
[14] K. He, X. Zhang, S. Ren and J. Sun, "Spatial Pyramid Pooling in Deep Convolutional Networks for Visual Recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 37, no. 9, pp:1904-1916, September, 2015. Article (CrossRef Link)
[15] S. Ren, R. Girshick, R. Girshick and J Sun, "Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 39, no. 6, pp:1137-1149, June, 2017. Article (CrossRef Link)
[16] Yuesheng Zhu, Yifeng Jiang, Zhuandi Huang and Guibo Luo, "SuperDepthTransfer: Depth Extraction from Image Using Instance-Based Learning with Superpixels," *KSII Transactions on Internet and Information Systems*, vol. 11, no. 10, pp. 4968-4986, 2017. Article (CrossRef Link)
[17] Yiyu Hong and Jongweon Kim, "Retrieval of Non-rigid 3D Models Based on Approximated Topological Structure and Local Volume," *KSII Transactions on Internet and Information Systems*, vol. 11, no. 8, pp. 3950-3964, 2017. Article (CrossRef Link)
[18] T. Xi, W. Zhao, H. Wang and W. Lin, "Salient object detection with spatiotemporal background priors for video," *IEEE Transactions on Image Processing*, vol. 26, no. 7, pp:3425-3436, July, 2017. Article (CrossRef Link)
[19] EshedOhn-Bar and Mohan ManubhaiTrivedi, "Multi-scale volumes for deep object detection and localization," *Pattern Recognition*, vol. 61, no. 1, pp:557-572, January, 2017. Article (CrossRef Link)
[20] Xiao Li, Ming Fang, JuJie Zhang and Jinqiao Wu, "Learning Coupled Classifiers with RGB images for RGB-D object recognition," *Pattern Recognition*, vol. 61, no. 1, pp:433-446, January, 2017. Article (CrossRef Link)
[21] S. Gupta, R. Girshick and J. Malik, "Indoor Scene Understanding with RGB-D Images: Bottom-up Segmentation, Object Detection and Semantic Segmentation," *International Journal of Computer Vision*, vol. 112, no. 2, pp:133-149, April, 2015. Article (CrossRef Link)
[22] U. Asif, M. Bennamoun and F.A. Sohel, "RGB-D Object Recognition and Grasp Detection Using Hierarchical Cascaded Forests," *IEEE Transactions on Robotics*, vol. 33, no. 3, pp:547-564, June, 2017. Article (CrossRef Link)\

