Semantic Scene Completion with Cleaner Self

Fengyun Wang[1] Dong Zhang[2] Hanwang Zhang[3] Jinhui Tang[1*] Qianru Sun[4]

[1]School of Computer Science and Engineering, Nanjing University of Science & Technology

[2]The Hong Kong University of Science & Technology [3]Nanyang Technological University [4]Singapore Management University

E-mail: {fereenwong, jinhuitang}@njust.edu.cn; dongz@ust.hk; hanwangzhang@ntu.edu.sg; qianrusun@smu.edu.sg

Abstract

Semantic Scene Completion (SSC) transforms an image of single-view depth and/or RGB 2D pixels into 3D voxels, each of whose semantic labels are predicted. SSC is a well-known ill-posed problem as the prediction model has to "imagine" what is behind the visible surface, which is usually represented by Truncated Signed Distance Function (TSDF). Due to the sensory imperfection of the depth camera, most existing methods based on the noisy TSDF estimated from depth values suffer from 1) incomplete volumetric predictions and 2) confused semantic labels. To this end, we use the ground-truth 3D voxels to generate a perfect visible surface, called TSDF-CAD, and then train a "cleaner" SSC model. As the model is noise-free, it is expected to focus more on the "imagination" of unseen voxels. Then, we propose to distill the intermediate "cleaner" knowledge into another model with noisy TSDF input. In particular, we use the 3D occupancy feature and the semantic relations of the "cleaner self" to supervise the counterparts of the "noisy self" to respectively address the above two incorrect predictions. Experimental results validate that our method improves the noisy counterparts with 3.1% IoU and 2.2% mIoU for measuring scene completion and SSC, and also achieves new state-of-the-art accuracy on the popular NYU dataset. The code is available at https://github.com/fereenwong/CleanerS.

1. Introduction

3D scene understanding is an important visual task for many practical applications, *e.g.*, robotic navigation [16] and augmented reality [55], where the scene geometry and semantics are two key factors to the agent interaction with the real world [24, 64]. However, visual sensors can only perceive a partial world given their limited field of view with sensory noises [49]. Therefore, an agent is expected to leverage prior knowledge to estimate the complete geome-

try and semantics from the imperfect perception. Semantic Scene Completion (SSC) is designed for such an ability to infer complete volumetric occupancy and semantic labels for a scene from a single depth and/or RGB image [49, 52].

Based on an input 2D image, the 2D→3D projection is a vital bond for mapping 2D perception to the corresponding 3D spatial positions, which is determined by the depth value [6]. After this, the model recovers the visible surface in 3D space, which sheds light on completing and labeling the occluded regions [31, 52], because the geometry of the visible and occluded areas is tightly intertwined. For example, you can easily infer the shapes and the semantic labels when you see a part of a "chair" or "bed". Thus, a high-quality visible surface is crucial for the SSC task.

However, due to the inherent imperfection of the depth camera, the depth information is quite noisy, what follows is an imperfect visible surface that is usually represented by Truncated Signed Distance Function (TSDF) [52]. In general, the existing depth noises can be roughly categorized into the following two basic types:

1) Zero Noise. This type of noise happens when a depth sensor cannot confirm the depth value of some local regions, it will fill these regions with zeroes [14, 43]. Zero noise generally occurs on object surfaces with reflection or unevenness [41]. Based on zero noise, the visible surface will be incomplete after the 2D-3D projection via TSDF [49], so the incomplete volumetric prediction problem may occur in the final 3D voxels. For example, as shown in the upper-half of Figure 1, for the input RGB "kitchen" image, the depth value of some parts of the "cupboard" surface (marked with the red dotted frames) (in (b)) is set to zero due to reflections. Based on this, both the visible surface (in (d)) and the predicted 3D voxels (in (f)) appear incomplete in reflective regions of this "cupboard". Our method uses the perfect visible surface (in (e)) generated by the noise-free ground-truth depth value (in (c)) as intermediate supervision in training , which helps the model to estimate "cupboard" 3D voxels in inference even with the noisy depth value as input.

2) Delta Noise. This type of noise refers to the inevitable deviation of the obtained depth value due to the inherent

---

*Corresponding author.

(1) Zero Noise Leads to Incomplete Volumetric Predictions

(a) RGB   (b) Noisy Depth   (c) Depth-CAD   (d) Noisy TSDF   (e) TSDF-CAD   (f) Prediction with (a, d)   (g) Our Prediction

(2) Delta Noise Leads to Confused Semantic Labels

furniture✗   chair✗   table✓

(a) RGB   (b) Noisy Depth   (c) Depth-CAD   (d) Noisy TSDF   (e) TSDF-CAD   (f) Prediction with (a, d)   (g) Our Prediction

○··○ / ○··○ noisy / noise-free depth value to visible surface   ○ / ○ incomplete / complete visible surface   ● / ● visible surface with class label
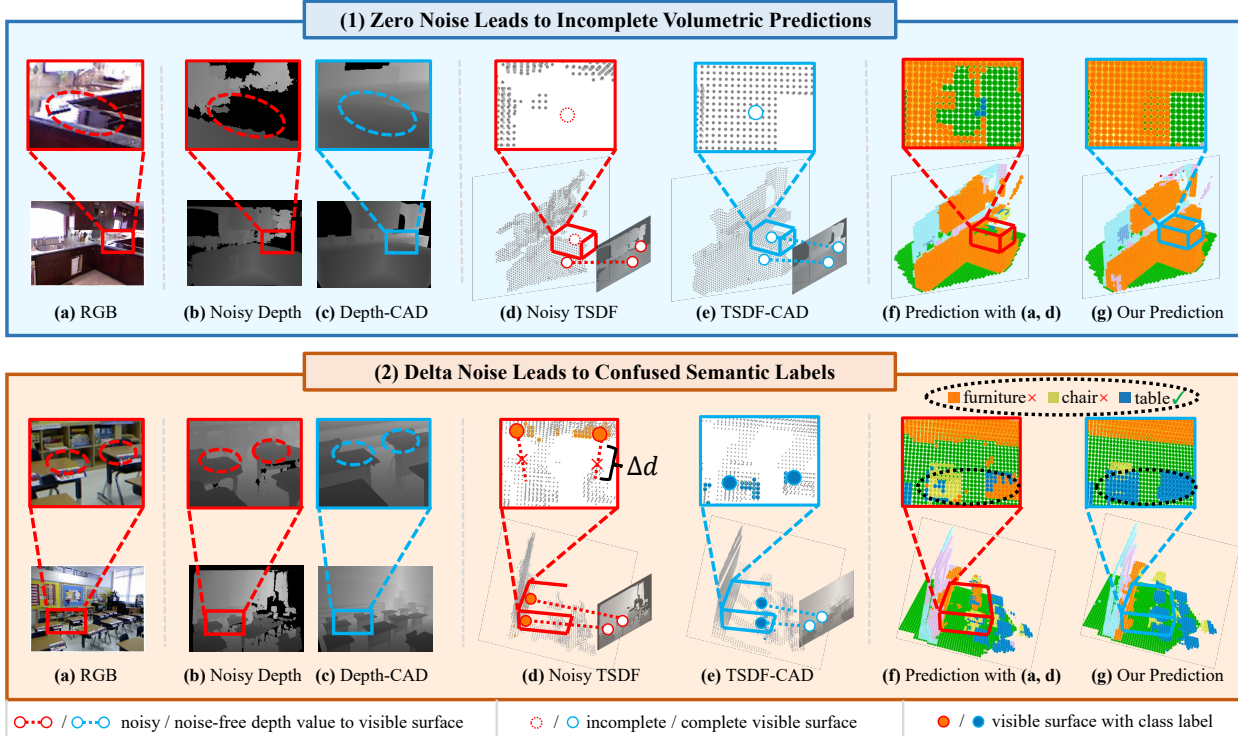
Figure 1. The existing depth noises can be roughly categorized into: 1) **zero noise** and 2) **delta noise**. By **zero noise**, we mean that when the depth camera cannot confirm the depth value of some local regions, it fills these regions with zeroes, leading to the problem of incomplete volumetric predictions. By **delta noise**, we mean the inevitable deviation (*i.e.*, Δd) of the obtained depth value due to the inherent quality defects of the depth camera, which leads to the problem of confused semantic labels in the final 3D voxels. In the above blocks, the pairwise subfigures (e.g., (d) and (e)) show the cases of "with noise" and "without noise" on the left and right, respectively.

