3D Object Reconstruction from a Single Depth View with Adversarial Learning

Bo Yang
University of Oxford
bo.yang@cs.ox.ac.uk

Hongkai Wen
University of Warwick
hongkai.wen@dcs.warwick.ac.uk

Sen Wang
Heriot-Watt University
s.wang@hw.ac.uk

Ronald Clark
Imperial College London
ronald.clark@imperial.ac.uk

Andrew Markham
University of Oxford
andrew.markham@cs.ox.ac.uk

Niki Trigoni
University of Oxford
niki.trigoni@cs.ox.ac.uk

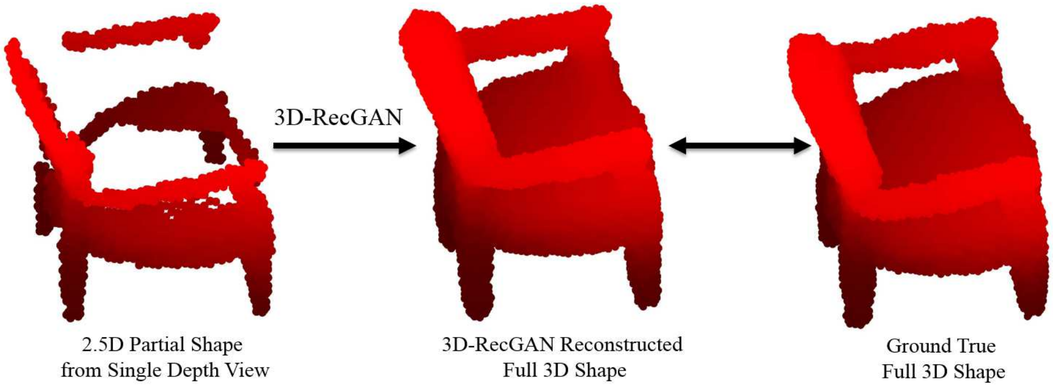

Figure 1. Our method 3D-RecGAN reconstructs a full 3D shape from a single 2.5D depth view.

Abstract

*In this paper, we propose a novel **3D-RecGAN** approach, which reconstructs the complete 3D structure of a given object from a single arbitrary depth view using generative adversarial networks. Unlike the existing work which typically requires multiple views of the same object or class labels to recover the full 3D geometry, the proposed 3D-RecGAN only takes the voxel grid representation of a depth view of the object as input, and is able to generate the complete 3D occupancy grid by filling in the occluded/missing regions. The key idea is to combine the generative capabilities of autoencoders and the conditional Generative Adversarial Networks (GAN) framework, to infer accurate and fine-grained 3D structures of objects in high-dimensional voxel space. Extensive experiments on large synthetic datasets show that the proposed 3D-RecGAN significantly outperforms the state of the art in single view 3D object reconstruction, and is able to reconstruct unseen types of objects. Our code and data are available at: `https://github.com/Yang7879/3D-RecGAN`.*

1. Introduction

The ability to reconstruct the complete and accurate 3D geometry of an object is essential for a broad spectrum of scenarios, from AR/VR applications [46] and semantic understanding, to robot grasping [58] and obstacle avoidance. One class of popular approaches is to use the off-the-shelf low-cost depth sensing devices such as Kinect and RealSense cameras to recover the 3D model of an object from captured depth images. Most of those approaches typically sample multiple depth images from different views of the object to create the complete 3D structure [37] [39] [53]. However, in practice it is not always feasible to scan all surfaces of the object, which leads to incomplete models with occluded regions and large holes. In addition, acquiring and processing multiple depth views require significant computational power, which is not ideal in many applications that require real-time response.

In this paper, we aim to tackle the problem of inferring the complete 3D model of an object using a single depth view. This is a very challenging task, since the partial observation of the object (i.e. a depth image from one viewing angle) can be theoretically associated with infinite number of possible 3D models. Traditional reconstruction approaches typically use interpolation techniques such as plane fitting [51] or Poisson surface estimation [23] [24] to estimate the underlying 3D structure. However, they can only recover very limited occluded/missing regions, e.g. small holes or gaps due to quantization artifacts, sensor noise and insufficient geometry information.

Interestingly, humans are surprisingly talent at such ambiguity by implicitly leveraging prior knowledge. For example, given a view of a chair with two rear legs occluded by front legs, humans are easily able to guess the most likely shape behind the visible parts. Recent advances in deep neural nets and data driven approaches are suitable to deal with such a task.

In this paper, we aim to acquire the complete 3D geometry of an object given a single depth view. By utilizing the high performance of 3D convolutional neural nets and large open datasets of 3D models, our approach learns a smooth function to map a 2.5D view to a complete 3D shape. Particularly, we train an end-to-end model which estimates full volumetric occupancy from only one 2.5D depth view of an object, thus predicting occluded structures from a partial scan.

While state-of-the-art deep learning approaches [7] [61] [6] [58] [62] for 3D shape reconstruction achieve encouraging and compelling results, they are limited to a very small resolution, typically less than 40^3 voxel grids. As a result, the learnt 3D shape tends to be coarse and inaccurate. However, to increase the model resolution without sacrificing recovery accuracy is challenging, as even a slightly higher resolution would exponentially increase the search space of potential 2.5D to 3D mapping functions, resulting in difficulties in convergence of neural nets.

Recently, deep generative models achieve impressive success in modeling complex high-dimensional data distribution, among which Generative Adversarial Networks (GANs) [14] and Variational Autoencoders (VAEs) [27] emerge as two powerful frameworks for generative learning, including image and text generation [41] [20], and latent space learning [5] [28]. In the past two years, a number of works [13] [60] [15] [21] apply such generative models to learn latent space to represent 3D object shapes, and then to solve simple discriminative tasks such as new image generation, object classification, recognition and shape retrieval. However, 3D shape reconstruction, as a more difficult generative task, has yet to be fully explored.

In this paper, we propose 3D-RecGAN, a novel model that combines both an autoencoder and GAN to generate a full 3D structure conditioned on a single 2.5D view. Particularly, our model first encodes the 2.5D view to a low-dimensional latent space vector which implicitly represents general 3D geometric structures, then decodes it back to recover the most likely complete 3D structure. The rough 3D structure is then feed into a conditional discriminator which is adversarially trained to distinguish whether the coarse 3D shape is plausible or not.The autoencoder is able to approximate the corresponding shape, while the adversarial training tends to add fine details to the estimated shape. To ensure the final generated 3D shape corresponds to the input single partial 2.5D view, adversarial training of our model is based on conditional GAN [33] instead of random guessing.

