Chapter 3

Weakly-Supervised Reconstruction

3.1 Introduction

Recovering the three-dimensional (3D) shape of an object is a fundamental attribute of human perception. This problem has been explored by a large body of work in computer vision, within domains such as structure from motion [18, 12] or multiview stereo [13, 14, 16, 19]. While tremendous success has been achieved with conventional approaches, they often require several images to either establish accurate correspondences or ensure good coverage. This has been especially true of methods that rely on weak cues such as silhouettes [34] or aim to recover 3D volumes rather than point clouds or surfaces [24]. In contrast, human vision seems adept at 3D shape estimation from a single or a few images, which is also a useful ability for tasks such as robotic manipulation and augmented reality.

The advent of deep neural networks has allowed incorporation of semantic concepts and prior knowledge learned from large-scale datasets of examples, which has translated into approaches that achieve 3D reconstruction from a single or sparse viewpoints [6, 49, 15, 45, 46]. But conventional approaches to train convolutional neural networks (CNNs) for 3D reconstruction requires large-scale supervision. To learn the mapping from images to shapes, CAD models or point clouds are popularly used. However, ground truth alignments of models to images are challenging and expensive to acquire. Thus, existing datasets that contain an image to 3D model mappings simply label the closest model as ground truth [48, 47], which leads to suboptimal training.

This paper presents a framework for volumetric shape reconstruction using silhouettes (foreground mask) from a single or sparse set of viewpoints and camera viewpoints as input. Visual hull reconstruction from such inputs is an ill-posed problem no matter how many views are given (Fig. 3.1). For example, concavity cannot be recovered from silhouettes while it may contain crucial information regarding the functionality of the objects such as cups and chairs. In addition, it is difficult to collect dense viewpoints of silhouettes of the reconstruction target in practical settings

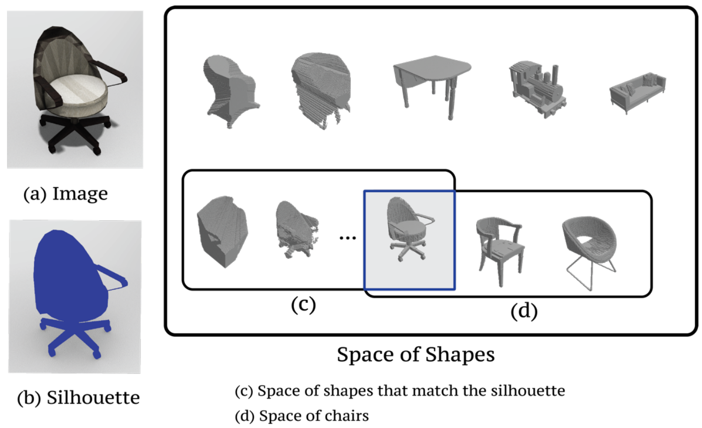

Figure 3.1: The 3D reconstruction using foreground masks (silhouette) is an ill-posed problem. Instead, we propose using a manifold constraint to regularize the ill posed problem.

such as in online retailers. Therefore, in order solve such ill-posed problem, we regularize the space of valid solution. For example, given an image or images of a chair, we make the reconstruction to be a *seatable* chair with concavity, which cannot be recovered from silhouettes. This problem becomes a constrained optimization where we solve

$$\begin{aligned} \underset{x}{\text{minimize}} \quad & \text{ReprojectionError}(x) \\ \text{subject to} \quad & \text{Reconstruction } x \text{ to be a valid chair} \end{aligned} \tag{3.1}$$

where x is the 3D reconstruction. We denote the space of valid chairs as **the manifold of realistic shapes**, $\mathcal{M}$ which can be defined using a set of hand-designed shapes or scanned 3D shapes, denoted as $\{x_i^\star\}_i$. Then, the constraint can be written concisely as

$$\text{subject to } x \in \mathcal{M}$$

We solve the above constrained optimization using the log barrier method [4] and learn the barrier function using $\{x_i^\star\}_i$. The log barrier function that we learn is similar to the discriminator in many variants of Generative Adversarial Networks [50, 20]. We differ in framing the problem as constrained optimization to make it *explicit* that we need the manifold constraint to solve such ill-posed problems and to provide a principled rationale for using an adversarial setting. Our formulation also allows clearer distinctions from other use of manifold and discriminators in Sec. 3.3.2.

To model the reprojection error, we propose a raytrace pooling layer in Sec. 3.3.3 that mimics the conventional volumetric reconstruction methods such as voxel carving [24] and does not suffer from aliasing compared to [49]. Once we train the network, it only uses images at test time.

In Sec. 5.7, we experimentally evaluate our framework using three different datasets and report quantitative reductions in error compared with various baselines. Our experiments demonstrate that the proposed framework better encapsulates semantic or category-level shape information while requiring less supervision or relatively inexpensive weak supervision compared to prior works [6, 49]. In contrast to traditional voxel carving, our manifold constraint allows recovering concavities by restricting the solution to the set of plausible shapes. Quantitative advantages of our framework are established by extensive validation and ablation study on ShapeNet, ObjectNet3D and OnlineProduct datasets.