quality defects of the depth camera [41], *i.e.*, the obtained depth value does not match the true depth value. Delta noise shifts the 3D position of the visible surface, resulting in the wrong semantic labels, such that the final 3D voxels will suffer from the problem of confusing semantic labels [52]. A real delta noise case is shown in the bottom half part of Figure 1. For the input RGB "classroom" image, the depth camera mistakenly estimates the depth value of the "table" as the depth value of "furniture" (in (b)). Therefore, the visible surface represented by TSDF shifts from the class of "table" (marked with blue points) to the class of "furniture" (marked with orange points in (d)). Based on this, the final estimated 3D voxels (in (f)) also mistakenly estimate the part of the "table" as the "furniture". In comparison, when our SSC model is trained on the visible surface in (e), which is generated by the correct depth value in (c), as the intermediate supervision, semantic labels for both the "table" and the "furniture" can be estimated correctly in (g).

In practice, these two types of noise are randomly mixed together to form a more complex noise [14, 65]. To handle these two noise types, although some recent SSC attempts have been made by rendering the noise-free depth value from 3D voxel ground-truth [12, 51], they are not of practi-

cal use as the 3D voxels ground-truth is still needed in inference. However, they indeed validate the potential that more accurate recognition performance can be achieved using the noise-free depth value [4, 56, 66]. To the best of our knowledge, no prior work focuses on mitigating the noisy depth values in SSC without the use of ground-truth depth values in inference. Therefore, the crux is to transfer the clean knowledge learned from ground-truth depth into the noisy-depth pipeline only during training. So, in inference, we can directly use this pipeline without the need for ground-truth.

In this paper, we propose a Cleaner Self (CleanerS) framework to shield the harmful effects of the two depth noises for SSC. CleanerS consists of two networks that share the same network architecture (that is what "self" means). The only difference between these two networks is that the depth value of the teacher network is rendered from ground-truth, while the depth value of the student network is inherently noisy. Therefore, the depth value of the teacher network is cleaner than the depth value of the student network. In the training stage, we make the teacher network to provide intermediate supervision for learning of the student network via knowledge distillation (KD), such that the student network can disentangle the clean visible surface

reconstruction and occluded region completion. To preserve both the detailed information and the abstract semantics of the teacher network, we adopt both **feature-based** and **logit-based** KD strategies. In inference, only the student network is used. Compared to the noisy self, as shown in Figure 1, CleanerS achieves more accurate performance with the help of ground-truth depth values in training but not in testing.

The main contributions of this work are summarized as the following two aspects: 1) we propose a novel CleanerS framework for SSC, which can mitigate the negative effects of the noisy depth value in training; 2) CleanerS achieves the new state-of-the-art results on the challenging NYU dataset with the input of noisy depth values.

2. Related Work

Semantic Scene Completion. SSCNet [52] is the pioneering work to predict both volumetric occupancies and semantic labels for completing 3D scenes. Follow-up works can be divided into four categories: 1) *Volume-based methods*. With TSDF as input [9, 52], these methods use 3D CNNs for volumetric and semantic predictions. The disadvantage is the high computational cost of 3D CNNs [7, 11]. 2) *View-volume-based methods*. With RGB image and depth value as inputs, these methods use 2D CNNs to extract features, and then reshape the features to 3D features by a 2D-3D projection layer for predicting 3D voxels [20, 31, 33, 36, 38]. These methods are less effective in extracting 3D geometry information due to the limitations of 2D CNNs. 3) *Point-based methods*. SPCNet [70] is a classic method in this category. It particularly resolves the problem of discretization in voxels and predicts SSC by using points as input. However, it is sensitive to point-label misalignment (*i.e.*, delta noise). 4) *Hybrid methods*. Many recent works [3, 46, 47, 54] take at least two kinds of inputs from the set of RGB image, depth value, TSDF, and points. They use dual-branch network architectures for feature fusion and achieve state-of-the-art performance. Our work belongs to this category using RGB image and TSDF as input. We highlight that our key difference with the above works is that we are the first to particularly resolve the problem of noisy TSDF estimated from noisy depth values.

Learning with Noises. Both image collection and image annotation bring *data noise* and *label noise* to computer vision datasets, respectively [2, 50]. Specifically, *data noises* are usually caused by the defects of sensors (*e.g.*, limited visual fields [22] and inaccurate cameras [41]) or data collection environments (*e.g.*, fog [17], rain [5], and nighttime [48]). To reduce the negative effects on the learning of models, the mainstream idea is to learn clean [58] and robust [1, 32] feature representations [63, 65]. Existing methods can be divided into two categories: noise-cleaning based methods [35, 44, 69] and robust-modeling based meth-

ods [42, 60, 61]. In particular, the existing models tend to result in unsatisfactory performance when the noises are unpredictable. Our method aims to address the problem of noisy data for SSC. We propose to transfer the clean knowledge learned from the ground-truth depth value into the noisy-depth pipeline during training. In the inference phase, we can directly use this pipeline without ground-truth.

Knowledge Distillation (KD). KD is proposed in [23] to transfer the knowledge from a teacher model (a stronger or larger model) to a student model (a weaker or smaller model). The original motivation is model compression. After KD, the student can achieve a competitive or even superior performance [34, 64]. KD can be roughly categorized into 1) feature-based [26, 27], logit-based [28, 68], and hybrid [18, 25]. In this work, we leverage KD for a new purpose: denoising in the SSC task. The teacher model learns cleaner knowledge than the student model with noisy input. This enables the teacher model to provide intermediate supervision for learning the student model via KD.

3. Depth Noises in SSC

In this section, we first compute the percentage of the two noise types: zero noise and delta noise, taking the standard datasets NYU [51] and NYUCAD [13] as examples.

Then, we show the performance gap between the learned models *with* and *without* these noises. The aim is to quantitatively demonstrate that these two kinds of noises ignored by the existing work of SSC are unfortunately making severe negative effects on the learning of SSC models.