Our contributions are as follows:

(1) We formulate a novel generative model to reconstruct the full 3D structure using a single arbitrary depth view. By drawing on both autoencoder and GAN, our approach is end-to-end trainable with high level of generality. Particularly, our model consumes a simple occupancy grid map without requiring object class labels or any annotations, while predicting a compelling shape with a high resolution of 64^3 voxel grid.

(2) We exploit conditional GAN during training to refine 3D shape estimates from autoencoder. Key contribution here is the use of a latent distribution rather than a binary variable from the discriminator to train both discriminator and autoencoder. Using a latent distribution of high-dimensional real or fake 3D reconstructed shapes from discriminator significantly stabilizes the training of GAN, while using the standard binary variable 0/1 for training leads to the GAN crash easily.

(3) We conduct extensive experiments for single category and multi-category reconstruction, outperforming the state of the art. Besides, our approach is also able to generalize previously unseen object categories.

We evaluate our approach on synthetic datasets from virtually scanned 3D CAD models. Ideally, this task should be evaluated on real world 2.5D depth views, but it is very challenging to obtain the ground truth of 3D shape with regard to a specific 2.5D view for both training and evaluation. To the best of our knowledge, there are no good open datasets which have the ground truth for occluded/missing parts and holes for each 2.5D view in real world. Extensive experiments demonstrate that our 3D-RecGAN outperforms the state of the art by a large margin. Our reconstruction results are not only quantitatively more accurate, but also qualitatively with more details. An example of chair completion is shown in Figure 1.

2. Related Work

We review different pipelines for 3D reconstruction or shape completion. Both conventional geometry based and the state-of-the-art deep learning based approaches are covered.

(1) **3D Model/Shape Fitting**. [35] uses plane fitting to complete small missing regions, while [32] [34] [40] [48] [52] [56] applies shape symmetry to fill in holes. Although these methods show good results, relying on predefined geometric regularities fundamentally limits the structure space to hand-crafted shapes. Besides, these approaches are likely to fail when missing or occluded regions are relatively big. Another similar fitting pipeline is to leverage database priors. Given a partial shape input, [25] [29] [36] [45] [47] try to retrieve an identical or most likely CAD model and align it with the partial scan. However, these approaches explic-

itly assume the database contains identical or very similar shapes, thus being unable to generalize novel objects or categories.

(2) **Multi-view Reconstruction**. Traditionally, 3D dense recovery requires a collection of images [19]. Geometric shape is recovered by dense feature extraction and matching [38], or by directly minimizing reprojection errors [2]. Basically, these methods are used for traditional SfM and visual SLAM, which is unable to build 3D structures for featureless regions such as white walls. Recently, [12] [42] [57] [54] [8] [6] [43] [49] [31] leverage deep neural nets to learn a 3D shape from multiple images. Although most of them do not directly require 3D ground-truth labels for supervision during training, they rely on additional signals such as contextual or camera information to supervise the view consistency. Obviously, extra efforts are required to acquire such additional signals. Additionally, resolution of the recovered occupancy shape is usually up to a small scale of 32^3.

(3) **Single-view Reconstruction**. Predicting a complete 3D object model from a single view is a long-standing and very challenging task. When reconstructing a specific object category, model templates can be used. For example, morphable 3D models are exploited for face recovery [3] [9]. This concept was extended to reconstruct simple objects in [22]. For general and complex object completion, recent machine learning approaches achieve promising results. Firman et al. [11] trained a random decision forest to predict unknown voxels. 3D ShapeNets [61] is amongst the early work using deep networks to predict multiple 3D solutions from a single partial view. Fan et al. [10] also adopted a similar strategy to generate multiple plausible 3D point clouds from a single image. However, that strategy is significantly less efficient than directly training an end-to-end predictor [7]. VConv-DAE [46] can be used for shape completion, but it is originally designed for shape denoising rather than partial range scans. Wu et al. proposed 3D-INN [59] to estimate a 3D skeleton from single image, which is far from recovering an accurate and complete 3D structure. Dai et al. developed 3D-EPN [7] to complete an object's shape using deep nets to both predict a 32^3 occupancy grid and then synthesize a higher resolution model based on a shape database. While it achieves promising results, it is not an end-to-end system and it relies on a prior model database. Perspective Transformer Nets [62] and the recent WS-GAN [18] are introduced to learn 3D object structures up to a 32^3 resolution occupancy grid. Although they do not need explicit 3D labels for supervision, it requires a large number of 2D silhouettes or masks and specific camera parameters. In addition, the training procedure of [62] is two-stage, rather than end-to-end. Song et al. [50] proposed SSCNet for both 3D scene completion and semantic label prediction. Although it outputs a high resolution occupancy map, it requires strong voxel-level annotations for supervision. It also needs special map encoding techniques such as elimination of both view dependency and strong gradients on TSDF. [55] [43] use tree structures, while [16] applies Hibert Maps for 3D map representation to recover the 3D shape, thus being able to produce a relatively higher resolution of 3D shape. However, their deep networks only consist of a 3D encoder and decoder, without taking advantage of adversarial learning. Varley et al. [58] provides an architecture for 3D shape completion from a single depth view, producing an up to 40^3 occupancy grid. Although reconstruction results are encouraging, the network is not scalable to higher resolution 3D shape because of the heavy fully connected layers.

3. 3D-RecGAN

3.1. Overview

Our method aims to predict a complete 3D shape of an object, which takes only an arbitrary single 2.5D depth view as input. The output 3D shape is automatically aligned with the corresponding 2.5D partial scan. To achieve this task, each object model is represented in a 3D voxel grid. We only use the simple occupancy information for map encoding, where 1 represents an occupied cell and 0 remains an empty cell. Specifically, both the input, denoted as I, and output 3D shape, denoted as Y, are 64^3 occupancy grids in our networks. The input shape is directly calculated from a single depth image. To generate ground true training and evaluation pairs, we virtually scan 3D objects from ModelNet40 [61]. Figure 2 is the t-SNE visualization of partial 2.5D views and the corresponding full 3D shapes for multiple general chair and bed models. Each green dot represents the t-SNE embedding of a 2.5D view, whilst a red dot is the embedding of corresponding 3D shapes. It can be seen that multiple categories inherently have similar 2.5D to 3D mapping relationships. Essentially, our neural network is to learn a smooth function, denoted as f, which maps green dots to red dots in high dimensional space as shown in Equation 1. The function f is parametrized by convolutional layers in general.

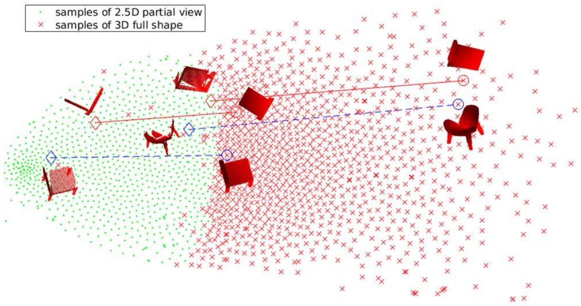

Figure 2. t-SNE embeddings of 2.5D partial views and 3D complete shapes of multiple object categories.

