This ICCV paper is the Open Access version, provided by the Computer Vision Foundation.
Except for this watermark, it is identical to the accepted version;
the final published version of the proceedings is available on IEEE Xplore.

ForkNet: Multi-branch Volumetric Semantic Completion from a Single Depth Image

Yida Wang[1], David Joseph Tan[2], Nassir Navab[1], Federico Tombari[1,2]
[1]Technische Universität München [2]Google Inc.

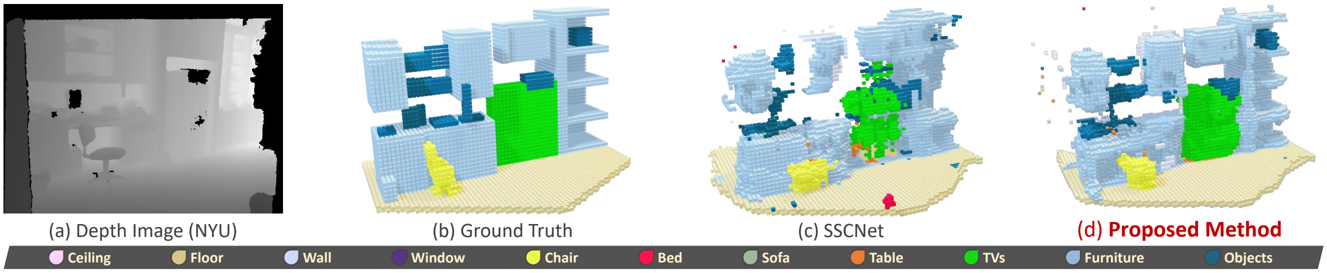


Figure 1: Our 3D semantic completion model (right-most) generates realistic yet accurate volumetric scene representations from a single depth image (left-most) affected by occlusion and noise, even if acquired from a real depth sensor.

Abstract

We propose a novel model for 3D semantic completion from a single depth image, based on a single encoder and three separate generators used to reconstruct different geometric and semantic representations of the original and completed scene, all sharing the same latent space. To transfer information between the geometric and semantic branches of the network, we introduce paths between them concatenating features at corresponding network layers. Motivated by the limited amount of training samples from real scenes, an interesting attribute of our architecture is the capacity to supplement the existing dataset by generating a new training dataset with high quality, realistic scenes that even includes occlusion and real noise. We build the new dataset by sampling the features directly from latent space which generates a pair of partial volumetric surface and completed volumetric semantic surface. Moreover, we utilize multiple discriminators to increase the accuracy and realism of the reconstructions. We demonstrate the benefits of our approach on standard benchmarks for the two most common completion tasks: semantic 3D scene completion and 3D object completion.

1. Introduction

The increasing abundance of depth data, thanks to the widespread presence of depth sensors on devices such as robots and smartphones, has recently fostered big advancements in 3D processing for augmented reality, robotics and scene understanding, unfolding new applications and technology that relies on the geometric rather than just the appearance information. Since 3D devices sense the environment from one specific viewpoint, the geometry that can be captured in one shot is only partial due to occlusion caused by foreground objects as well as self-occlusion from the same object.

As for many applications, this partial 3D information is insufficient to robustly carry-out 3D tasks such as object detection and tracking or scene understanding. A recent research direction has emerged that leverages deep learning to "complete" the depth images acquired by a 3D sensor, *i.e.* filling in the missing geometry that the sensor could not capture due to occlusion. The capability of deep learning to determine a latent space that captures the global context from the training samples proved useful in regressing completed 3D scenes and 3D shapes even when big portion of the geometry are missing [3, 4, 28, 30, 39]. Also, some of these approaches have been extended to jointly learn how to infer geometry and semantic information, in what is referred to as semantic 3D scene completion [4, 30, 34]. Nevertheless, current approaches are still limited by different factors, including the difficulty of regressing fine and sharp details of the completed geometry, as well as to generalize to shapes that significantly differ from those seen during training.

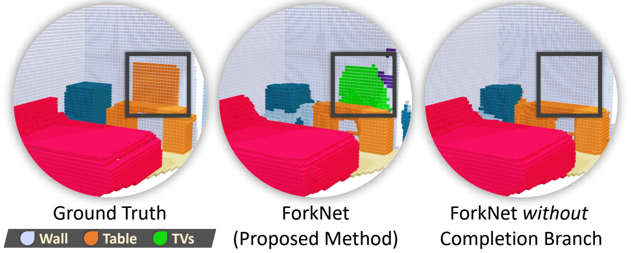


Figure 2: This figure shows the ground truth reconstruction where we notice the incorrect labels from SUNCG [30] dataset on the TVs, *i.e.* enclosed in the black box.

In this work, we aim to tackle 3D completion from a single depth image based on a novel learned model that relies on a single encoder and multiple generators, each trained to regress a different 3D representation of the input data: (*i*) a voxelized depth map, (*ii*) a geometric completed volume, (*iii*) a semantic completed volume. This particular architecture aims at two goals. The first is to supplement the lack of paired input-output data, *i.e.* a depth map and the associated completed volumetric scene, with novel pairs directly generated from the latent space, *i.e.* by means of (*i*) and (*iii*). The second goal is to overcome a common limitation of available benchmarks that provide imprecise semantic labels, by letting the geometric completion remain unaffected from it, *i.e.* by means of (*i*) and (*ii*). By means of specific connections between corresponding neural layers in the different branches, we let the semantic completion model be conditioned on geometric reconstruction information, this being beneficial to generate accurate reconstructions with aligned semantic information.

Overall, the proposed learning model uses a mix of supervised and unsupervised training stages which leverage the power of generative models in addition to the annotations provided by benchmark datasets. Additionally, we propose to further improve the effectiveness of our generative model by employing discriminators able to increase the accuracy and realism of the produced output, yielding completed scenes with high level details even in the presence of strong occlusion, as witnessed by Fig. 1 that reports an example from a real dataset (NYU [24]).

Our contributions can be summarized as follows: (*i*) a novel architecture, dubbed ForkNet, based on a single encoder and three generators built upon the same shared latent space, useful to generate additional paired training samples; (*ii*) the use of specific connections between generators to let geometric information condition and drive the completion process over the often imprecise ground truth annotations (see Fig. 2); and, (*iii*) the use of multiple discriminators to regress fine details and realistic completions. We demonstrate the benefits of our approach on standard benchmarks for the two most common completion tasks: semantic 3D scene completion and 3D object completion. For the former, we rely on SUNCG [30] (synthetic) and NYU [24] (real). For the latter, instead, we test on ShapeNet [1] and 3D-RecGAN [38]. Notably, we outperform the state of the art for both scene reconstruction and object completion on the real dataset.