The Quantity of Depth Noises. In Figure 2 (b) and Figure 2 (a), we show the class-wise quantities of zero noises and delta noises, respectively, on NYU [51] (including both training and test sets). Each class is represented by a unique color, and the correspondence between colors and class labels is given in Figure 2 (c). The way of calculating the percentage of the zero noise on each semantic class of the depth value is expressed as:

$$Zero(c) = \frac{\sum \mathbb{1}_{(Y(d)=c, d \neq 0, d'=0)}}{\sum \mathbb{1}_{(Y(d)=c, d \neq 0)}}, \quad (1)$$

where $\mathbb{1}$ is an indicator (*i.e.*, when conditions in the right parentheses are all satisfied, the current value is incremented by one). d' and d denote the noisy and the noise-free depth value, respectively. $Y(d)$ is the class label of the corresponding visible surface whose 3D position is decided by d. c denotes a certain semantic class. The zero noise rate of each class is shown as a pie chart in Figure 2 (b). The way of calculating the percentage of delta noise between c (class of the clean visible surface) and c' (class of the noisy visible surface) is formulated as:

$$Delta(c, c') = \frac{\sum \mathbb{1}_{(Y(d)=c, (d \cdot d') \neq 0, Y(d')=c', c \neq c')}}{\sum \mathbb{1}_{(Y(d)=c, (d \cdot d') \neq 0)}}. \quad (2)$$

(a) The percentage of confused class due to delta noise

(c) IoU@NYUCAD minus IoU@NYU
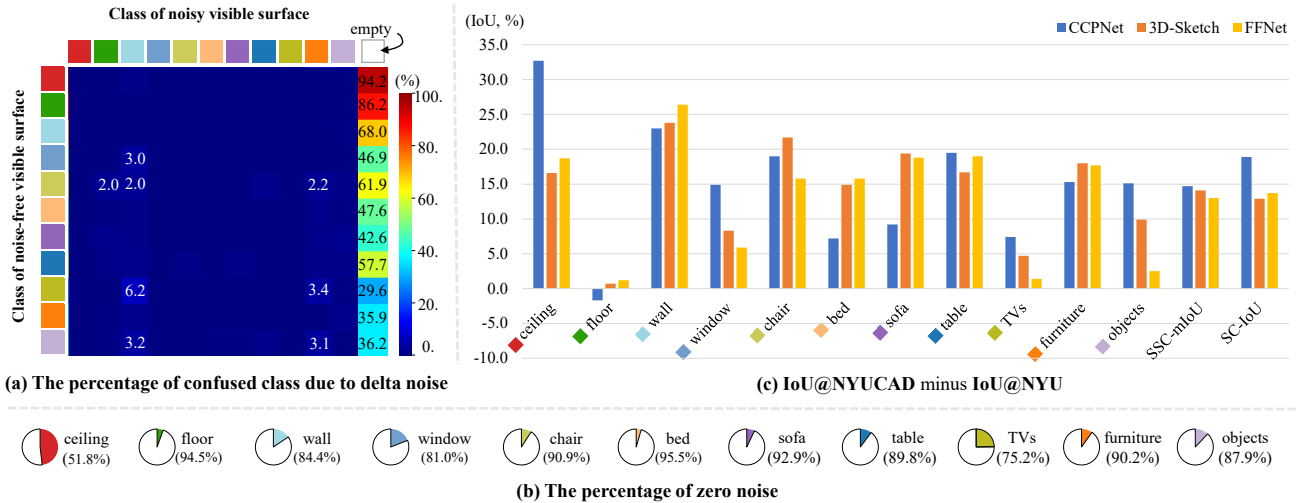
(b) The percentage of zero noise

Figure 2. Quantity of depth noises in NYU [51]. (a) the percentage of the delta noise in each object class of the reconstructed visual surface. (b) the percentage of the zero noise on each object class of the depth value. (c) the models' estimation performance gap on the noisy NYU and the noise-free NYUCAD [13] datasets, where CCPNet [66], 3D-Sketch [4] and FFNet [56] are three baselines. The performance gap is obtained from results on NYUCAD minus results on NYU. "IoU" denotes the evaluation metric of Intersection over Union.

It is visualized as a block (c row and c' column) on the confusion matrix in Figure 2 (a).

We can observe from Figure 2 (b) that all classes in NYU [51] have surprisingly significant zero noise rates, *i.e.*, between 51.8% and 95.5%. This means the model trained on such data might be biased by the incomplete visible surface and resulted in an incomplete prediction. Besides, from Figure 2 (a), we can observe that the main semantic confusion (caused by delta noises) is between any semantic class and the special class "empty", ranging from 29.6% (between "empty" and "TVs") to 94.2% (between "empty" and "ceiling"). Besides, among the non-empty semantic classes of the noisy visible surface, we find that: 1) several classes confused with the "wall" class; 2) the class "chair" is easily confused with other non-empty classes. The reasons for these phenomena may be that: 1) NYU is an indoor dataset, the visible surfaces of objects are easily shifted to the wall; 2) the chair is not face-structured, and it is difficult to obtain accurate depth values.

The Effects of Depth Noises. Figure 2 (c) shows the performance gap, which is is obtained from the experimental results on the noise-free NYUCAD dataset minus the experimental results on the noisy NYU dataset. We choose three top-performing methods, CCPNet [66], 3D-Sketch [4] and FFNet [56] as the baseline models. Each bar denotes a specific method, *e.g.*, the blue bar for CCPNet [66], the orange bar for 3D-Sketch [4], and the yellow bar denotes FFNet [56], separately. For each method, we present the difference of IoU according to the results reported in its paper, which are respectively trained on the NYUCAD [13] and NYU [51] datasets. In addition to IoU for each seman-

tic class, SSC mean IoU (SSC-mIoU) and SC-IoU are also used as overall evaluation metrics. As in Figure 2 (c), most of the performance gaps on IoU exceed 10%, and some of them even exceed 20%, such as CCPNet [66] in "ceiling", and FFNet [56] in "wall", which are significant for SSC. We can also observe that there is a large performance gap between the experimental results *with* or *without* noise from SSC-mIoU and SC-IoU. The above quantitative analysis validates that the depth noises have indeed caused substantial damage to the experimental results of SSC models.

4. Cleaner Self (CleanerS)

In this section, we introduce the implementation details of our approach CleanerS. Section 4.1 presents the overall framework of CleanerS. Section 4.2 elaborates on the feature-based as well as the logit-based KD pipelines implemented in CleanerS. Section 4.3 introduces the overall training losses used in CleanerS.