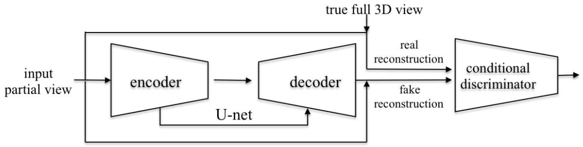

Figure 3. Overview of network architecture for training.

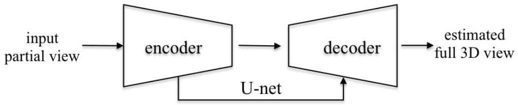

Figure 4. Overview of network architecture for testing.

$$Y = f(I) \quad \left(I, Y \in Z_2^{64^3}, where\ Z_2 = \{0,1\}\right) \qquad (1)$$

After generating training pairs, we feed them into our networks. The first part of our network loosely follows the idea of an autoencoder with U-net architecture [44]. The autoencoder serves as a generator which is followed by a conditional discriminator [33] for adversarial learning. Instead of reconstructing the original input and learning an efficient encoding, the autoencoder in our network aims to learn a correlation between partial and complete 3D structures. With the supervision of complete 3D labels, the autoencoder is able to learn a function f and generate a reasonable 3D shape given a brand new partial 2.5D view. In the testing phase, however, the results tend to be graining and without fine details.

To address this issue, in the training phase, the reconstructed 3D shape from the autoencoder is further fed into a conditional discriminator to verify its plausibility. In particular, a partial 2.5D input view is paired with its corresponding complete 3D shape, which is called the "real reconstruction", while the partial 2.5D view is paired with its corresponding output 3D shape from autoencoder, which is called "fake reconstruction". The discriminator aims to discriminate all "fake reconstruction" against "real reconstruction". In the original GAN framework [14], the task of discriminator is to simply classify real and fake input, but its Jensen-Shannon divergence-based loss function is difficult to converge. The recent WGAN [1] leverages Wasserstein distance with weight clipping as a loss function to stabilize the training procedure, whilst the extended work WGAN-GP [17] further improves the training process using a gradient penalty with respect to its input. In our 3D-RecGAN, we apply WGAN-GP as the loss function of our conditional discriminator, which guarantees fast and stable convergence. The overall network architecture for training is shown in Figure 3, while the testing phase only needs the well trained autoencoder as shown in Figure 4.

Overall, the main challenge of 3D reconstruction from an arbitrary single view is to generate new information including filling the missing and occluded regions from unseen views, while keeping the estimated 3D shape corresponding to the specific input 2.5D view. In the training phase, our 3D-RecGAN firstly leverages the autoencoder to generate a reasonable "fake reconstruction", then applies adversarial learning to refine the "fake reconstruction" to make it as similar to "real reconstruction" through jointly updating parameters of autoencoder. In the testing phase, given a novel 2.5D view as input, the jointly trained autoencoder is able to recover a full 3D model with satisfactory accuracy, while the discriminator is no longer used.

3.2. Architecture

Figure 5 shows the detailed architecture of our proposed 3D-RecGAN. It consists of two main networks: the generator as in the top block and the discriminator as in the bottom block.

The generator is based on autoencoder with skip-connections between encoder and decoder. Unlike the vanilla GAN generator which generates data from arbitrary latent distributions, our 3D-RecGAN generator synthesizes data from latent distribution of 2.5D views. Particularly, the encoder has five 3D convolutional layers, each of which has a bank of 4x4x4 filters with strides of 1x1x1, followed by a leaky ReLU activation function and a max pooling layer which has 2x2x2 filters and strides of 2x2x2. The number of output channels of max pooling layer starts with 64, doubling at each subsequent layer and ends up with 512. The encoder is lastly followed by two fully-connected layers to embed semantic information into latent space. The decoder is composed of 5 symmetric up-convolutional layers which are followed by ReLU activations except for the last layer with sigmoid function. Skip-connections between encoder and decoder guarantee propagation of local structures of the input 2.5D view. It should be noted that without the two fully connected layers and skip-connections, the vanilla autoencoder is unable to learn reasonable full 3D structures as the latent space is limited and the local structure is not preserved. During training, the generator is supervised supplying by ground true 3D shapes. The loss function and optimization methods are described in Section 3.3.

The discriminator aims to distinguish whether the estimated 3D shapes are plausible or not. Based on conditional GAN, the discriminator takes both real reconstruction pairs and fake reconstruction pairs as input. Particularly, it consists of five 3D convolutional layers, each of which has a bank of 4x4x4 filters with strides of 2x2x2, followed by a ReLU activation function except for the last layer which is followed by a sigmoid activation function. The number of output channels of each layer is the same as that in the encoder part. Unlike the original GAN and conditional GAN, our discriminator is not designed as a binary discriminator to simply classify fake against real reconstructions. The reason is both real reconstruction pairs and fake reconstruction pairs are extremely high dimensional distributions, i.e. $2 * 64^3$ dimensions. To naively classify it as only two cate-