2. Related work

Semantic scene completion. 3D semantic scene completion starts from a depth image or a point cloud to provide an occlusion-free 3D reconstruction of the visible scene within the viewpoint's frustrum while labeling each 3D element with a semantic class from a pre-defined category set. Scene completion could be in principle achieved by exploiting simple geometric cues such as plane consistency [23] or object symmetry [17]. Moreover, meshing approaches such as Poisson reconstruction [16] as well as purely geometric works [7] can also be employed for this goal.

Recent approaches suggested to leverage deep learning to predict how to fill-in occluded parts in a globally coherent way with respect to the training set. SSCNet [30] carries out semantic scene completion from a single depth image using dilated convolution [40] to capture 3D spatial information at multiple scales. They rely on a volumetric representation to represent both input and output data. Based on SSCNet, VVNet [12] applies view-based 3D convolutions as a replacement for SDF back-projections, this resulting more effective in extracting geometric information from the input depth image. SaTNet [21] relies on the RGB-D images. They initially predict the 2D semantic segments with the RGB. The depth image then back-projects the semantically labelled pixels to a 3D volume which goes through another architecture for 3D scene completion. ScanComplete [4] also targets semantic scene completion but, instead of starting from a single depth image, they assume to process a large-scale reconstruction of a scene acquired via a consumer depth camera. They suggest a coarse-to-fine scheme based on an auto-regressive architecture [27], where each level predicts the completion and the per-voxel semantic labeling at a different voxel resolution. The work in [34] proposes to use GANs for the task of semantic scene completion from a single depth image. In particular, it proposes to use adversarial losses applied on both the output and latent space to enforce realistic interpolation of scene parts. The work in [34] proposes to use GANs for the task of semantic scene completion from a single depth image. In particular, it proposes to use adversarial losses applied on both the output and latent space to enforce realistic interpolation of scene parts. Partially related to this field, the work in [31] leverages input object proposals in the form of 2D bounding boxes to extract the layout of a 3D scene from a single RGB image, while estimating the pose of the objects therein. A similar task is tackled by [9] starting from an

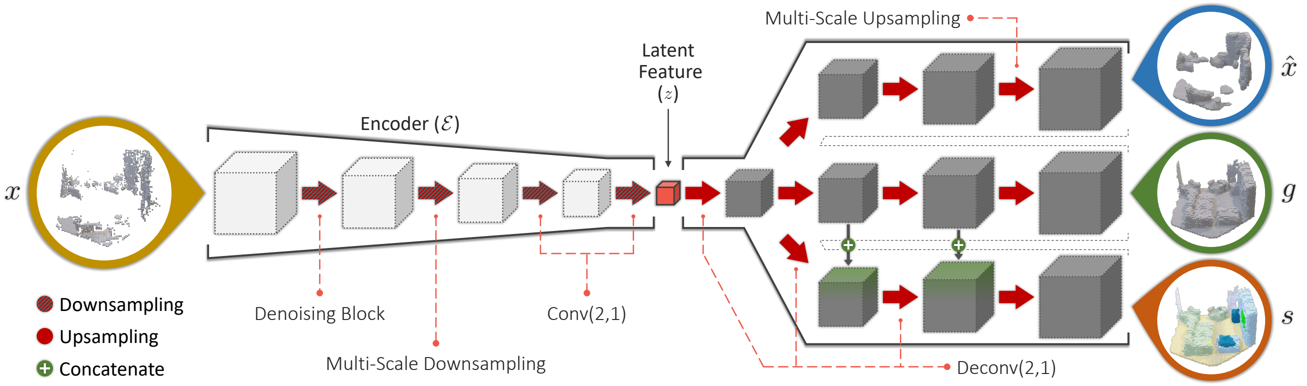


Figure 3: ForkNet – the proposed volumetric network architecture for semantic completion relies on a shared latent space encoded from SDF volume x reconstructed from the input depth image. The two decoding paths are trained to generate, respectively, incomplete surface geometry ($\hat{x}$), completed geometric volume (g) and completed semantic volumes (s).

RGB-D image.

Object completion. 3D object completion aims at obtaining a full 3D object representation from either a single depth or RGB image. While several RGB-based approaches have been recently proposed [2, 6, 35], in this section, we will focus only on those based on depth images as input since they are more related to the scope of this work. The work in [28] uses a hybrid architecture based on a CNN and an autoencoder to learn completing 3D shapes from a single depth map. 3D-RecGAN [38, 39] proposes to complete an observed object from a single depth image using a network based on skip connections [29] between the encoder and the generator so to fetch more spatial information from the input depth image to the generator. 3D-EPN [3] performs shape completion based on a latent feature concatenated with object classification information via one-hot coding, so that this additional semantic information could drive an accurate extrapolation of the missing shape parts. Han *et al*. [13] complete shapes with multiple depth images fused via LSTM Fusion [19] and process the fused data using a 3D fully convolutional approach. MarrNet [35] reconstructs the 3D shape by applying reprojection consistency between 2.5D sketch and 3D shape.

GANs for 3D shapes. Although the use of GANs for 3D semantic scene completion tasks is almost an unexplored territory, GANs have been frequently employed in recent proposals for the task of learning a latent space for 3D shapes, useful for object completion as well as for tasks such as object retrieval and object part segmentation. For instance, 3D-VAE-GAN [36] trains a volumetric GAN in an unsupervised way from a dataset of 3D models, so to be able to generate realistic 3D shapes by sampling the learned latent space. ShapeHD [37] tackles the difficult problem of reconstructing 3D shapes from a single RGB image and suggests to overcome the 2D-3D ambiguity by adversarially learning a regularizer for shapes. PrGAN [8] learns to generate 3D volumes in an unsupervised way, trained by a discriminator that distinguishes whether 2D images projected from a generated 3D volume are realistic or fake. 3D-ED-GAN [33] transforms a coarse 3D shape into a more complete one using a Long Short-term Memory (LSTM) Network by interpreting 3D volumes as sequences of 2D images.