4.1. Overall Framework

Figure 3 illustrates the overall framework of CleanerS, consisting of two networks: a teacher network T, and a student network S. The two networks share the same architectures of 2D and 3D processing modules. The training process of CleanerS includes the following two steps. *Step 1* is a fully supervised training of T on noise-free depth values that are rendered from 3D voxels ground-truth. *Step 2* includes the fully supervised training of S on noisy depth values, as well as feature-based and logit-based KD training between teacher T and student S.
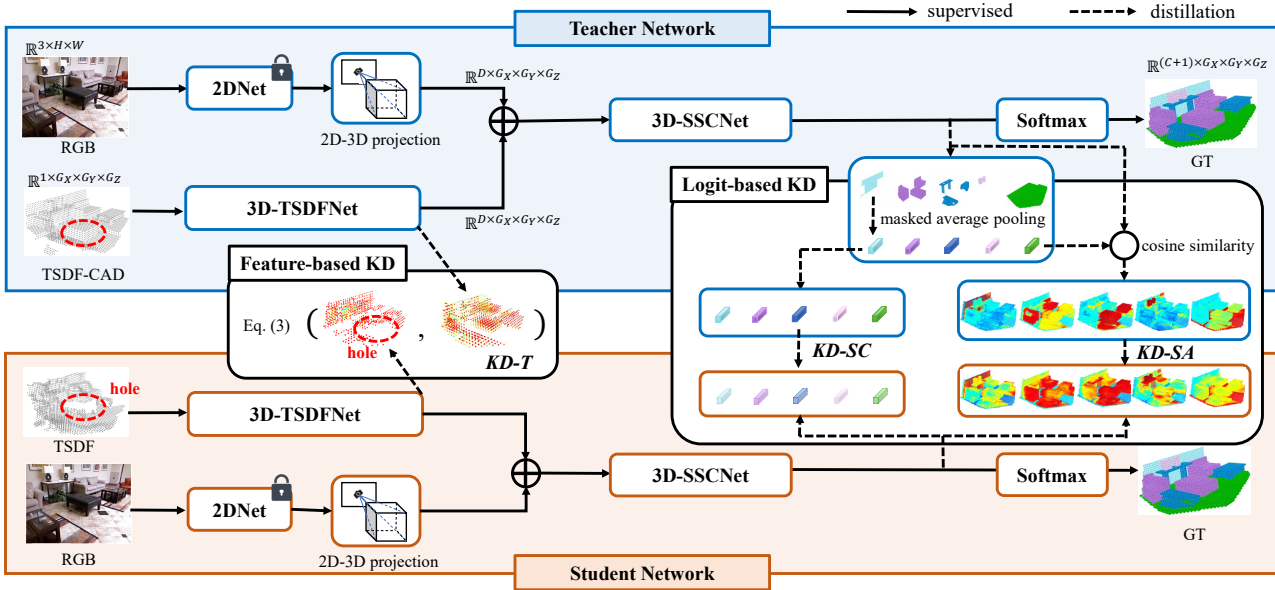
Figure 3. Overall architecture of our proposed CleanerS, consisting of two networks: a teacher network, and a student network. The two networks share the same architectures but have different weights. The distillation pipelines include a feature-based cleaner surface distillation (*i.e.*, *KD-T*), and logit-based cleaner semantic distillations (*i.e.*, *KD-SC* and *KD-SA*). The dimensions of the inputs and outputs in the student network are omitted as they are the same as in the teacher network.

Specifically, for each network, the input consists of an RGB image and a TSDF volume (noise-free TSDF-CAD for T, and noisy TSDF for S). The outputs are 3D voxels, each of which contains volumetric occupancy and semantic labels. In the following, we introduce the three modules in the network: 1) Image feature extraction and reformation module. This module has a 2D network to extract image features (a D-channel 2D feature $\mathbf{F}_r$) and a 2D to 3D projection layer that translates 2D feature to 3D feature $\mathbf{V}_r \in \mathbb{R}^{D \times G_X \times G_Y \times G_Z}$. G_X, G_Y, G_Z are the sizes of 3D voxels. The 2D to 3D projection layer is used to lift the feature vector at 2D position (u, v) in $\mathbf{F}_r$ to the corresponding 3D position (x, y, z) according to the depth values of all visible surfaces (otherwise filled by zero vectors). 2) TSDF feature extraction module. This module transforms the 1-channel 3D input TSDF into a D-channel 3D feature $\mathbf{V}_t$, where $\mathbf{V}_t \in \mathbb{R}^{D \times G_X \times G_Y \times G_Z}$. 3) SSC prediction module. This module is used to fuse both features $\mathbf{V}_r$ and $\mathbf{V}_t$ by element-wise addition, and then makes predictions via 3D-SSCNet [4]. The prediction is denoted as $\hat{\mathbf{Y}} \in \mathbb{R}^{(C+1) \times G_X \times G_Y \times G_Z}$, where C is the number of classes in the dataset, "+1" is for the volumetric occupancy, *i.e.* "empty" or not.

4.2. Distilling "Cleaner" Knowledge

In the following, we introduce the implementation details of feature-based KD and logit-based KD.

Feature-Based KD. We use TSDF features for feature-based KD, as follows,

$$L_{KD-T} = MSE(\mathbf{V}_t^S, \mathbf{V}_t^T), \tag{3}$$

where MSE denotes mean square error. $\mathbf{V}_t^S$ and $\mathbf{V}_t^T$ are the TSDF features output by the 3D-TSDFNets of student and teacher, respectively. For visualization (Figure 3), these features are flattened and normalized with a softmax layer along the spatial dimension, and then re-scaled their values into $(0, 1)$. In the "Feature-based KD" block of Figure 3, we visualize the features of large values (> 0.1). Intuitively, minimizing L_{KD-T} encourages the student to learn to complete the visible "hole" in $\mathbf{V}_t^S$ (as shown by the red dashed circle in Figure 3).

Logit-Based KD. Teacher and student have different inputs, *i.e.*, clean TSDF-CAD *vs.* noisy TSDF. This results in a large output gap. It is thus ineffective if forcing straightforward distillation between teacher's and student's prediction logits. We show an empirical validation in Section 5.4. On the other hand, teacher and student have the common RGB image input, and aim to predict 3D voxels of the same semantic scene.

Motivated by these, we design two strategies, Semantic-Centered distillation (*KD-SC*) and Semantic Affinity distillation (*KD-SA*), to distill clean semantic knowledge from teacher to student via their prediction logits. *KD-SC* is a semantic-centered distillation to transfer global semantic knowledge (from teacher to student). But it ignores local structures. *KD-SA* is complementary to *KD-SC*. It is

a voxel-to-center affinity distillation to transfer the local structure knowledge. Both of them are based on prediction logits, denoted as $\bar{\mathbf{Y}}^T$ (teacher) and $\bar{\mathbf{Y}}^S$ (student), as elaborated in the following.

KD-SC. Given an input sample, we use its ground-truth $\mathbf{Y}$ to generate a binary foreground mask $\mathbf{M}_c \in \mathbb{R}^{G_X \times G_Y \times G_Z}$ for each class c. In M_c, each voxel stores 1 if its semantic label is c, and otherwise 0. Then, we distill the semantic-centered logits for each sample by using the loss:

$$L_{KD-SC} = \frac{1}{N_c} \sum_{c=0}^{C} KL(\bar{\mathbf{y}}_c^S, \bar{\mathbf{y}}_c^T), \quad (4)$$

where KL is the Kullback–Leibler divergence. $N_c \le (C+1)$ is the number of classes appearing on the input sample. $\bar{\mathbf{y}}_c^S$ and $\bar{\mathbf{y}}_c^T$ are the semantic-centered logits of class c in S and T, respectively These logits are calculated based on the prediction logits by applying a masked average pooling along the spatial dimension. E.g., in the student network, they can be derived as:

$$\bar{\mathbf{y}}_c^S = \frac{\sum_{i=1}^{G_X \cdot G_Y \cdot G_Z}(\bar{\mathbf{Y}}^S[i] \cdot \mathbf{M}_c[i])}{\sum_{i=1}^{G_X \cdot G_Y \cdot G_Z} \mathbf{M}_c}, \quad (5)$$

where i is the voxel index along the spatial dimension. Due to delta noises, $\bar{\mathbf{y}}_c^S$ will include certain features of the voxels that do not belong to the class c into the average pooling. This results in noisy semantic-centered logits. Similarly, we can calculate $\bar{\mathbf{y}}_c^T$ for the teacher network, which is "cleaner" than $\bar{\mathbf{y}}_c^S$.