3.2 Prior Work

In this section, we briefly discuss prior works related to the three aspects of our framework: Convolutional Neural Networks for 3D data, supervised 3D reconstruction and Generative Adversarial Networks.

3D Convolutional Neural Networks. First introduced in video classification, the 3D Convolutional Neural Networks have been widely used as a tool for spatiotemporal data analysis [22, 2, 40, 28, 41]. Instead of using the third dimension for temporal convolution, [46, 25] use the third dimension for the spatial convolution and propose 3D convolutional deep networks for 3D shape classification. Recently, 3D-CNNs have been widely used for various 3D data analysis tasks such as 3D detection or classification [39, 31, 29], semantic segmentation [7, 32] and reconstruction [44, 6, 45, 15, 49]. Our work is closely related to those that use the 3D-CNN for reconstruction, as discussed in the following section.

Supervised 3D voxel reconstruction. Among many lines of work within the 3D reconstruction [18, 24, 13, 14, 3, 8, 23, 35, 44, 33], ours is related to recent works that use neural networks for 3D voxel reconstruction. Grant *et al.*[15] propose an autoencoder to learn the 3D voxelized shape embedding and regress to the embedding from 2D images using a CNN and generated 3D voxelized shape from a 2D image. Choy *et al.*[6] use a 3D-Convolutional Recurrent Neural Network to directly reconstruct a voxelized shape from multiple images of the object. The work of [45] combines a 3D-CNN with a Generative Adversarial Network to learn the latent space of 3D shapes. Given the latent space of 3D shapes, [45] regresses the image feature from a 2D-CNN to the latent space to reconstruct a single-view image. These approaches require associated 3D shapes for training. Recently, Yan *et al.*[49] propose a way to train a neural network to reconstruct 3D shapes using a large number of foreground masks (silhouettes) and viewpoints for weak supervision. The silhouette is used to carve out spaces analogous to voxel carving [24, 36, 26] and to generate the visual hull.

Our work is different from [49, 42] in that it makes use of both unmatched 3D shape and inexpensive 2D weak supervision to generate realistic 3D shapes without explicit 3D supervision. This allows the network to learn reconstruction with minimal 2D supervision (as low as one view 2D mask). And the key mechanism that allows such 2D weak supervision is the projection. Unlike [49], we propose the Raytrace Pooling layer that is not limited to the grid sampling and experimentally compare with it in Sec. 3.4.3. In addition, we use a recurrent neural network that can handle both single and multi-view images as the weak supervision is done on single or multi view images.

3.3 Weakly supervised 3D Reconstruction with Adversarial Constraint

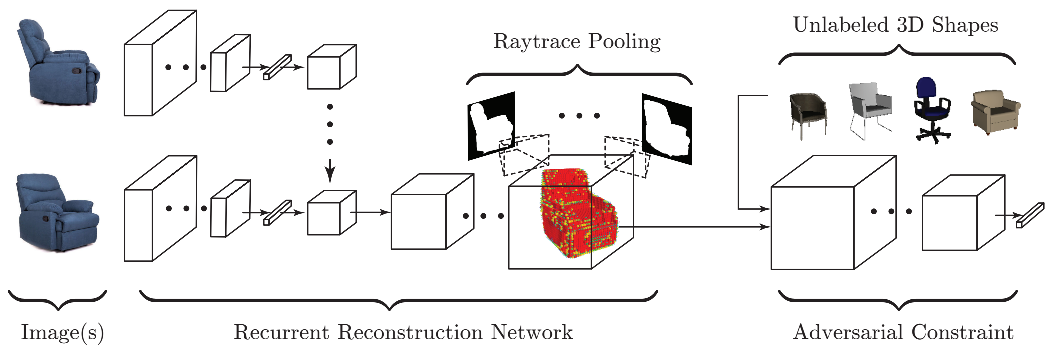

Figure 3.2: Visualization of McRecon network structure. Our network encodes a set of images into a latent variable. Then, the latent variable is decoded into a voxel representation of 3D shape. Perspective Raytrace Pooling layer renders this 3D shape into 2D occupancy map, allowing us to give mask supervision. Additionally, discriminator takes the generated voxel as an input, filling the missing information of the 3D shape distribution learned from unlabeled 3D models.

Recent supervised single view reconstruction methods [15, 6, 44, 45] require associated 3D shapes. However, such 3D annotations are hard to acquire for real image datasets such as [9, 38]. Instead, we propose a framework, termed as Weakly supervised 3D Reconstruction with Adversarial Constraint (McRecon), that relies on inexpensive 2D silhouette and approximate viewpoint for weak supervision. McRecon makes use of unlabeled 3D shapes to constrain the ill-posed single/sparse-view reconstruction problem. In this section, we propose how we solve the constrained optimization in 3.1 using the log barrier method and show the connection between the constrained optimization and the Generative Adversarial Networks. Then we define the reprojection error using ray tracing and conclude the section with the optimization of the entire framework.