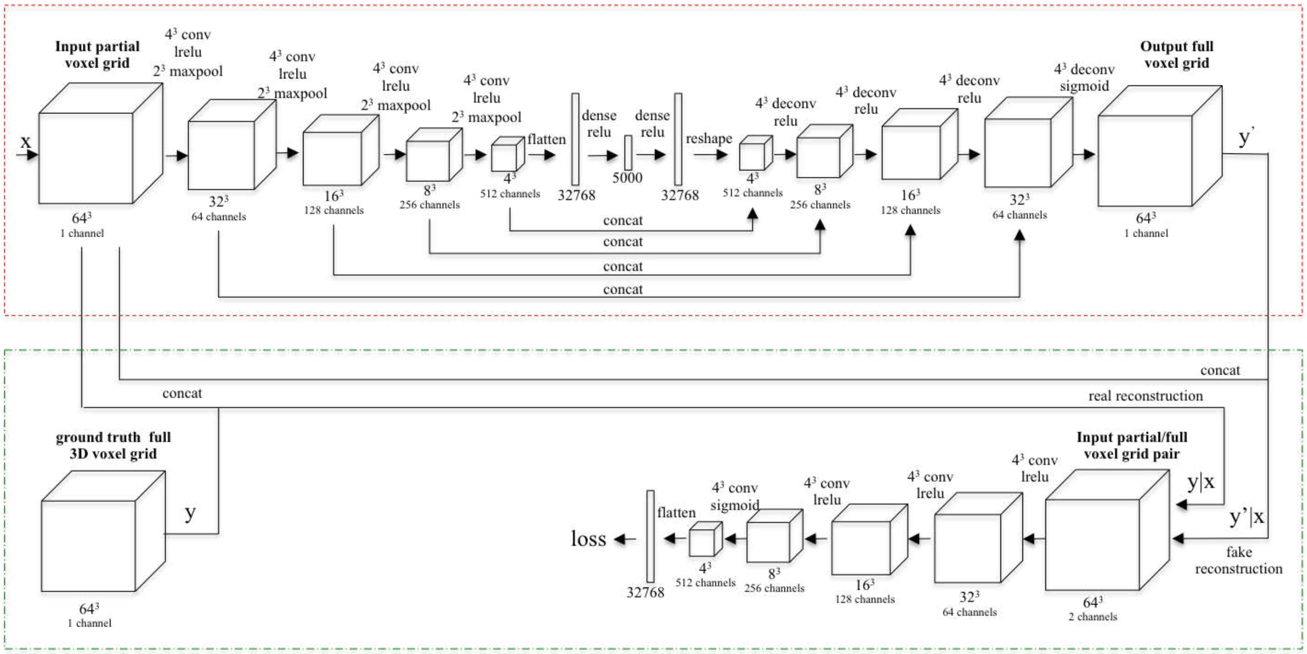

Figure 5. 3D-RecGAN Architecture.

gories would result in it being unable to capture geometric details of the object, and the discrimination loss is unlikely to benefit the generator through back-propagation. Instead, our discriminator is designed to output a long latent vector which represents distributions of real and fake reconstructions. Therefore, our discriminator is to distinguish the distributions of latent representations of fake and real reconstructions, while the generator is trained to make the two distributions as similar as possible. We use WGAN-GP as loss functions for our 3D-RecGAN.

3.3. Objectives

The objective function of our 3D-RecGAN includes two main parts: an object reconstruction loss L_{ae} for autoencoder based generator; the objective function L_{gan} for conditional GAN.

(1) L_{ae} For the generator, inspired by the work [4], we use modified binary cross-entropy loss function instead of the standard version. The standard binary cross-entropy weights both false positive and false negative results equally. However, most of the voxel grid tends to be empty and the network easily gets a false positive estimation. In this regard, we impose a high penalty on false positive than false negative results. Particularly, a weight hyper-parameter α is assigned to false positives, with $(1-\alpha)$ for false negative results, as shown in following Equation 2.

$$L_{ae} = -\alpha y \log(y^{'}) - (1-\alpha)(1-y)\log(1-y^{'}) \quad (2)$$

where y is the target value in {0,1} and $y^{'}$ is the estimated value in (0,1) for each voxel from the autoencoder.

(2) L_{gan} For the discriminator, we leverage the state-of-the-art WGAN-GP loss functions. Unlike the original GAN loss function which presents an overall loss for both real and fake inputs, we separately represent the loss function L^{g}_{gan} in Equation 3 for generating fake reconstruction pairs and L^{d}_{gan} in Equation 4 for discriminating fake and real reconstruction pairs. Detailed definitions and derivation of the loss functions can be found in [1] [17], but we modify them for our conditional GAN settings.

$$L^{g}_{gan} = -\mathbf{E}\left[D(y^{'}|x)\right] \quad (3)$$

$$\begin{aligned} L^{d}_{gan} = \mathbf{E}\left[D(y^{'}|x)\right] - \mathbf{E}\left[D(y|x)\right] \\ +\lambda\mathbf{E}\left[\left(\left\|\nabla_{\hat{y}}D(\hat{y}|x)\right\|_2 - 1\right)^2\right] \end{aligned} \quad (4)$$

where $\hat{y} = \epsilon x + (1-\epsilon)y^{'}, \epsilon \sim U[0,1]$. λ controls the tradeoff between optimizing the gradient penalty and the original objective in WGAN, x represents a voxel value, e.g.{0,1}, of an input 2.5D view, while $y^{'}$ is the estimated value in (0,1) for the corresponding voxel from generator, and y is the target value in {0,1} for the same voxel.

For the generator in our 3D-RecGAN network, there are two loss functions, L_{ae} and L^{g}_{gan}, to optimize. As we discussed in Section 3. Minimizing L_{ae} tends to learn the overall 3D shapes, whilst minimizing L^{g}_{gan} estimates more plausible 3D structures conditioned on input 2.5D views. To minimize L^{d}_{gan} is to improve the performance of discriminator to distinguish fake and real reconstruction pairs. To jointly optimize the generator, we assign weight β to L_{ae}, $(1-\beta)$ to L^{g}_{gan}. Overall, the loss functions for generator and discriminator are as follows:

$$L_g = \beta L_{ae} + (1-\beta)L^{g}_{gan} \tag{5}$$

$$L_d = L^{d}_{gan} \tag{6}$$

3.4. Training

We adopt an end-to-end training procedure for the whole network. To simultaneously optimize both generator and discriminator, we alternate between one gradient descent step on discriminator and then one step on generator. For the WGAN-GP, λ is set as 10 for gradient penalty as in [17]. α ends up as 0.85 for our modified cross entropy loss function, while β is 0.05 for the joint loss function L_g.

The Adam solver [26] is applied for both discriminator and generator with batch size of 8. The other three Adam parameters are set as default values, i.e. β_1 is 0.9, β_2 is 0.999 and ϵ is 1e-8. Learning rate is set to 0.0005 in the first epoch, decaying to 0.0001 in the following epochs. As we do not use dropout or batch normalization, the testing phase is exactly the same as training stage without reconfiguring network parameters. The whole network is trained on a single Titan X GPU from scratch.

3.5. Data Synthesis

For the task of 3D dense reconstruction from a single view, obtaining a large amount of training data is an obstacle. Existing real RGB-D datasets for surface reconstruction suffer from occlusions and missing data and there is no corresponding complete 3D structure for each single view. The recent work 3D-EPN [7] synthesizes data for 3D object completion, but their map encoding scheme is the complicated TSDF which is different from our network requirement.

To tackle this issue, we use the ModelNet40 [61] database to generate a large amount of training and testing data with synthetically rendered depth images and the corresponding complete 3D shape ground truth. Particularly, a subset of object categories is selected for our experiments. For each category, we generate training data from around 200 CAD models in the train folder, while synthesizing testing data from around 20 CAD models in the test folder. For each CAD model, we create a virtual depth camera to scan it from 125 different angles, 5 uniformly sampled views for each of roll, pitch and yaw space. For each virtual scan, both a depth image and the corresponding complete 3D voxelized structure are generated with regard to the same camera angle. That depth image is simultaneously transformed to a partial 2.5D voxel grid using virtual camera parameters. Then a pair of partial 2.5D view and the complete 3D shape is synthesized. Overall, around 20K training pairs and 2K testing pairs are generated for each 3D object category. All data are produced in Blender.