3. Proposed semantic completion

Taking the depth image as input, we reconstruct the visible surface by back-projecting each pixel onto a voxel of the volumetric data. Denoted as x, we represent the surface reconstruction from the depth image as a signed distance function (SDF) [25] with $n_l \times n_w \times n_h$ voxels such that the value of the voxel approaches zero when it is closer to the visible surface.

Our task then is to produce the completed reconstruction of the scene with a semantic label for each voxel. Having N object categories, the class labels are assigned as $\mathcal{C} = \{c_i\}_{i=0}^{N}$ where c_0 is the empty space. Thus, denoted as s, we represent the resulting semantic volume as a one-hot encoding [22] with $N + 1$ dimensional feature. Similarly, we define g as the completed reconstruction of the scene without the semantic information by setting N to 1.

3.1. Model architecture

We assemble an encoder-generator architecture [36] that builds the completed semantic volume from the partial scene derived from a single depth image. As illustrated in Fig. 3, the encoder $\mathcal{E}(\cdot)$ is composed of 3D convolutional operators where the spatial resolutions are decreased by a factor of two in each layer. In effect, this continuously reduces the volume into its simplest form, denoted by the latent feature z such that $z = \mathcal{E}(x)$.

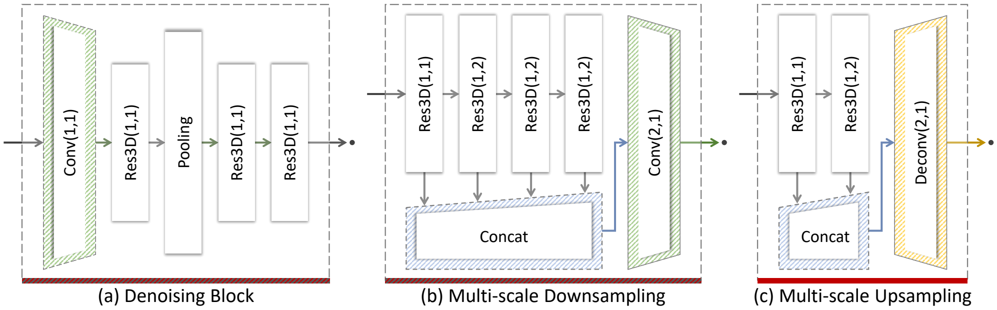


Figure 4: (a-b) Downsampling and (c) upsampling convolutional layers in our architecture (see Fig. 3). Note that the two parameters (s, d) in all the functions are the stride and dilation while the kernel size is set to 3.

In detail, the encoder is composed of four downsampling operators. The first aims at denoising [30] the SDF volumes as illustrated in Fig. 4(a). This involves a combination of a 3D convolutional operator, several 3D ResNet blocks [14], denoted as Res3D(s, d) where s is the stride while d is the dilation, and a pooling layer. The second layer aims at including different objects in the scene even with varying sizes by concatenating the output of four sequentially connected 3D ResNet blocks in Fig. 4(b). Consequently, the information from the smaller objects are captured on the first Res3D$(\cdot, \cdot)$ while the larger object are captured on the subsequent blocks. Notably, the first block is parameterized with a dilation of 1 while the other three with dilations of 2. The concatenated result is then downsampled by a 3D convolutional operator. In the final two layers, we further downsample the volume with 3D convolutional operators until we form the latent feature with a size of $16 \times 5 \times 3 \times 5$.

Branching from the same latent feature, we design three generators that reconstructs:

(i) the *SDF volume* $(\hat{x})$ which, with respect to x, formulates as an autoencoder;

(ii) the *completed volume* (g) which focuses on reconstructing the geometric structure of the scene; and,

(iii) the *completed semantic volume* (s) which is the desired outcome.

We assign these generators as the functions $\mathcal{G}_{\hat{x}}(\cdot)$, $\mathcal{G}_g(\cdot)$ and $\mathcal{G}_s(\cdot)$, respectively. Notably, we distinguish x, which is the SDF volume obtained from the input depth image, from $\hat{x}$, which is the inferred SDF volume obtained from the generator. The structure of each generator is composed of 3D deconvolutional operators that increases the spatial resolution by two in each layer.

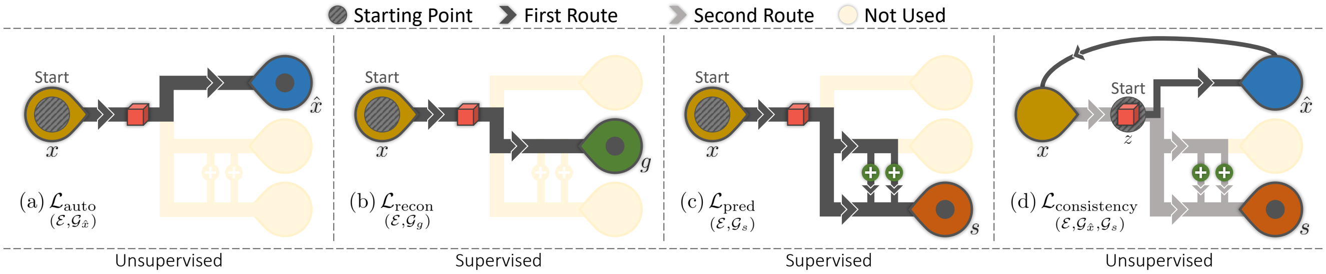
While the first 3 convolutional upsampling layers in the generators are composed of 3D deconvolutional operators as shown in Fig. 3, the last layer is a multi-scale upsampling which is sketched in Fig. 4(c). This layer is similar to the multi-scale downsampling of the encoder where the goal is to consider the variation of sizes from different objects. In this case, we concatenate the results of two sequentially connected 3D ResNet blocks then end with a 3D deconvolution operator. With the same operations as the other generators, the generator that builds the completed semantical volume $\mathcal{G}_s$ additionally incorporates the data from the generator of the geometric scene reconstruction $\mathcal{G}_g$ as shown in Fig. 3 by concatenating the results from the second and the third layers. Since the resulting $\hat{x}$, g and s have different number of channels, only the dimension of the output from the deconvolutional operator in the last layer changes for each structure.

Giving a holistic perspective, we can simplify the sketch of the architecture in Fig. 3 to Fig. 5 by plotting the relation of the variables x, $\hat{x}$, g, s and z. When we focus on certain structures, we notice that we have an autoencoder that builds an SDF volume in Fig. 5(a), the reconstruction of the scene in Fig. 5(b) and the volumetric semantic completion in Fig. 5(c), where all of these structures branch out from the same latent feature. Later in Sec. 3.2, these plots are used to explain the loss terms in training.