KD-SA. To encode local structures, we calculate the voxel-to-center affinity matrices between every voxel and the semantic-centered logits, and then conduct the distillation from T to S, as follows,

$$L_{KD-SA} = \frac{1}{N_c} \sum_{c=0}^{C} MSE(\mathbf{A}_c^S, \mathbf{A}_c^T), \quad (6)$$

where $\mathbf{A}_c^S$, and $\mathbf{A}_c^T$ are the voxel-to-center affinity matrices, which are calculated in the student and teacher networks, respectively. E.g., in the student network, the affinity score $a_c^S \in \mathbf{A}_c^S$ is calculated between the semantic-centered logits $\bar{\mathbf{y}}_c^S$ and the logits of each voxel $\bar{\mathbf{y}}^S \in \bar{\mathbf{Y}}^S$ as:

$$a_c^S = cos(\bar{\mathbf{y}}_c^S, \bar{\mathbf{y}}^S), \quad (7)$$

where cos denotes cosine similarity, $\mathbf{A}_c^S \in \mathbb{R}^{G_X \times G_Y \times G_Z}$ is the resultant affinity matrix. The higher affinity value indicates a stronger semantic correlation between the voxel and the center of its label class, and vice versa.

4.3. Overall Loss

The overall loss of CleanerS consists of two parts: SSC loss and KD loss. The SSC loss is applied in both teacher and student networks. First, we minimize the difference between prediction $\hat{\mathbf{Y}}$ and ground-truth $\mathbf{Y}$. Then, we incorporate the following additional supervision for 2DNet. For the 2D feature $\mathbf{F}_r$, we add a 3×3 convolution for 2D semantic prediction, which outputs $\hat{\mathbf{Y}}_{2D} \in \mathbb{R}^{(C+1) \times H \times W}$. The 2D semantic ground-truth is noted as $\mathbf{Y}_{2D}$, the class of each pixel at 2D coordinate $[u, v]$ is $\mathbf{Y}[x, y, z]$, where $[x, y, z]$ is the corresponding 3D position after 2D-3D projection. Hence, the SSC loss is formulated as:

$$L_{SSC} = SCE(\hat{\mathbf{Y}}, \mathbf{Y}) + \lambda SCE(\hat{\mathbf{Y}}_{2D}, \mathbf{Y}_{2D}), \quad (8)$$

where SCE denotes smooth cross entropy loss [53], and λ is a coefficient to balance between 3D and 2D semantic losses. The KD loss is applied in the student network. Therefore, the overall loss for the student network can be formulated as:

$$L_{all}^S = L_{SSC} + \beta \cdot (L_{KD-T} + L_{KD-S}), \quad (9)$$

where $L_{KD-S} = L_{KD-SC} + L_{KD-SA}$ is the logit-based KD loss, and β is a coefficient to balance between SSC and distillation losses.

5. Experiments

5.1. Datasets and Evaluation Metrics

Datasets. Following [4, 15, 52], we use the challenging NYU [51] dataset and its variants NYUCAD [13] dataset in our experiments. NYU [51] consists of 1,449 RGB-D images (795 for training, and 654 for testing) captured via a Kinect sensor [67]. NYU is composed of mainly office and house room scenes. The 3D voxel ground-truth is generated by voxelizing the computer-aided design (CAD) mesh annotations provided by [19]. The difference between the two datasets is that the depth values of NYUCAD are rendered from 3D voxel ground-truth, but the depth values of NYU are noisy. We use the training set of NYUCAD to train the teacher network, and use training and testing sets of NYU to respectively train and evaluate the student network.

Evaluation Metrics. Following [4, 15, 52], we use Precision (Prec.), Recall, and Intersection over Union (IoU) as evaluation metrics for two tasks: semantic scene completion (SSC), and scene completion (SC). For SSC, we evaluate the IoU of each category on both observed and occluded voxels, and mean IoU across all categories. For SC, we treat every voxel as a binary prediction task, *i.e.*, empty or non-empty, and we evaluate the performance on the occluded 3D voxels.

5.2. Implementation Details

Network Architectures. For 2DNet, we adopt the "B2" Segformer [59] as baseline, and initialize it with ImageNet [8] pretrained weights. For 3D-TSDFNet, we follow

Methods	Inputs	Scene Completion IoU(%)			Semantic Scene Completion mIoU(%)											
		Prec.	Recall	IoU	ceil.	floor	wall	win.	chair	bed	sofa	table	TVs	furn.	objs.	avg.
SSCNet [52]	TSDF	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
ForkNet [57]	TSDF	-	-	63.4	36.2	93.8	29.2	18.9	17.7	61.6	52.9	23.3	19.5	45.4	20.0	37.1
CCPNet [66]	TSDF	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
DDRNet [31]	RGB+D	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
AIC-Net [30]	RGB+D	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
IPF-SPCNet [70]	RGB+Point	70.5	46.7	39.0	32.7	66.0	41.2	17.2	34.7	55.3	47.0	21.7	12.5	38.4	19.2	35.1
IMENet [29]	RGB+D	**90.0**	78.4	72.1	43.6	93.6	42.9	31.3	36.6	57.6	48.4	32.1	16.0	**47.8**	**36.7**	44.2
FFNet [56]†	RGB+D+TSDF	89.3	78.5	71.8	44.0	93.7	41.5	29.3	36.2	59.0	51.1	28.9	26.5	45.0	32.6	44.4
3D-Sketch [4]†	RGB+TSDF	85.0	81.6	71.3	43.1	93.6	40.5	24.3	30.0	57.1	49.3	29.2	14.3	42.5	28.6	41.1
CleanerS(Res50†)	RGB+TSDF	89.9	79.6	73.1	**48.2**	94.0	41.5	29.9	34.4	60.7	49.7	30.9	33.9	42.7	31.0	45.2
CleanerS⋆	RGB+TSDF	88.0	83.5	**75.0**	46.3	93.9	**43.2**	**33.7**	**38.5**	**62.2**	54.8	**33.7**	**39.2**	45.7	33.8	**47.7**

Table 1. Result comparisons with the state-of-the-art methods on the test set of NYU [51] (*take the noisy depth value as input*). Results with "†" and "⋆" denote that these results are based on ResNet50 [21] and Segformer-B2 [59] as the backbone network for RGB image feature extraction, respectively. Bold numbers represent the best performance.

3D-Sketch [4] to use the following architecture: three layers of 3D convolution; two DDR blocks [31] with feature downsampling by a rate of 4; and two layers of 3D deconvolution for upsampling. For 3D-SSCNet, we use the same architecture as 3D-TSDFNet except that we remove the first three layers of 3D convolution. *Please refer to the supplementary materials for the detailed architecture of each net.* The feature channel size in both 3D-TSDFNet and 3D-SSCNet is set to $D = 256$, and the 3D resolution of prediction (G_X, G_Y, G_Z) is set to $(60, 36, 60)$.

Training Settings. We implement all experiments in this work on the PyTorch [45] with 2 NVIDIA 3090 GPUs. Our model is trained by AdamW [40] with a weight decay of 0.05. The batch size is 4 and the learning rate is initially set as 0.001 and scheduled by a cosine learning rate decay policy [39] with a minimum learning rate of $1e$-7. We train our model for 100 epochs. For each input RGB image, we apply resize, random cropping, and random flipping as 2D data augmentation. For 3D data augmentation, we use random x-axis and z-axis flipping as in [10]. In our loss function, both λ and β are set as 0.25.