4. Evaluation

In this section, we evaluate our 3D-RecGAN with comparison to alternative approaches and an ablation study to fully investigate the proposed network.

4.1. Metrics

We use two metrics to evaluate the performance of 3D reconstruction. The first metric is voxel Intersection-over-Union (IoU) between a predicted 3D voxel grid and its ground true voxel grid. It is formally defined as follows:

$$IoU = \frac{\sum_{ijk}\left[I(y^{'}_{ijk} > p) * I(y_{ijk})\right]}{\sum_{ijk}\left[I\left(I(y^{'}_{ijk} > p) + I(y_{ijk})\right)\right]}$$

where $I()$ is an indicator function, (i,j,k) is the index of a voxel in three dimensions, $y^{'}_{ijk}$ is the predicted value at the (i,j,k) voxel, y_{ijk} is the ground true value at (i,j,k), and p is the threshold for voxelization. In all our experiments, p is set as 0.5. If the predicted value is over 0.5, it is more likely to be occupied from the probabilistic aspect. The higher the IoU value, the better the reconstruction of a 3D model.

The second metric is the mean value of standard cross-entropy loss (CE) between a reconstructed shape and the ground true 3D model. It is formally presented as:

$$CE = \frac{1}{IJK}\sum_{ijk}\left[y_{ijk}\log(y^{'}_{ijk}) + (1-y_{ijk})\log(1-y^{'}_{ijk})\right]$$

where $y^{'}_{ijk}$ and y_{ijk} are the same as defined in above IoU, (I, J, K) are the voxel dimension sizes of output 3D models. The lower CE value is, the better 3D prediction.

The above two metrics can evaluate the overall reconstruction performance, but the reconstructed geometric details are unlikely to be well evaluated in such way. Therefore, a large number of qualitative results from reconstructed 3D models are visualized in Section 4.2.

4.2. Comparison

We compare against two alternative reconstruction methods. The first is the well-known traditional Poisson surface reconstruction [23] [24], which is mostly used for completing surfaces on dense point clouds. The second is the state-of-the-art deep learning based approach proposed by Varley et al. in [58], which is most similar to our approach in terms of input and output data encoding and the 3D completion task. It has encouraging reconstruction performance because of its two fully connected layers [30] in the model, but it is unable to deal with higher resolutions and it has less generality for shape completion. We also compare against the autoencoder alone in our network, i.e. without the GAN, named as 3D-RecAE for short.

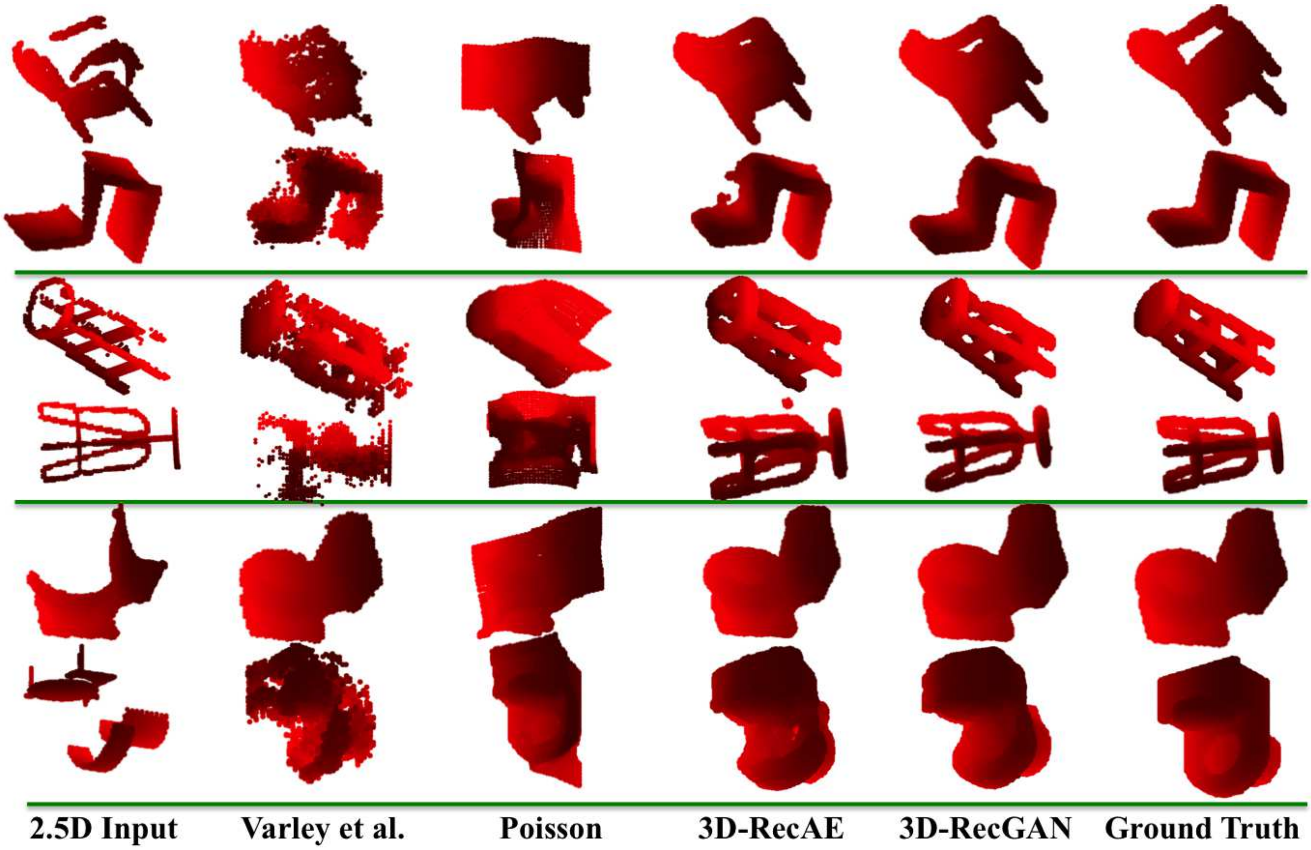

Figure 6. Qualitative results of per-category reconstruction from different approaches.

(1) **Per-category Results**. The networks are separately trained and tested on three different categories with the same network configurations. Table 1 shows the IoU and CE results, and Figure 6 compares qualitative results from different reconstruction approaches.

Table 1. Per-category IoU and CE Loss.