The rationale of having multiple generators is twofold. First, in contrast to the typical encoder-decoder architecture, we introduce the connection that relates the two generators. Taking the output from the $\mathcal{G}_{\hat{x}}$ in each layer, we concatenate the results to the data from $\mathcal{G}_s$ as shown in Fig. 3. By establishing this relation, we incorporate the SDF reconstruction from the $\mathcal{G}_{\hat{x}}$ into the semantic completion in order to capture the geometric information of the observed scene.

Second, the latent feature can generate a pair of SDF and completed semantic volumes. Through this set of paired volumes, we can supplement the learning dataset in an unsupervised manner. This becomes a significant component in evaluating the NYU dataset [24] in Sec. 4.1 where the amount of learning dataset is limited because, since they use a consumer depth camera to capture real scenes, annotation becomes difficult. However, evaluating on this dataset is more essential compared to the synthetic dataset because it brings us a step closer to real applications. Relying on this idea in Sec. 3.2, we propose an unsupervised loss term that optimizes the entire architecture based on its own learning dataset.

Discriminators. Inspired by GANs [11, 26], we introduce the discriminator $\mathcal{D}_x$ that evaluates whether the generated SDF volumes from $\mathcal{G}_{\hat{x}}$ are realistic or not by comparing them to the learning dataset. Here, $\mathcal{D}_x$ is constructed by a



▨ Starting Point
First Route
Second Route
Not Used
Start
x
$\hat{x}$
(a) $\mathcal{L}_{\text{auto}}$ $(\mathcal{E},\mathcal{G}_{\hat{x}})$
Unsupervised
Start
x
g
(b) $\mathcal{L}_{\text{recon}}$ $(\mathcal{E},\mathcal{G}_g)$
Supervised
Start
x
s
(c) $\mathcal{L}_{\text{pred}}$ $(\mathcal{E},\mathcal{G}_s)$
Supervised
Start
x
z
$\hat{x}$
s
(d) $\mathcal{L}_{\text{consistency}}$ $(\mathcal{E},\mathcal{G}_{\hat{x}},\mathcal{G}_s)$
Unsupervised

Figure 5: Graphical models of the 4 data flows (and the associated loss terms) used during training and derived from Fig. 3.

sequentially connected 3D convolutional operators with the kernel size of 3×3×3 and stride of 2. This implies that the resolution of the input volume is sequentially decreased by a factor of two after each operation. To capture the local information of the volume [5], the results from $\mathcal{D}_x$ is set to a resolution of 5×3×5.

With a similar architecture as $\mathcal{D}_x$, we also introduce a second discriminator $\mathcal{D}_s$ that evaluates the authenticity of the generated volume s. Notably, the two discriminators are evaluated in the loss terms in Sec. 3.2 to optimize the generators.

3.2. Loss terms

Leveraging on the forward passes of smaller architectures in Fig. 5, we can optimize the entire architecture by simultaneously optimizing different paths. We also optimize the architecture of the two discriminators that distinguishes whether the generated volumes are realistic or not. During training, the learning dataset is given by a set of the pairs $\{(x, s_{\text{gt}})\}$, where we distinguish s_{gt} as the ground truth from the generated s. Note that the ground truth for the geometric completion g_{gt} is the binarized summation of non-empty space in s_{gt} and an occupancy volume from the SDF surface.

SDF autoencoder. Motivated to reconstruct as similar SDF volume from the generator $\mathcal{G}_{\hat{x}}$ as the original input, we define the loss function

$$\underset{(\mathcal{E},\mathcal{G}_{\hat{x}})}{\mathcal{L}_{\text{auto}}} = \|\mathcal{G}_{\hat{x}}(\mathcal{E}(x)) - x\|^2 \tag{1}$$

for the autoencoder in Fig. 5(a) in order to minimize the difference between the observed x and the inferred $\hat{x}$.

Geometric completion. In Fig. 5(b), a conditional generative model combines the encoder $\mathcal{E}(\cdot)$ and the generator $\mathcal{G}_g(\cdot)$ in order to reconstruct the scene (*i.e.* without the semantic labels). Since the reconstruction is a two-channel volume that represents the empty and non-empty category, we use a binary cross-entropy loss

$$\underset{(\mathcal{E},\mathcal{G}_g)}{\mathcal{L}_{\text{recon}}} = \sum_{i=0}^{1} (\epsilon(\mathcal{G}_g(\mathcal{E}(x)), g_{\text{gt}})) \tag{2}$$

to train the inference network, where $\epsilon(\cdot,\cdot)$ is the per-category error

$$\epsilon(q, r) = -\lambda r \log q - (1-\lambda)(1-r)\log(1-q) \,. \tag{3}$$

In (3), λ, which ranges from 0 to 1, weighs the importance of reconstructing true positive regions in the volume. If $\lambda = 1$, the penalty for the false positive predictions will not be considered; while, if λ is set to 0, the false negatives will not be corrected.

Semantic completion. Similar to (2), in Fig. 5(c), we train a conditional generative model that is composed of the encoder $\mathcal{E}(\cdot)$ and generator $\mathcal{G}_s(\cdot)$ linking x and s. Hence, we also use a binary cross-entropy loss

$$\underset{(\mathcal{E},\mathcal{G}_s)}{\mathcal{L}_{\text{pred}}} = \sum_{i=0}^{N} (\epsilon(\mathcal{G}_s(\mathcal{E}(x)), s_{\text{gt}})) \tag{4}$$

where N is the number of categories in the semantic scene.