5.3. Comparisons with State-of-the-Arts

Table 1 shows the quantitative results of our CleanerS compared to state-of-the-art methods on the test set of NYU. We can see that CleanerS outperforms these methods by clear margins. In particular, it achieves the improvements of 3.2% SC-IoU and 3.3% SSC-mIoU compared to the top-performing method FFNet [56], and becomes the new state-of-the-art. Another closely related work is 3D-Sketch [4]. It is also a voxel-based framework with the inputs of RGB image and TSDF. However, 3D-Sketch uses ResNet50 for 2D feature extraction. To compare with it fairly, we implement the 2DNet in our method with ResNet50 [21], and name it as CleanerS (Res50) in Ta-

ble 1. Results show that our CleanerS (Res50) outperforms 3D-Sketch by 1.8% SC-IoU and 4.1% SSC-mIoU.

Qualitative Results. Figure 4 demonstrates the qualitative results compared to the state-of-the-arts methods, including SSCNet [52] and 3D-Sketch [4]. On the first row, zero noises in depth values cause incomplete surfaces (in (b)). As a result, both SSCNet and 3D-Sketch can hardly "imagine" the objects on the incomplete surface, resulting in an incomplete prediction in SSCNet [52] and wrong semantic predictions in 3D-Sketch [4]. In the second row, delta noises cause confusion about semantic classes. Thus, both SSCNet and 3D-Sketch are confused about the semantics between the "furniture" and the "table", as well as between the "furniture" and the "sofa". Compared to them, we explicitly handle these two kinds of noises, and achieve more accurate predictions as shown in (e) and (f).

5.4. Ablation Study

CleanerS Components. In Table 2, the $1st$ row shows the performance of the teacher model with inputs of RGB and TSDF-CAD (*without noise*). It can be taken as the upper bound of the student model. The $2nd$ row shows the performance of the baseline taking *noisy depth values* as inputs. It is the lower bound. It is clear that there is a great gap between the upper and lower bounds. The $3rd$ row lists the performance of our CleanerS with **feature-based KD** only. It shows a significant improvement at SC-IoU (+2.7%), validating that CleanerS helps completing the scene and gets higher occupancy. The $4th$ row shows the performance of our CleanerS with **logit-based KD** only. The model learns from the semantic-centered logits and semantic affinities between voxels and class centers given by the teacher network. It achieves a clear improvement, i.e., 1.7 percentage points on SSC-mIoU, for the semantic prediction task SSC. It also brings a surprising improvement

(a) RGB (Depth) **(b)** TSDF **(c)** SSCNet **(d)** 3D-Sketch **(e)** CleanerS(Res50) **(f)** CleanerS **(g)** Ground-Truth
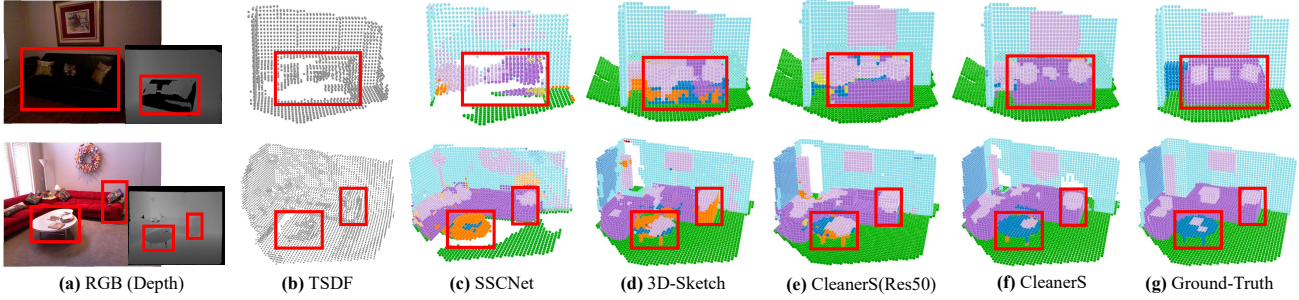
Figure 4. Qualitative comparisons with state-of-the-art methods, including SSCNet [52] and 3D-Sketch [4]. We show that 1) the incomplete visible surface will lead to incomplete prediction (in the 1_{st} row of (c)); 2) the models trained with the supervision of confused class labels will confuse the object semantics in inference (1_{st} row (d) and 2_{nd} row (c, d)).

Methods	Inputs	L_{SSC}	L_{KD-T}	L_{KD-S}	SC-IoU	SSC-mIoU
Teacher	RGB+TSDF-CAD	✓			85.3%	59.4%
Baseline	RGB+TSDF	✓			71.9%	45.5%
CleanerS	RGB+TSDF	✓	✓		74.6%	46.0%
CleanerS	RGB+TSDF	✓		✓	74.0%	47.2%
CleanerS	RGB+TSDF	✓	✓	✓	75.0%	47.7%

Table 2. Ablation study results of different components in our CleanerS on the test set of NYU [51].

Methods	L_{KD-SC}	L_{KD-SA}	$L_{KD-Soft}$	SC-IoU	SSC-mIoU
Baseline				71.9%	45.5%
CleanerS			✓	73.7%	46.3%
CleanerS		✓		73.6%	46.3%
CleanerS	✓			74.0 %	46.7%
CleanerS	✓	✓		74.0%	47.2%

Table 3. The results of different semantic distillation strategies in our CleanerS on the test set of NYU [51].

on SC-IoU (2.1 percentage points). The reason is that the TSDF-CAD features are fused for final prediction, and this enables an implicitly cleaner surface distillation. Combining both feature- and logit-based KDs achieves the best results. 1) it boosts the best SSC mIoU (in 4_{th} row) by an extra 0.5%. The more accurate completion means more true positives on volumetric occupancy. This provides the model a chance to infer correct semantic labels; 2) it achieves a further gain of 0.4% above the best SC-IoU (in 3_{rd} row). The accurate semantic prediction in turn enhances the prediction of shapes of occluded regions. Overall, our CleanerS gains the improvements of 3.1% SC-IoU and 2.2% SSC-mIoU, compared to the baseline.

Logit-based KD *vs*. Soft-target KD. As mentioned in the second paragraph of Section 4.2, directly forcing straightforward distillation is ineffective. To validate this, we compare our logit-based distillation (including L_{KD-SC} and L_{KD-SA}) with soft target distillation [23], and we denote the loss of the latter one as $L_{KD-Soft}$. From the results in Table 3, we can observe that the L_{KD-SC} plays a major role, and $L_{KD-Soft}$ achieves similar performance as using L_{KD-SA}. More specifically, we have the following conclusions: 1) it is sub-optimal to apply only a local voxel-wise logits distillation, and 2) it is a better choice to distill the global semantic center embedding. The reasons might be that the inputs to teacher and student models are not exactly the same, clean TSDF-CAD *v.s.* noisy TSDF, and it is hard to match their outputs in a voxel-wise manner. Be-

sides, with the same RGB image input, both teacher and student models aim to infer 3D semantic voxels in the same scene, and it is thus reasonable to enforce the two models to predict similar semantic-centered logits for each semantic class.

6. Conclusions

We presented a novel CleanerS framework for tackling the task of SSC. We first computed the percentage of zero noise and delta noise in the depth values. Then, we showed that these two types of noises have caused severe negative effects on SSC models. In our CleanerS, we trained a teacher network to learn clean knowledge and provided intermediate supervision for the learning of a student network. Compared to the existing SSC methods, CleanerS can not only achieve more accurate performance but also requires no ground-truth labels in inference. Extensive experimental results validated that our CleanerS can shield the negative effects of noisy depth values, and achieve the new state-of-the-art on the challenging NYU dataset.