	IoU			CE Loss		
trained/tested on	chair	stool	toilet	chair	stool	toilet
Poisson	0.180	0.189	0.150	-	-	-
Varley [58]	0.564	0.273	0.503	0.132	0.189	0.177
3D-RecAE	0.633	0.488	0.520	**0.069**	0.085	0.166
3D-RecGAN	**0.661**	**0.501**	**0.569**	0.074	**0.083**	**0.157**

Table 2. Multi-category IoU and CE Loss.

	IoU		CE Loss	
trained/ tested on	chair/toilet	chair/toilet /stool	chair/toilet	chair/toilet /stool
Poisson	0.165	0.173	-	-
Varley [58]	0.493	0.453	0.125	0.173
3D-RecAE	0.514	0.487	0.127	0.109
3D-RecGAN	**0.554**	**0.513**	**0.117**	**0.101**

(2) **Multi-category Results**. To study the generality, the networks are trained and tested on multiple categories without given any class labels. Table 2 shows the IoU and CE results, and Figure 7 shows the qualitative results.

(3) **Cross-category Results**. To further investigate the generality, the network is trained on one category, but tested on another five different categories. Particularly, in Group 1, the network is trained on chair, tested on sofa, stool, table, toilet, and TV stand; in Group 2, the network is trained on stool, tested on chair, sofa, table, toilet, and TV stand; in Group 3, the network is trained on toilet, tested on chair, sofa, stool, table, and TV stand. Table 3 shows the IoU and CE results; Figure 8, 9 and 10 compare qualitative cross-category reconstruction results of Group 1, Group 2 and Group 3 respectively.

Table 3. Cross-category IoU and CE Loss.

	IoU			CE Loss		
	Group1	Group2	Group3	Group1	Group2	Group3
Varley [58]	0.253	0.221	0.277	0.430	0.425	0.297
3D-RecAE	0.353	0.362	0.349	**0.218**	**0.117**	**0.149**
3D-RecGAN	**0.356**	**0.369**	**0.351**	0.264	0.345	0.162

Overall, the above extensive experiments for per-category and multi-category object reconstruction demonstrate that our proposed 3D-RecGAN is able to complete partial 2.5D views with accurate structures and fine-grained details, outperforming the state of the art by a large margin. In addition, our 3D-RecGAN performs well in the challenging cross-category reconstruction task, which demonstrates

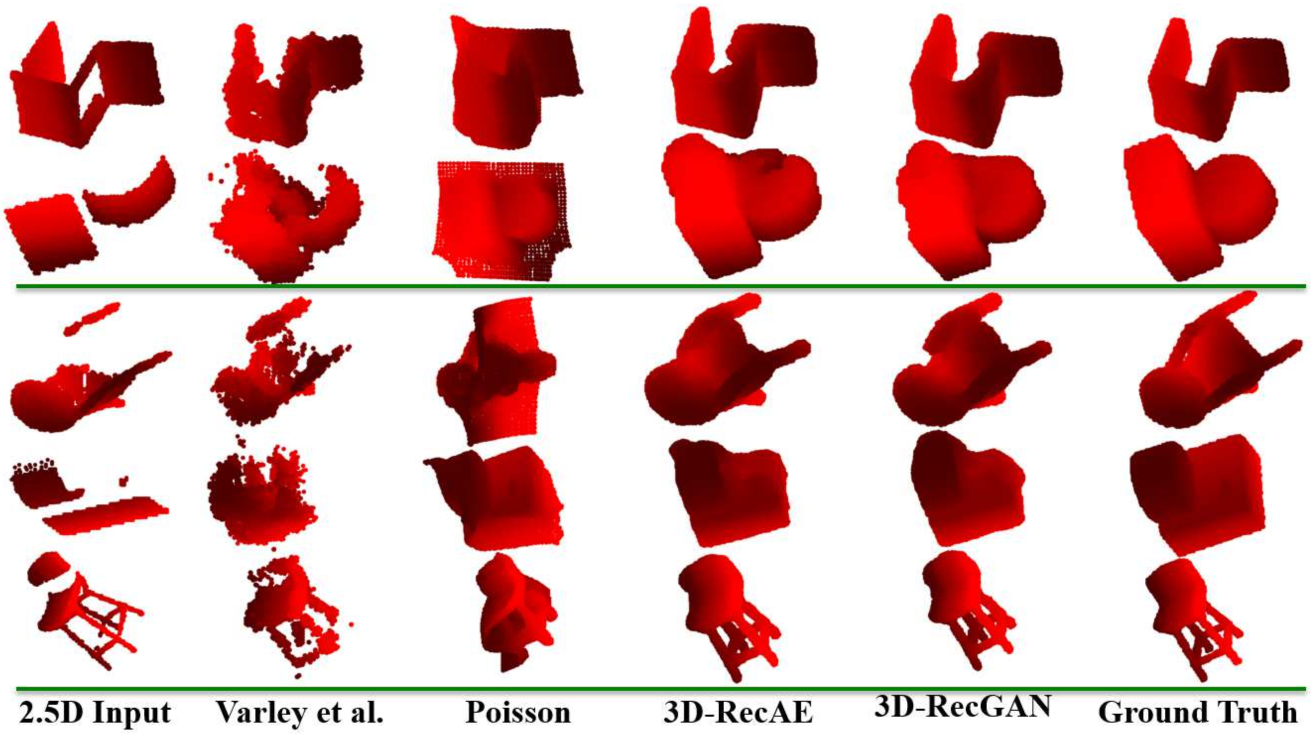

Figure 7. Qualitative results of multi-category reconstruction from different approaches.

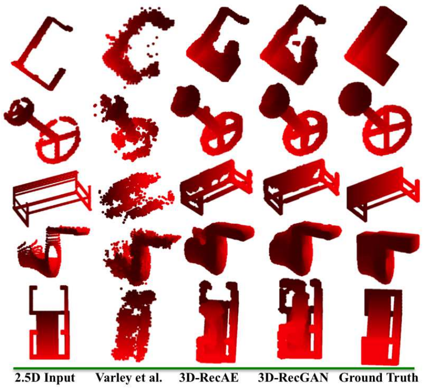

Figure 8. Cross-category reconstruction results of Group 1.

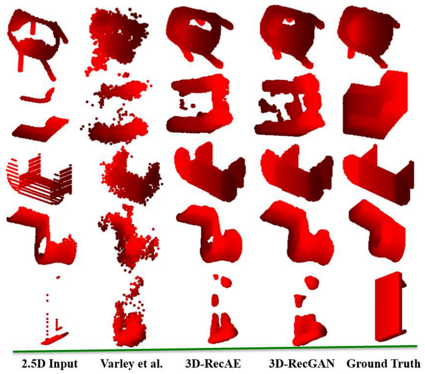

Figure 9. Cross-category reconstruction results of Group 2.