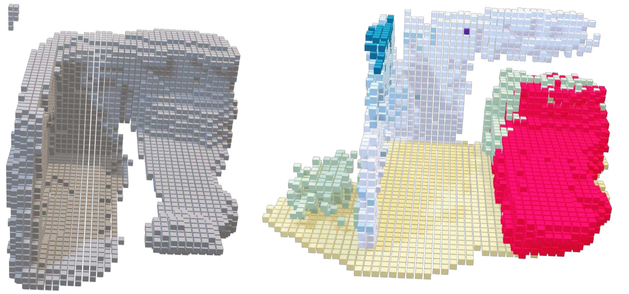
Discriminators on the architecture. In relation to the architecture, we use two discriminators to optimize the generators [36] through

$$\begin{aligned}\underset{\mathcal{G}_{\hat{x}}}{\mathcal{L}_{\text{gen-}\hat{x}}} &= -\log\left(\mathcal{D}_x(\mathcal{G}_{\hat{x}}(z))\right)\\ \underset{\mathcal{G}_s}{\mathcal{L}_{\text{gen-}s}} &= -\log\left(\mathcal{D}_s(\mathcal{G}_s(z))\right) .\end{aligned} \tag{5}$$

In this manner, we optimize the two generative models including both the SDF encoder and the semantic scene generator by randomly sampling the latent features. On the other hand, when we update the parameters of both discriminators, we optimize the loss functions

$$\begin{aligned}\underset{(\mathcal{D}_x)}{\mathcal{L}_{\text{dis-}x}} &= -\log(\mathcal{D}_x(x)) - \log\left(1 - \mathcal{D}_x(\mathcal{G}_{\hat{x}}(z))\right)\\ \underset{(\mathcal{D}_s)}{\mathcal{L}_{\text{dis-}s}} &= -\log(\mathcal{D}_s(s_{\text{gt}})) - \log\left(1 - \mathcal{D}_s(\mathcal{G}_s(z))\right) .\end{aligned} \tag{6}$$

During training, we apply the set of equations in (5) and (6) alternatingly to optimize the generators and the discriminators separately. Note that we use the KL-divergence from the variational inference [10, 15] to penalize the deviation



(a) SDF Volume

(b) Completed Semantic Scene

Figure 6: An example of the generated SDF volume and the corresponding completed semantic scene parameterized from the latent feature, which are used to supplement the existing learning dataset.

between the distribution of $\mathcal{E}(x)$ and a normal distribution with zero mean and identity variance matrix. The advantage of such is the capacity to easily sample from the latent space in the generative model, which becomes helpful in the succeeding loss term.

SDF-Semantic consistency. Since the generators are trained to produce SDF volumes and semantic scenes while being optimized to produce realistic data by the discriminator, we can build a new set of paired volumes to act as the learning dataset in order to supplement the existing one. Thus, we propose to generate paired volumes directly from the latent feature in order to optimize the architecture in an unsupervised learning.

Exploiting the latent space, we reconstruct the set of pairs $\{(\mathcal{G}_{\hat{x}}(z), \mathcal{G}_s(z))\}$, where z is randomly sampled from a Gaussian distribution centered on the average of latent features of a batch of samples. Following the inference model in Fig. 5(c), we formulate a similar loss function as (4) but with the newly acquired data such that

$$\mathcal{L}_{\substack{\text{consistency}\\(\mathcal{E},\mathcal{G}_{\hat{x}},\mathcal{G}_s)}} = \sum_{i=0}^{N} (\epsilon(\mathcal{G}_s(\mathcal{E}(\mathcal{G}_{\hat{x}}(z))), \mathcal{G}_s(z))) \ . \tag{7}$$

By drawing the data flow of the first term $\mathcal{G}_s(\mathcal{E}(\mathcal{G}_{\hat{x}}(z)))$ in Fig. 5(d), we observe that the loss term in (7) optimizes the entire architecture.

Interestingly, when we take a closer look at the newly generated pairs $\{(\mathcal{G}_{\hat{x}}(z), \mathcal{G}_s(z))\}$ in Fig. 6, we can easily notice the realistic results. The SDF volume in Fig. 6(a) considers missing regions due to the camera position while the semantic scene in Fig. 6(b) generates lifelike structures and reasonable positions of the objects in the scene (*e.g.* the bed in red). By adding the newly generated pairs, we numerically show in Sec. 4.1 that there is a significant boost in performance when evaluating the NYU dataset [24] where the size of the learning dataset is small.

Optimization. With all the loss terms given, achieving the optimum parameters in our architecture requires us to simultaneously minimize them. We start by optimizing (1), (2), (4) and (5) altogether. Then, the loss functions in (6) for the two discriminators are optimized alternatively (*i.e.* batch-by-batch) with (1), (2), (4) and (5). In practice, we employ the Adam optimizer [18] with a learning rate of 0.0001. For the data flows, Fig. 5(a) and (d) are both unsupervised while Fig. 5(b) and (c) are supervised. In addition, for the discriminators, (5) is unsupervised while (6) is supervised.

4. Experiments

There are two tasks at hand – (1) 3D semantic scene completion; and, (2) 3D object completion. Although they perform similar tasks in reconstructing from a single view, the former completes the structure of a scene with semantic labels while the latter requires a more detailed completion with the assumption of a single category.

Metric. For each of the N classes, the accuracy of the predicted volumes is measured based on the Intersection over Union (IoU). Analogously to the evaluation carried out by other methods, the average IoU is taken from all the categories except for the empty space.

Implementation details. We learn our model with an Nvidia Titan Xp with a batch size of 8. We applied batch normalization after every convolutional and deconvolutional operations except for the convolutional operations in the last deconvolutional layers in 3 generators. Leaky ReLU with a negative slope of 0.2 is applied on the output of each convolutional layer in the Res3D$(\cdot,\cdot)$ modules in Fig. 4. In addition, ReLU is applied on the output of deconvolutional operations in the generators except for the last deconvolution operation in the Multi-Scale Upsampling. Finally, the sigmoid operation is applied to the last deconvolution layer of the generators for the geometric and semantic completion. Notably, the factor λ from (3) is set to be 0.5 for the geometric completion in (2). For the semantic completion, it is initially set to 0.9 in (4). However, when the network is capable of revealing objects from the depth image, more and more false positive predictions in the empty space appears. Due to this, we set λ to 0.6 after five epochs.