Acknowledgements

References

[1] Ehsan Adeli, Kim-Han Thung, Le An, Guorong Wu, Feng Shi, Tao Wang, and Dinggang Shen. Semi-supervised discriminative classification robust to sample-outliers and feature-noises. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 41(2):515–522, 2018. 3

[2] Benoit Brummer and Christophe De Vleeschouwer. Natural image noise dataset. In *IEEE Conference on Computer Vision and Pattern Recognition Workshops (ICCVW)*, 2019. 3

[3] Yingjie Cai, Xuesong Chen, Chao Zhang, Kwan-Yee Lin, Xiaogang Wang, and Hongsheng Li. Semantic scene completion via integrating instances and scene in-the-loop. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021. 3

[4] Xiaokang Chen, Kwan-Yee Lin, Chen Qian, Gang Zeng, and Hongsheng Li. 3d sketch-aware semantic scene completion via semi-supervised structure prior. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020. 2, 4, 5, 6, 7, 8, 12, 14

[5] Xiang Chen, Jinshan Pan, Kui Jiang, Yufeng Li, Yufeng Huang, Caihua Kong, Longguang Dai, and Zhentao Fan. Unpaired deep image deraining using dual contrastive learning. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022. 3

[6] Yun Chen, Bin Yang, Ming Liang, and Raquel Urtasun. Learning joint 2d-3d representations for depth completion. In *IEEE International Conference on Computer Vision (CVPR)*, 2019. 1

[7] Yueh-Tung Chen, Martin Garbade, and Juergen Gall. 3d semantic scene completion from a single depth image using adversarial training. In *IEEE International Conference on Image Processing (ICIP)*, 2019. 3

[8] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2009. 6

[9] Aloisio Dourado, Teofilo Emidio de Campos, Hansung Kim, and Adrian Hilton. Edgenet: Semantic scene completion from rgb-d images. *arXiv preprint arXiv:1908.02893*, 2019. 3

[10] Aloisio Dourado, Frederico Guth, and Teofilo de Campos. Data augmented 3d semantic scene completion with 2d segmentation priors. In *IEEE Winter Conference on Applications of Computer Vision (WACV)*, 2022. 7

[11] Aloisio Dourado, Hansung Kim, Teofilo E de Campos, and Adrian Hilton. Semantic scene completion from a single 360-degree image and depth map. In *International Conference on Computer Vision Theory and Application*, 2020. 3

[12] Michael Firman, Oisin Mac Aodha, Simon Julier, and Gabriel J Brostow. Structured prediction of unobserved voxels from a single depth image. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016. 2

[13] Michael Firman, Oisin Mac Aodha, Simon Julier, and Gabriel J Brostow. Structured prediction of unobserved voxels from a single depth image. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016. 3, 4, 6

[14] Jingjing Fu, Shiqi Wang, Yan Lu, Shipeng Li, and Wenjun Zeng. Kinect-like depth denoising. In *IEEE International Symposium on Circuits and Systems (ISCAS)*, 2012. 1, 2

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

[16] Sourav Garg, Niko Sünderhauf, Feras Dayoub, Douglas Morrison, Akansel Cosgun, Gustavo Carneiro, Qi Wu, Tat-Jun Chin, Ian Reid, Stephen Gould, et al. Semantics for robotic mapping, perception and interaction: A survey. *Foundations and Trends® in Robotics*, 8(1–2):1–224, 2020. 1

[17] Kristofor B Gibson and Truong Q Nguyen. Fast single image fog removal using the adaptive wiener filter. In *IEEE International Conference on Image Processing (ICIP)*, 2013. 3

[18] Jianping Gou, Baosheng Yu, Stephen J Maybank, and Dacheng Tao. Knowledge distillation: A survey. *International Journal of Computer Vision*, 129(6):1789–1819, 2021. 3

[19] Ruiqi Guo and Derek Hoiem. Support surface prediction in indoor scenes. In *IEEE International Conference on Computer Vision (ICCV)*, 2013. 6

[20] Yuxiao Guo and Xin Tong. View-volume network for semantic scene completion from a single depth image. In *International Joint Conference on Artificial Intelligence (IJCAI)*, 2018. 3

[21] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *IEEE conference on Computer Vision and Pattern Recognition (CVPR)*, 2016. 7, 12

[22] Peter Henry, Michael Krainin, Evan Herbst, Xiaofeng Ren, and Dieter Fox. Rgb-d mapping: Using depth cameras for dense 3d modeling of indoor environments. In *International Symposium on Experimental Robotics (ISER)*, 2014. 3

[23] Geoffrey Hinton, Oriol Vinyals, Jeff Dean, et al. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, 2015. 3, 8

[24] Ji Hou, Angela Dai, and Matthias Nießner. Revealnet: Seeing behind objects in rgb-d scans. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020. 1

[25] Yen-Chang Hsu, James Smith, Yilin Shen, Zsolt Kira, and Hongxia Jin. A closer look at knowledge distillation with features, logits, and gradients. *arXiv preprint arXiv:2203.10163*, 2022. 3

[26] Mingi Ji, Byeongho Heo, and Sungrae Park. Show, attend and distill: Knowledge distillation via attention-based feature matching. In *AAAI Conference on Artificial Intelligence (AAAI)*, 2021. 3

[27] Sangwon Jung, Donggyu Lee, Taeeon Park, and Taesup Moon. Fair feature distillation for visual recognition. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021. 3

[28] Youmin Kim, Jinbae Park, YounHo Jang, Muhammad Ali, Tae-Hyun Oh, and Sung-Ho Bae. Distilling global and local logits with densely connected relations. In *IEEE International Conference on Computer Vision (ICCV)*, 2021. 3

[29] Jie Li, Laiyan Ding, and Rui Huang. Imenet: Joint 3d semantic scene completion and 2d semantic segmentation through iterative mutual enhancement. *arXiv preprint arXiv:2106.15413*, 2021. 7

[30] Jie Li, Kai Han, Peng Wang, Yu Liu, and Xia Yuan. Anisotropic convolutional networks for 3d semantic scene completion. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020. 7

[31] 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 *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019. 1, 3, 7, 13

[32] Junnan Li, Caiming Xiong, and Steven CH Hoi. Learning from noisy data with robust representation learning. In *IEEE International Conference on Computer Vision (ICCV)*, 2021. 3

[33] Siqi Li, Changqing Zou, Yipeng Li, Xibin Zhao, and Yue Gao. Attention-based multi-modal fusion network for semantic scene completion. In *AAAI Conference on Artificial Intelligence (AAAI)*, 2020. 3

[34] Tianhong Li, Jianguo Li, Zhuang Liu, and Changshui Zhang. Few sample knowledge distillation for efficient network compression. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020. 3

[35] Jingyun Liang, Jiezhang Cao, Guolei Sun, Kai Zhang, Luc Van Gool, and Radu Timofte. Swinir: Image restoration using swin transformer. In *IEEE International Conference on Computer Vision (ICCV)*, 2021. 3

[36] 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 (NeurIPS)*, 2018. 3

[37] Yang Liu, Saeed Anwar, Zhenyue Qin, Pan Ji, Sabrina Caldwell, and Tom Gedeon. Disentangling noise from images: A flow-based image denoising neural network. *arXiv preprint arXiv:2105.04746*, 2021. 13