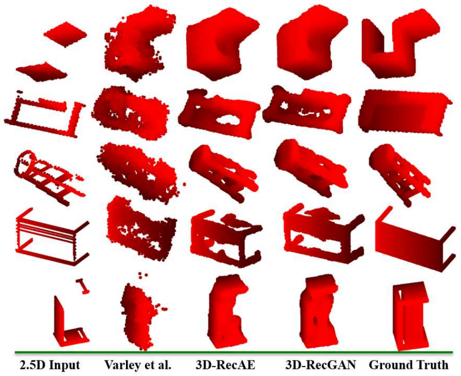

Figure 10. Cross-category reconstruction results of Group 3.

that our novel model implicitly learns the geometric features and their correlations among different object categories.

5. Conclusion

In this work, we proposed a novel framework 3D-RecGAN that reconstructs the full 3D structure of an object from an arbitrary depth view. By leveraging the generalization capabilities of autoencoders and generative networks, our 3D-RecGAN predicts accurate 3D structures with fine details, outperforming the traditional Poisson algorithm and the method in Varley et al.[58] in single-view shape completion for individual object category. We further tested our network's ability to perform reconstruction on multiple categories without providing any object class labels during training or testing, and it showed that our network is able to predict satisfactory 3D shapes. Finally, we investigated the network's reconstruction performance on unseen categories of objects. We showed that even in very challenging cases, the proposed approach can still predict plausible 3D shapes. This confirms that our network has the capability of learning general 3D latent features of the objects, rather than simply fitting a function for the training datasets. In summary, our network only requires a single depth view to recover an accurate complete 3D shape with fine details.

References

[1] M. Arjovsky, S. Chintala, and L. Bottou. Wasserstein GAN. *ICML*, 2017. 4, 5

[2] S. Baker and I. Matthews. Lucas-Kanade 20 Years On : A Unifying Framework : Part 1. *International Journal of Computer Vision*, 56(3):221–255, 2004. 3

[3] V. Blanz and T.Vetter. Face Recognition based on Fitting a 3D Morphable Model. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 25(9):1063–1074, 2003. 3

[4] A. Brock, T. Lim, J. M. Ritchie, and N. Weston. Generative and Discriminative Voxel Modeling with Convolutional Neural Networks. *arXiv*, 2016. 5

[5] X. Chen, Y. Duan, R. Houthooft, J. Schulman, I. Sutskever, and P. Abbeel. InfoGAN: Interpretable Representation Learning by Information Maximizing Generative Adversarial Nets. *NIPS*, 2016. 2

[6] C. B. Choy, D. Xu, J. Gwak, K. Chen, and S. Savarese. 3D-R2N2: A Unified Approach for Single and Multi-view 3D Object Reconstruction. *ECCV*, 2016. 2, 3

[7] A. Dai, C. R. Qi, and M. Nießner. Shape Completion using 3D-Encoder-Predictor CNNs and Shape Synthesis. *CVPR*, 2017. 2, 3, 6

[8] X. Di, R. Dahyot, and M. Prasad. Deep Shape from a Low Number of Silhouettes. *ECCV*, 2016. 3

[9] P. Dou, S. K. Shah, and I. A. Kakadiaris. End-to-end 3D face reconstruction with deep neural networks. *CVPR*, 2017. 3

[10] H. Fan, H. Su, and L. Guibas. A Point Set Generation Network for 3D Object Reconstruction from a Single Image. *CVPR*, 2017. 3

[11] M. Firman, O. M. Aodha, S. Julier, and G. J. Brostow. Structured Prediction of Unobserved Voxels From a Single Depth Image. *CVPR*, 2016. 3

[12] M. Gadelha, S. Maji, and R. Wang. 3D Shape Induction from 2D Views of Multiple Objects. *arXiv*, 2016. 3

[13] R. Girdhar, D. F. Fouhey, M. Rodriguez, and A. Gupta. Learning a Predictable and Generative Vector Representation for Objects. *ECCV*, 2016. 2

[14] I. J. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio. Generative Adversarial Nets. *NIPS*, 2014. 2, 4

[15] E. Grant, P. Kohli, and M. V. Gerven. Deep Disentangled Representations for Volumetric Reconstruction. *ECCV Workshops*, 2016. 2

[16] V. Guizilini and F. Ramos. Learning to Reconstruct 3D Structures for Occupancy Mapping. *RSS*, 2017. 3

[17] I. Gulrajani, F. Ahmed, M. Arjovsky, V. Dumoulin, and A. Courville. Improved Training of Wasserstein GANs. *arXiv*, 2017. 4, 5, 6

[18] J. Gwak, C. B. Choy, A. Garg, M. Chandraker, and S. Savarese. Weakly Supervised Generative Adversarial Networks for 3D Reconstruction. *arXiv*, 2017. 3

[19] R. Hartley and A. Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, 2004. 3

[20] Z. Hu, Z. Yang, X. Liang, R. Salakhutdinov, and E. P. Xing. Controllable Text Generation. *ICML*, 2017. 2

[21] H. Huang, E. Kalogerakis, and B. Marlin. Analysis and synthesis of 3D shape families via deep-learned generative models of surfaces. *Computer Graphics Forum*, 34(5):25–38, 2015. 2

[22] A. Kar, S. Tulsiani, J. Carreira, and J. Malik. Category-specific object reconstruction from a single image. *CVPR*, 2015. 3

[23] M. Kazhdan, M. Bolitho, and H. Hoppe. Poisson Surface Reconstruction. *Symposium on Geometry Processing*, 2006. 1, 6

[24] M. Kazhdan and H. Hoppe. Screened poisson surface reconstruction. *ACM Transactions on Graphics*, 32(3):1–13, 2013. 1, 6

[25] Y. M. Kim, N. J. Mitra, D.-M. Yan, and L. Guibas. Acquiring 3D Indoor Environments with Variability and Repetition. *ACM Transactions on Graphics*, 31(6), 2012. 2

[26] D. P. Kingma and J. Ba. Adam: A method for stochastic optimization. *ICLR*, 2015. 6

[27] D. P. Kingma and M. Welling. Auto-Encoding Variational Bayes. *ICLR*, 2014. 2

[28] T. D. Kulkarni, W. F. Whitney, P. Kohli, and J. B. Tenenbaum. Deep Convolutional Inverse Graphics Network. *NIPS*, 2015. 2

[29] Y. Li, A. Dai, L. Guibas, and M. Nießner. Database-Assisted Object Retrieval for Real-Time 3D Reconstruction. *Computer Graphics Forum*, 34(2):435–446, 2015. 2