4.1. Semantic scene completion

The SUNCG [30] and NYU [24] datasets are currently the most relevant benchmarks for semantic scene completion, and include a paired depth image and the corresponding semantically labeled volume. While SUNCG comprises synthetically rendered depth data, NYU includes real scenes acquired with a Kinect depth sensor. This makes the evaluation of NYU more challenging, due to the presence of real nuisances, as well as due to a limited training set of

	ceil.	floor	wall	win.	chair	bed	sofa	table	tvs	furn.	objs.	*Avg.*
SSCNet [30] (observed)	97.7	94.5	66.4	30.0	**36.9**	60.2	62.5	**56.3**	12.1	46.7	**33.0**	54.2
Proposed Method (observed)	**98.2**	**96.9**	**67.8**	**37.4**	35.9	**72.9**	**69.6**	48.8	**20.5**	**48.4**	32.4	**57.2**
Wang *et al.* [34]	41.4	37.7	45.8	26.5	26.4	21.8	25.4	23.7	20.1	16.2	5.7	26.4
3D-RecGAN [38]	79.9	75.2	48.2	28.9	20.2	64.4	54.6	25.7	17.4	33.7	24.4	43.0
SSCNet [30]	96.3	84.9	56.8	28.2	21.3	56.0	52.7	33.7	10.9	44.3	25.4	46.4
VVNet [12]	**98.4**	**87.0**	61.0	54.8	**49.3**	**83.0**	75.5	**55.1**	43.5	**68.8**	**57.7**	**66.7**
SaTNet [21]	97.9	82.5	57.7	**58.5**	45.1	78.4	72.3	47.3	**45.7**	67.1	55.2	64.3
Proposed Method	95.0	85.9	**73.2**	54.5	46.0	81.3	74.2	42.8	31.9	63.1	49.3	63.4
– *without completion branch*	94.1	83.5	68.2	49.6	43.1	80.5	**77.7**	41.8	33.8	61.7	51.7	62.3
– *without scene consistency*	89.6	79.5	63.4	46.3	39.0	77.5	73.2	37.7	29.8	57.4	46.7	58.2

Table 1: Semantic scene completion results on the SUNCG test set with depth map for IoU (in %).

	ceil.	floor	wall	win.	chair	bed	sofa	table	tvs	furn.	objs.	*Avg.*
SSCNet [30] (observed)	37.7	**91.9**	**75.4**	64.0	29.0	51.1	63.3	43.7	**29.7**	73.3	54.5	50.8
Proposed Method (observed)	**41.5**	90.8	69.6	**54.8**	**27.7**	**53.1**	**66.3**	**44.4**	27.1	**74.7**	**57.5**	**55.2**
Lin *et al.* [20] (NYU only)	0.0	11.7	13.3	14.1	9.4	29.0	24.0	6.0	7.0	16.2	1.1	12.0
3D-RecGAN [38]	35.3	70.3	24.1	3.8	11.9	47.4	43.1	11.4	16.9	30.6	7.2	27.5
Geiger and Wang [9] (NYU only)	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 [30]	15.1	94.6	24.7	10.8	17.3	53.2	45.9	15.9	13.9	31.1	12.6	30.5
VVNet [12]	19.3	**94.8**	28.0	12.2	**19.6**	57.0	50.5	17.6	11.9	35.6	15.3	32.9
SaTNet [21]	17.3	92.1	28.0	16.6	19.3	57.5	**53.8**	17.7	18.5	38.4	18.9	34.4
Proposed Method	36.2	93.8	**29.2**	18.9	17.7	**61.6**	52.9	**23.3**	**19.5**	45.4	**20.0**	**37.1**
– *without completion branch*	35.8	94.1	28.9	**19.2**	16.8	61.4	53.5	23.0	14.0	**45.6**	18.9	36.5
– *without scene consistency*	**36.8**	91.7	28.0	18.3	8.3	58.8	49.5	13.0	16.7	42.6	17.6	34.7

Table 2: Semantic scene completion results on the NYU test set with depth map for IoU (in %).

	SUNCG	NYU
Lin *et al.* [20]	–	36.4
3D-RecGAN [38]	72.1	51.3
Geiger and Wang [9]	–	44.4
SSCNet [30]	73.5	56.6
VVNet [12]	84.0	61.1
SaTNet [21]	78.5	60.6
Proposed Method	**86.9**	**63.4**
– *without completion branch*	82.3	62.6
– *without scene consistency*	82.0	61.1

Table 3: Scene completion results on the SUNCG and the NYU test set in terms of IoU (in %).

less than 1000 samples. We compare our method against Wang *et al.* [34], Lin *et al.* [20], 3D-RecGAN [38], Geiger and Wang [9], SSCNet [30], VVNet [12], and SaTNet [21]. The resolution of our input volume is given in the scale of 80×48×80 voxels. While [9, 12, 20, 21, 30] produce 60×36×60 semantic volumes for evaluation, [34, 38] and us produce a slightly higher resolution of 80×48×80.

Following SUNCG [30], the semantic categories include 12 classes of varying shapes and sizes, *i.e.*: *empty space*, *ceiling*, *floor*, *wall*, *window*, *chair*, *bed*, *sofa*, *table*, *tvs*, *furniture* and *other objects*. We follow two types of evaluation as introduced by [30]. One evaluates the semantic segmentation accuracy on the observed surface reconstruction, while the other considers the semantic segmentation of the predicted full volumetric reconstruction.

SUNCG dataset. Based on an online interior design platform, the evaluation of SUNCG contains more than 130,000 paired depth images and voxel-wise semantic labels taken from 45,622 houses with realistic rooms and furniture layouts [30]. Focusing on the semantic segmentation on the observed surface, our approach performs at an IoU of 57.2% which is 3.0% higher than SSCNet [30]. On the other hand, when we evaluate the IoU measure on the entire volume in Table 1, our method reaches an average IoU of 63.4% which is significantly better than Wang *et al.* [34], 3D-RecGAN [38] and SSCNet [30] but slightly worse than VVNet [12] and SaTNet [21].

NYU dataset (real). The NYU dataset [24] is composed of 1,449 indoor depth images captured with a Kinect depth sensor. Like SUNCG, each image is also annotated with

	bench	chair	couch	table	*Avg.*
Varley *et al.* [32]	65.3	61.9	81.8	67.8	69.2
3D-EPN [3]	75.8	73.9	83.4	77.2	77.6
Han *et al.* [13]	54.4	46.9	48.3	56.0	51.4
3D-RecAE [38]	73.3	73.6	83.2	75.0	76.3
3D-RecGAN [38]	74.5	74.1	84.4	77.0	77.5
Proposed Method	**79.1**	**80.6**	**92.4**	**84.0**	**84.1**
– without scene consistency	76.3	76.4	87.5	81.2	80.4

Table 4: Object completion results on the ShapeNet test set in terms of IoU (in %). The resolution for Varley *et al.* [32] and 3D-EPN [3]: 32×32×32, for others: 64×64×64.