[38] Yu Liu, Jie Li, Qingsen Yan, Xia Yuan, Chunxia Zhao, Ian Reid, and Cesar Cadena. 3d gated recurrent fusion for semantic scene completion. *arXiv preprint arXiv:2002.07269*, 2020. 3

[39] Ilya Loshchilov and Frank Hutter. Sgdr: Stochastic gradient descent with warm restarts. *arXiv preprint arXiv:1608.03983*, 2016. 7

[40] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017. 7

[41] Tanwi Mallick, Partha Pratim Das, and Arun Kumar Majumdar. Characterizations of noise in kinect depth images: A review. *IEEE Sensors Journal*, 14(6):1731–1740, 2014. 1, 2, 3

[42] Mohammad Momeny, Mehdi Agha Sarram, AliMohammad Latif, and Razieh Sheikhpour. A convolutional neural network based on adaptive pooling for classification of noisy images. *Signal and Data Processing*, 17(4):139–154, 2021. 3

[43] Chuong V Nguyen, Shahram Izadi, and David Lovell. Modeling kinect sensor noise for improved 3d reconstruction and tracking. In *International Conference on 3D Imaging, Modeling, Processing, Visualization & Transmission (3DIMPVT)*, 2012. 1

[44] Jean Ollion, Charles Ollion, Elisabeth Gassiat, Luc Lehéricy, and Sylvain Le Corff. Joint self-supervised blind denoising and noise estimation. *arXiv preprint arXiv:2102.08023*, 2021. 3

[45] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. Pytorch: An imperative style, high-performance deep learning library. *Advances in Neural Information Processing Systems (NeurIPS)*, 2019. 7

[46] Christoph B Rist, David Emmerichs, Markus Enzweiler, and Dariu M Gavrila. Semantic scene completion using local deep implicit functions on lidar data. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(10):7205–7218, 2021. 3

[47] Christoph B Rist, David Schmidt, Markus Enzweiler, and Dariu M Gavrila. Scssnet: Learning spatially-conditioned scene segmentation on lidar point clouds. In *IEEE Intelligent Vehicles Symposium (IV)*, 2020. 3

[48] Adin Ramirez Rivera, Byungyong Ryu, and Oksam Chae. Content-aware dark image enhancement through channel division. *IEEE Transactions on Image Processing*, 21(9):3967–3980, 2012. 3

[49] Luis Roldao, Raoul De Charette, and Anne Verroust-Blondet. 3d semantic scene completion: a survey. *International Journal of Computer Vision*, 130(8):1978–2005, 2022. 1

[50] David Rolnick, Andreas Veit, Serge Belongie, and Nir Shavit. Deep learning is robust to massive label noise. *arXiv preprint arXiv:1705.10694*, 2017. 3

[51] Nathan Silberman, Derek Hoiem, Pushmeet Kohli, and Rob Fergus. Indoor segmentation and support inference from rgbd images. In *European Conference on Computer Vision (ECCV)*, 2012. 2, 3, 4, 6, 7, 8, 12, 13

[52] Shuran Song, Fisher Yu, Andy Zeng, Angel X Chang, Manolis Savva, and Thomas Funkhouser. Semantic scene completion from a single depth image. In *IEEE International Conference on Computer Vision (ICCV)*, 2017. 1, 2, 3, 6, 7, 8, 14

[53] Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jon Shlens, and Zbigniew Wojna. Rethinking the inception architecture for computer vision. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016. 6

[54] Jiaxiang Tang, Xiaokang Chen, Jingbo Wang, and Gang Zeng. Not all voxels are equal: Semantic scene completion from the point-voxel perspective. In *AAAI Conference on Artificial Intelligence (AAAI)*, 2022. 3

[55] Leonardo Tanzi, Pietro Piazzolla, Francesco Porpiglia, and Enrico Vezzetti. Real-time deep learning semantic segmentation during intra-operative surgery for 3d augmented real-

ity assistance. *International Journal of Computer Assisted Radiology and Surgery*, 16(9):1435–1445, 2021. 1

[56] Xuzhi Wang, Di Lin, and Liang Wan. Ffnet: Frequency fusion network for semantic scene completion. In *AAAI Conference on Artificial Intelligence (AAAI)*, 2022. 2, 4, 7

[57] Yida Wang, David Joseph Tan, Nassir Navab, and Federico Tombari. Forknet: Multi-branch volumetric semantic completion from a single depth image. In *IEEE International Conference on Computer Vision (ICCV)*, 2019. 7

[58] Cihang Xie, Yuxin Wu, Laurens van der Maaten, Alan L Yuille, and Kaiming He. Feature denoising for improving adversarial robustness. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019. 3

[59] Enze Xie, Wenhai Wang, Zhiding Yu, Anima Anandkumar, Jose M Alvarez, and Ping Luo. Segformer: Simple and efficient design for semantic segmentation with transformers. *Advances in Neural Information Processing Systems (NeurIPS)*, 2021. 6, 7, 12

[60] Chao-Han Huck Yang, I Hung, Te Danny, Yi-Chieh Liu, and Pin-Yu Chen. Treatment learning transformer for noisy image classification. *arXiv preprint arXiv:2203.15529*, 2022. 3

[61] Chao-Han Huck Yang, Yi-Chieh Liu, Pin-Yu Chen, Xiaoli Ma, and Yi-Chang James Tsai. When causal intervention meets image masking and adversarial perturbation for deep neural networks. *Computing Research Repository*, abs/1902.03380, 2019. 3

[62] Dong Yang and Jian Sun. Proximal dehaze-net: A prior learning-based deep network for single image dehazing. In *European conference on computer vision (ECCV)*, 2018. 13

[63] Dong Zhang, Hanwang Zhang, Jinhui Tang, Xian-Sheng Hua, and Qianru Sun. Causal intervention for weakly-supervised semantic segmentation. *Advances in Neural Information Processing Systems (NeurIPS)*, 2020. 3

[64] Dong Zhang, Hanwang Zhang, Jinhui Tang, Xian-Sheng Hua, and Qianru Sun. Self-regulation for semantic segmentation. In *IEEE International Conference on Computer Vision (ICCV)*, 2021. 1, 3

[65] Dong Zhang, Hanwang Zhang, Jinhui Tang, Meng Wang, Xiansheng Hua, and Qianru Sun. Feature pyramid transformer. In *European Conference on Computer Vision (ECCV)*, 2020. 2, 3

[66] Pingping Zhang, Wei Liu, Yinjie Lei, Huchuan Lu, and Xiaoyun Yang. Cascaded context pyramid for full-resolution 3d semantic scene completion. In *IEEE International Conference on Computer Vision (ICCV)*, 2019. 2, 4, 7

[67] Zhengyou Zhang. Microsoft kinect sensor and its effect. *IEEE MultiMedia*, 19(2):4–10, 2012. 6

[68] Borui Zhao, Quan Cui, Renjie Song, Yiyu Qiu, and Jiajun Liang. Decoupled knowledge distillation. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022. 3

[69] Chuanjun Zheng, Daming Shi, and Wentian Shi. Adaptive unfolding total variation network for low-light image enhancement. In *IEEE International Conference on Computer Vision (ICCV)*, 2021. 3

[70] Min Zhong and Gang Zeng. Semantic point completion network for 3d semantic scene completion. In *European Conference on Artificial Intelligence (ECAI)*, 2020. 3, 7