[30] J. Long, E. Shelhamer, and T. Darrell. Fully Convolutional Networks for Semantic Segmentation. *CVPR*, 2015. 6

[31] Z. Lun, M. Gadelha, E. Kalogerakis, S. Maji, and R. Wang. 3D Shape Reconstruction from Sketches via Multi-view Convolutional Networks. *arXiv*, 2017. 3

[32] O. Mattausch, D. Panozzo, C. Mura, O. Sorkine-Hornung, and R. Pajarola. Object detection and classification from large-scale cluttered indoor scans. *Computer Graphics Forum*, 33(2):11–21, 2014. 2

[33] M. Mirza and S. Osindero. Conditional Generative Adversarial Nets. *arXiv*, 2014. 2, 4

[34] N. J. Mitra, L. J. Guibas, and M. Pauly. Partial and Approximate Symmetry Detection for 3D Geometry. *SIGGRAPH*, 2006. 2

[35] A. Monszpart, N. Mellado, G. J. Brostow, and N. J. Mitra. RAPter: Rebuilding Man-made Scenes with Regular Arrangements of Planes. *ACM Transactions on Graphics*, 34(4):1–12, 2015. 2

[36] L. Nan, K. Xie, and A. Sharf. A Search-Classify Approach for Cluttered Indoor Scene Understanding. *ACM Transactions on Graphics*, 31(6):1–10, 2012. 2

[37] R. A. Newcombe, S. Izadi, O. Hilliges, D. Molyneaux, D. Kim, A. J. Davison, P. Kohli, J. Shotton, S. Hodges, and A. Fitzgibbon. KinectFusion: Real-time dense surface mapping and tracking. *ISMAR*, 2011. 1

[38] R. A. Newcombe, S. J. Lovegrove, and A. J. Davision. DTAM: Dense Tracking and Mapping in Real-time. *ICCV*, 2011. 3

[39] M. Nießner, M. Zollhöfer, S. Izadi, and M. Stamminger. Real-time 3D reconstruction at scale using voxel hashing. *ACM Transactions on Graphics*, 32(6):1–11, 2013. 1

[40] M. Pauly, N. J. Mitra, J. Wallner, H. Pottmann, and L. J. Guibas. Discovering structural regularity in 3D geometry. *ACM Transactions on Graphics*, 27(3):1, 2008. 2

[41] A. Radford, L. Metz, and S. Chintala. Unsupervised Representation Learning with Deep Convolutional Generative Adversarial Networks. *ICLR*, 2016. 2

[42] D. J. Rezende, S. M. A. Eslami, S. Mohamed, P. Battaglia, M. Jaderberg, and N. Heess. Unsupervised Learning of 3D Structure from Images. *NIPS*, 2016. 3

[43] G. Riegler, A. O. Ulusoy, H. Bischof, and A. Geiger. OctNetFusion: Learning Depth Fusion from Data. *arXiv*, 2017. 3

[44] O. Ronneberger, P. Fischer, and T. Brox. U-Net : Convolutional Networks for Biomedical Image Segmentation. *MICCAI*, 2015. 4

[45] T. Shao, W. Xu, K. Zhou, J. Wang, D. Li, and B. Guo. An interactive approach to semantic modeling of indoor scenes with an RGBD camera. *ACM Transactions on Graphics*, 31(6):1–11, 2012. 2

[46] A. Sharma, O. Grau, and M. Fritz. VConv-DAE : Deep Volumetric Shape Learning Without Object Labels. *ECCV*, 2016. 1, 3

[47] Y. Shi, P. Long, K. Xu, H. Huang, and Y. Xiong. Data-driven contextual modeling for 3d scene understanding. *Computers & Graphics*, 55:55–67, 2016. 2

[48] I. Sipiran, R. Gregor, and T. Schreck. Approximate Symmetry Detection in Partial 3D Meshes. *Computer Graphics Forum*, 33(7):131–140, 2014. 2

[49] A. A. Soltani, H. Huang, J. Wu, T. D. Kulkarni, and J. B. Tenenbaum. Synthesizing 3D Shapes via Modeling Multi-View Depth Maps and Silhouettes with Deep Generative Networks. *CVPR*, 2017. 3

[50] S. Song, F. Yu, A. Zeng, A. X. Chang, M. Savva, and T. Funkhouser. Semantic Scene Completion from a Single Depth Image. *CVPR*, 2017. 3

[51] O. Sorkine and D. Cohen-Or. Least-squares Meshes. *SMI*, 2004. 1

[52] P. Speciale, M. R. Oswald, A. Cohen, and M. Pollefeys. A Symmetry Prior for Convex Variational 3D Reconstruction. *ECCV*, 2016. 2

[53] F. Steinbrucker, C. Kerl, J. Sturm, and D. Cremers. Large-Scale Multi-Resolution Surface Reconstruction from RGB-D Sequences. *ICCV*, 2013. 1

[54] M. Tatarchenko, A. Dosovitskiy, and T. Brox. Multi-view 3D models from single images with a convolutional network. *ECCV*, 2016. 3

[55] M. Tatarchenko, A. Dosovitskiy, and T. Brox. Octree Generating Networks : Efficient Convolutional Architectures for High-resolution 3D Outputs. *ICCV*, 2017. 3

[56] S. Thrun and B. Wegbreit. Shape from symmetry. *ICCV*, 2005. 2

[57] S. Tulsiani, T. Zhou, A. A. Efros, and J. Malik. Multi-view Supervision for Single-view Reconstruction via Differentiable Ray Consistency. *CVPR*, 2017. 3

[58] J. Varley, C. Dechant, A. Richardson, J. Ruales, and P. Allen. Shape Completion Enabled Robotic Grasping. *IROS*, 2017. 1, 2, 3, 6, 7, 8

[59] J. Wu, T. Xue, J. J. Lim, Y. Tian, J. B. Tenenbaum, A. Torralba, and W. T. Freeman. Single Image 3D Interpreter Network. *ECCV*, 2016. 3

[60] J. Wu, C. Zhang, T. Xue, W. T. Freeman, and J. B. Tenenbaum. Learning a Probabilistic Latent Space of Object Shapes via 3D Generative-Adversarial Modeling. *NIPS*, 2016. 2

[61] Z. Wu, S. Song, A. Khosla, F. Yu, L. Zhang, X. Tang, and J. Xiao. 3D ShapeNets : A Deep Representation for Volumetric Shapes. *CVPR*, 2015. 2, 3, 6

[62] X. Yan, J. Yang, E. Yumer, Y. Guo, and H. Lee. Perspective Transformer Nets: Learning Single-View 3D Object Reconstruction without 3D Supervision. *NIPS*, 2016. 2, 3