3D semantic labels. Due to its size, training our network on this dataset alone is insufficient. As a solution already used in [30], we take the network trained on the SUNCG then refine it by supplementing the training data from NYU with 1,500 randomly selected samples from SUNCG in each epoch of training.

Although we achieved slightly worse results than VVNet [12] and SaTNet [21] on the synthetic dataset, we performed better than the state of the art on the real images, reaching an IoU measure of 37.1% as shown in Table 2. Consequently, we attain a 4.2% improvement compared to VVNet [12] and 2.7% to SaTNet [21].

Looking at the other approaches, we achieve even more significant improvements with at least 6.6% increase in IoU. For the evaluation on the semantic labels on the observed surface, we gained 4.4% increase in IoU against SSCNet. Notably, our approach outperforms other works not only on the average IoU but also on individual object categories. In addition, we also achieve similar improvements in the scene completion task in Table 3 with approximately 2.8% better in IoU compared to SaTNet [21].

Moreover, while the re-implementation SSCNet [30] in our experiments does not fit into any of our contributions, we used it in order to qualitatively compare our results with them (see Fig. 1).

Ablation study for loss terms. In Tables 1 and 2, we investigate the contribution of $\mathcal{L}_{\text{recon}}$ from the supervised learning and $\mathcal{L}_{\text{consistency}}$ from the unsupervised learning. Our ablation study indicates that $\mathcal{L}_{\text{consistency}}$ prompts the highest boost in IoU with 5.2% in Table 1. When using the $\mathcal{L}_{\text{recon}}$ in the geometric completion, it improves by 1.1% on the SUNCG dataset. A similar conclusion for the loss terms is presented in Table 1 for NYU.

4.2. 3D object completion

Adapting the assessment data and strategy from 3D-RecGAN [38], we use ShapeNet [1] to generate the training and test data for 3D object completion, wherein each reconstructed object surface x is paired with a corresponding ground truth voxelized shape with a size of 64×64×64. The dataset comprises four object classes: *bench*, *chair*, *couch* and *table*. [38] prepared an evaluation for both synthetic and real input data. Notably, for both synthetic and real test data, we can express the same conclusions as the ablation studies in Sec. 4.1 (see Tables 4 and 5).

	bench	chair	couch	table	*Avg.*
Han *et al.* [13]	18.4	14.8	10.1	12.6	14.0
3D-RecAE [38]	23.1	17.8	10.7	14.8	16.6
3D-RecGAN [38]	23.0	17.4	10.9	14.6	16.5
Proposed Method	**32.7**	**24.1**	**15.9**	**22.5**	**23.8**
– without scene consistency	26.1	21.5	14.9	18.6	20.3

Table 5: Object completion results on the real-world test set provided by 3D-RecGAN [38] in terms of IoU (in %). The resolution for all methods is 64×64×64.

Synthetic test data. We perform two evaluations in Table 4. The first is a single category test [38] such that each category is trained and tested separately while the second considers the categories in order to label the voxels. We compare our results against [3, 13, 32, 38].

In the single category test, we achieve the best results with 84.1%. This result is 6.5% higher than 3D-EPN [3], 6.6% higher than 3D-RecGAN [38], 7.8% higher than 3D-RecAE [38], 32.7% higher than Han *et al.* [13] and 14.9% higher than Varley *et al.* [32]. Moreover, this table also shows the we achieve the best results across all categories.

Real test data. Using the single category test in Table 5, we also evaluate the 3D object completion task on the real world test data provided by [38]. In this evaluation, we generate the state-of-the art results with 23.8% IoU measure, which is higher than 3D-RecAE [38] by 7.2%, 3D-RecGAN [38] by 7.3% and Han *et al.* [13] by 9.8%.

5. Conclusion

We propose ForkNet, a novel architecture for volumetric semantic 3D completion that leverages a shared embedding encoding both geometric and semantic surface cues, as well as multiple generators designed to deal with limited paired data and imprecise semantic annotations. Experimental results numerically demonstrate the benefits of our approach for the two tasks of scene and object completion, as well as the effectiveness of the proposed contributions in terms of architecture, loss terms and use of discriminators. However, since we compress the input SDF volume into a lower resolution through the encoder then increase the resolution through the generator, small or thin structures such as the legs of the chair or TVs tend to disappear during compression. This is an aspect we plan to improve in the future work. In addition, for 3D scene understanding, the volumetric representations are typically memory and power-hungry, we also plan to extend our model for completion of efficient and sparse representations such as point clouds.

References

[1] Angel X Chang, Thomas Funkhouser, Leonidas Guibas, Pat Hanrahan, Qixing Huang, Zimo Li, Silvio Savarese, Manolis Savva, Shuran Song, Hao Su, et al. Shapenet: An information-rich 3d model repository. *arXiv preprint arXiv:1512.03012*, 2015. 2, 8

[2] Christopher B Choy, Danfei Xu, JunYoung Gwak, Kevin Chen, and Silvio Savarese. 3d-r2n2: A unified approach for single and multi-view 3d object reconstruction. In *European conference on computer vision (ECCV)*. Springer, 2016. 3

[3] Angela Dai, Charles Ruizhongtai Qi, and Matthias Nießner. Shape completion using 3d-encoder-predictor cnns and shape synthesis. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, volume 3, 2017. 1, 3, 8

[4] Angela Dai, Daniel Ritchie, Martin Bokeloh, Scott Reed, Jürgen Sturm, and Matthias Nießner. Scancomplete: Large-scale scene completion and semantic segmentation for 3d scans. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, volume 1, 2018. 1, 2

[5] Ugur Demir and Gözde B. Ünal. Patch-based image inpainting with generative adversarial networks. *CoRR*, abs/1803.07422, 2018. 5

[6] Haoqiang Fan, Hao Su, and Leonidas J Guibas. A point set generation network for 3d object reconstruction from a single image. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, volume 2, 2017. 3

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

[8] Matheus Gadelha, Subhransu Maji, and Rui Wang. 3d shape induction from 2d views of multiple objects. In *International Conference on 3D Vision (3DV)*. IEEE, 2017. 3

[9] Andreas Geiger and Chaohui Wang. Joint 3d object and layout inference from a single rgb-d image. In *German Conference on Pattern Recognition (GCPR)*. Springer, 2015. 2, 7

[10] Zoubin Ghahramani and Matthew J Beal. Variational inference for bayesian mixtures of factor analysers. In *Advances in Neural Information Processing Systems (NIPS)*, 2000. 5