[23] X. Xu, Y. Li, G. Wu and J. Luo, "Multi-modal Deep Feature Learning for RGB-D Object Detection," *Pattern Recognition*, vol. 72, no. 4, pp:300-313, December, 2017. Article (CrossRef Link)
[24] C.Y. Ren, V.A. Prisacariu, O. Kähler, ID Reid and DW Murray, "Real-Time Tracking of Single and Multiple Objects from Depth-Colour Imagery Using 3D Signed Distance Functions," *International Journal of Computer Vision*, vol.124, no. 1, pp:1-16, August, 2017. Article (CrossRef Link)
[25] Syed Afaq Ali Shah, Mohammed Bennamoun and Farid Boussaid, "Keypoints-based surface representation for 3D modeling and 3D object recognition," *Pattern Recognition*, vol. 64, no. 3, pp:29-38, April, 2017. Article (CrossRef Link)
[26] ZehuanYuan, Tong Lu and Chew LimTan, "Learning Discriminated and Correlated Patches for Multi-View Object Detection using Sparse Coding," *Pattern Recognition*, vol. 69, no. 4, pp:26-38, September, 2017. Article (CrossRef Link)
[27] PengShuai Wang, Yang Liu, YuXiao Guo and Xin Tong, "O-CNN: octree-based convolutional neural networks for 3D shape analysis," *ACM Transactions on Graphics*, vol. 36, no. 4, pp:1-11, July, 2017. Article (CrossRef Link)
[28] Radu Bogdan Rusu and Steve Cousins, "3D is here: Point Cloud Library (PCL)," *in Proc. of IEEE International Conference on Robotics and Automation*, pp. 1-4, May 9-13, 2011. Article (CrossRef Link)
[29] J. Digne and J.M. Morel, "Numerical analysis of differential operators on raw point clouds," *Numerische Mathematik*, vol. 127, no. 2, pp:255-289, June, 2014. Article (CrossRef Link)
[30] W. Cheng, W. Lin, X.Zhang, M. Goesele and M.T. Sun, "A Data-Driven Point Cloud Simplification Framework for City-Scale Image-Based Localization," *IEEE Transactions on Image Processing*, vol. 26, no. 1, pp:262-275, January, 2016. Article (CrossRef Link)
[31] Zhenyu Shu, Chengwu Qi, Ligang Liu, Shiqing Xin, Chao Hu, Li Wang and Yu Zhang, "Unsupervised 3D shape segmentation and co-segmentation via deep learning," *Computer Aided Geometric Design*, vol. 43, no. C, pp:39-52, March, 2016. Article (CrossRef Link)
[32] Kaan Yücer, Alexander Sorkine-Hornung, Oliver Wang and Olga Sorkine-Hornung, "Efficient 3D Object Segmentation from Densely Sampled Light Fields with Applications to 3D Reconstruction," *ACM Transactions on Graphics*, vol. 35, no. 3, pp:22, June, 2016. Article (CrossRef Link)
[33] Anurag Arnab and Philip H. S. Torr, "Pixelwise Instance Segmentation with a Dynamically Instantiated Network," in *Proc. of the 30th IEEE Conference on Computer Vision and Pattern Recognition*, pp. 879-888, July 21-26, 2017. Article (CrossRef Link)
[34] Jonathan Long, Evan Shelhamer and Trevor Darrell, "Fully convolutional networks for semantic segmentation," Computer Vision and Pattern Recognition. in *Proc. of the 28th IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3431-3440, June 7-12, 2015. Article (CrossRef Link)
[35] Daniel Maturana and Sebastian Scherer, "VoxNet: A 3D Convolutional Neural Network for real-time object recognition," in *Proc. of IEEE International Conference on Intelligent Robots and Systems*, pp:922-928, September 28- October 2, 2015. Article (CrossRef Link)
[36] G. Hackenberg, R. McCall and W. Broll, "Lightweight palm and finger tracking for real-time 3D gesture control," in *Proc. of the 11th IEEE Conf. on Virtual Reality*, pp:19-26, March 19-23, 2011. Article (CrossRef Link)
[37] Luís A. Alexandre, "3D Object Recognition Using Convolutional Neural Networks with Transfer Learning Between Input Channels," in *Proc. of the 13th International Conference on Advances in Intelligent Systems and Computing*, pp:889-898, January, 2016. Article (CrossRef Link)
[38] Z. Cai, X. He, J. Sun and N, "Vasconcelos. Deep Learning with Low Precision by Half-wave Gaussian Quantization," in *Proc. of the 30th IEEE Conference on Computer Vision and Pattern Recognition*, pp. 5406-5414, July 23-28, 2017. Article (CrossRef Link)

[39] S. Han, H. Mao and W.J. Dally, "Deep Compression: Compressing Deep Neural Networks with Pruning, Trained Quantization and Huffman Coding," *Fiber*, vol. 56, no. 4, pp:3-7, October, 2016. Article (CrossRef Link)
[40] I. Lenz, H. Lee and A. Saxena, "Deep Learning for Detecting Robotic Grasps," *International Journal of Robotics Research*, vol. 34, no. 4-5, pp:705-724, January, 2013. Article (CrossRef Link)

**Jinhua Lin** received her Ph.D. degree from Chinese Academy of Sciences University in July 2017. Her research interests include computer vision, digital image processing, mechatronic Engineering, AR/VR.

**Yu Yao** received her Ph.D degree from Jilin University in 2016. Her research interests include computer vision, Mechatronic Engineering.

**Yanjie Wang** received his M.S degree in computer science from Jilin University. He was a senior researcher in Changchun Institute of Optics, Fine Mehcanics and Physics, Chinese Academy of Sciences. His research interests include Optical image processing, Mechatronic Engineering.