[11] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *Advances in Neural Information Processing Systems (NIPS)*, 2014. 4

[12] Yuxiao Guo and Xin Tong. View-volume network for semantic scene completion from a single depth image. In *Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI)*. AAAI Press, 2018. 2, 7, 8

[13] Xiaoguang Han, Zhen Li, Haibin Huang, Evangelos Kalogerakis, and Yizhou Yu. High-resolution shape completion using deep neural networks for global structure and local geometry inference. In *Proceedings of IEEE International Conference on Computer Vision (ICCV)*, 2017. 3, 8

[14] Kensho Hara, Hirokatsu Kataoka, and Yutaka Satoh. Can spatiotemporal 3d cnns retrace the history of 2d cnns and imagenet? In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, pages 6546–6555, 2018. 4

[15] Matthew D Hoffman, David M Blei, Chong Wang, and John Paisley. Stochastic variational inference. *The Journal of Machine Learning Research*, 14(1):1303–1347, 2013. 5

[16] Michael Kazhdan and Hugues Hoppe. Screened poisson surface reconstruction. *ACM Transactions on Graphics (ToG)*, 32(3):29, 2013. 2

[17] Young Min Kim, Niloy J Mitra, Dong-Ming Yan, and Leonidas Guibas. Acquiring 3d indoor environments with variability and repetition. *ACM Transactions on Graphics (TOG)*, 31(6):138, 2012. 2

[18] Diederick P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *International Conference on Learning Representations (ICLR)*, 2015. 6

[19] Zhen Li, Yukang Gan, Xiaodan Liang, Yizhou Yu, Hui Cheng, and Liang Lin. Lstm-cf: Unifying context modeling and fusion with lstms for rgb-d scene labeling. In *European Conference on Computer Vision (ECCV)*. Springer, 2016. 3

[20] Dahua Lin, Sanja Fidler, and Raquel Urtasun. Holistic scene understanding for 3d object detection with rgbd cameras. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2013. 7

[21] Shice Liu, YU HU, Yiming Zeng, Qiankun Tang, Beibei Jin, Yinhe Han, and Xiaowei Li. See and think: Disentangling semantic scene completion. In S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett, editors, *Advances in Neural Information Processing Systems 31*, pages 263–274. Curran Associates, Inc., 2018. 2, 7, 8

[22] Pauline Luc, Camille Couprie, Soumith Chintala, and Jakob Verbeek. Semantic segmentation using adversarial networks. *CoRR*, abs/1611.08408, 2016. 3

[23] Aron Monszpart, Nicolas Mellado, Gabriel J Brostow, and Niloy J Mitra. Rapter: rebuilding man-made scenes with regular arrangements of planes. *ACM Trans. Graph.*, 34(4):103–1, 2015. 2

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

[25] Stanley Osher and Ronald Fedkiw. *Level set methods and dynamic implicit surfaces*, volume 153. Springer Science & Business Media, 2006. 3

[26] Alec Radford, Luke Metz, and Soumith Chintala. Unsupervised representation learning with deep convolutional generative adversarial networks. *arXiv preprint arXiv:1511.06434*, 2015. 4

[27] Scott Reed, Aäron van den Oord, Nal Kalchbrenner, Sergio Gómez Colmenarejo, Ziyu Wang, Yutian Chen, Dan Belov, and Nando de Freitas. Parallel multiscale autoregressive density estimation. In *Proceedings of the 34th International Conference on Machine Learning (ICML)*. JMLR. org, 2017. 2

[28] Dario Rethage, Federico Tombari, Felix Achilles, and Nassir Navab. Deep learned full-3d object completion from single view. *CVPR '16 Scene Understanding Workshop (SUNw)*, 2016. 1, 3

[29] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmen-

tation. In *International Conference on Medical image computing and computer-assisted intervention*. Springer, 2015. 3

[30] Shuran Song, Fisher Yu, Andy Zeng, Angel X Chang, Manolis Savva, and Thomas Funkhouser. Semantic scene completion from a single depth image. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2017. 1, 2, 4, 6, 7, 8

[31] Shubham Tulsiani, Saurabh Gupta, David Fouhey, Alexei A Efros, and Jitendra Malik. Factoring shape, pose, and layout from the 2d image of a 3d scene. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2018. 2

[32] Jacob Varley, Chad DeChant, Adam Richardson, Joaquín Ruales, and Peter Allen. Shape completion enabled robotic grasping. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2017. 8

[33] W. Wang, Q. Huang, S. You, C. Yang, and U. Neumann. Shape inpainting using 3d generative adversarial network and recurrent convolutional networks. In *IEEE International Conference on Computer Vision (ICCV)*, Oct 2017. 3

[34] Y. Wang, D. J. Tan, N. Navab, and F. Tombari. Adversarial semantic scene completion from a single depth image. In *2018 International Conference on 3D Vision (3DV)*, 2018. 1, 2, 7

[35] Jiajun Wu, Yifan Wang, Tianfan Xue, Xingyuan Sun, Bill Freeman, and Josh Tenenbaum. Marrnet: 3d shape reconstruction via 2.5 d sketches. In *Advances in neural information processing systems (NIPS)*, 2017. 3

[36] Jiajun Wu, Chengkai Zhang, Tianfan Xue, Bill Freeman, and Josh Tenenbaum. Learning a probabilistic latent space of object shapes via 3d generative-adversarial modeling. In *Advances in Neural Information Processing Systems (NIPS)*, 2016. 3, 5

[37] Jiajun Wu, Chengkai Zhang, Xiuming Zhang, Zhoutong Zhang, William T Freeman, and Joshua B Tenenbaum. Learning 3D Shape Priors for Shape Completion and Reconstruction. In *European Conference on Computer Vision (ECCV)*, 2018. 3

[38] Bo Yang, Stefano Rosa, Andrew Markham, Niki Trigoni, and Hongkai Wen. Dense 3d object reconstruction from a single depth view. *IEEE transactions on pattern analysis and machine intelligence*, 2018. 2, 3, 7, 8

[39] Bo Yang, Hongkai Wen, Sen Wang, Ronald Clark, Andrew Markham, and Niki Trigoni. 3d object reconstruction from a single depth view with adversarial learning. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 2017. 1, 3

[40] Fisher Yu and Vladlen Koltun. Multi-scale context aggregation by dilated convolutions. *arXiv preprint arXiv:1511.07122*, 2015. 2