3.3.1 Log Barrier for Constrained Optimization

McRecon solves the constrained optimization problem where we minimize the reprojection error of the reconstruction while constraining the reconstruction to be in the manifold of realistic 3D shapes

(Eq. 3.1). Formally,

$$\begin{aligned} \underset{\hat{x}}{\text{minimize}} \quad & \mathop{\mathbb{E}}_{v \in \text{views}} [\mathcal{L}_{\text{reproj.}}(\hat{x}, c_v, m_v)] \\ \text{subject to} \quad & \hat{x} \in \mathcal{M} \end{aligned} \tag{3.2}$$

where $L_{\text{reproj.}}(\cdot, \cdot)$ denotes the reprojection error, x denotes the final reconstruction, m_v and c_v denote the foreground mask (silhouette) and associated camera viewpoint. We use a neural network $f(\cdot; W)$, composition of N functions parametrized by θ_f, to model the reconstruction function which takes multiview images $\mathbf{I}$ as an input.

$$\hat{x} = f(\mathbf{I}; \theta_f) \quad f \coloneqq f_N \circ f_{N-1} \circ ... \circ f_1 \tag{3.3}$$

Specifically, we use the log barrier method [4] and denote the penalty function as $g(x)$ and $g(x) = 1$ iff $x \in \mathcal{M}$ otherwise 0. Then the constrained optimization problem in Eq. 3.2 becomes an unconstrained optimization problem where we solve

$$\underset{\hat{x}}{\text{minimize}} \mathop{\mathbb{E}}_{v \in \text{views}} [\mathcal{L}_{\text{reproj.}}(\hat{x}, c_v, m_v)] - \frac{1}{t} \log g(\hat{x}) \tag{3.4}$$

As $t \to \infty$, the log barrier becomes an indicator function for the constraint violation. However, the function $g(\cdot)$ involves high level cognition (does the shape look like a chair?) which captures all underlying constraints that make a 3D shape look like a valid shape: geometric constraints (symmetry, physical stability), and semantic constraints (e.g. chairs should have concavity for a seat, a backrest is next to a seat). Naturally, the function cannot be simply approximated using hand designed functions.

3.3.2 Learning the Barrier for Manifold Constraint

Instead of hand-designing the constraint violation, we learn the constraint violation function $-\log g(\cdot)$ using a neural network. Specifically, we use the adversarial setting in [17] to **a)** adaptively learn the violation that the current generative model is violating the most, **b)** to capture constraints that are difficult to model, such as geometric constraints and semantic constraints, **c)** allow the reconstruction function to put more emphasis on the part that the current barrier focuses on as the penalty function becomes progressively more difficult.

To understand the penalty function $-\log g(\cdot)$, we should analyze the ideal scenario where the discriminator perfectly discriminates the reconstruction $\hat{x} = f(\mathbf{I})$ from the real 3D shapes $x^\star$. The ideal discriminator $g^\star(x)$ will output a value 1 when x is realistic and the log barrier will be $-\log 1 = 0$. On the other hand, if the reconstruction is not realistic (i.e. violates any physical or semantic constraints), then the discriminator will output 0 making the log barrier $-\log 0 = \infty$. Thus, the ideal discriminator works perfectly as the manifold constraint penalty function.

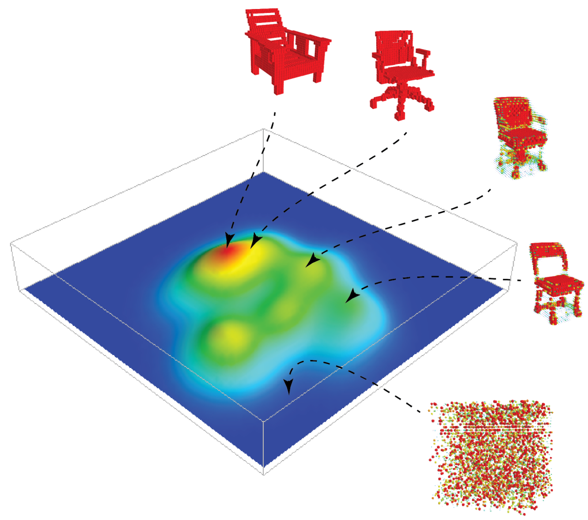

Figure 3.3: Illustration of the penalty function. $g(x)$ learns the manifold of realistic hand-designed or scanned 3D shapes.

We learn the penalty function by regressing the values and minimizing the following objective function.

$$\underset{g}{\text{minimize}} \quad \mathop{\mathbb{E}}_{x^\star \sim p} \log g(x^\star) + \mathop{\mathbb{E}}_{\hat{x} \sim q} \log(1 - g(\hat{x})) \tag{3.5}$$

where p and q denote the distribution of the unlabeled 3D shapes and the reconstruction, respectively.

Penalty Functions and Discriminators

The log barrier we propose is similar to the discriminators in many variants of Generative Adversarial Networks that model the perceptual loss [50, 20, 37]. The discriminators work by learning the distribution of the real images and fake images and thus, it is related to learning the penalty. However, to the best of our knowledge, we are the first to make the formal connection between the discriminator and the log barrier method in constrained optimization. We provide such novel interpretation for the following reasons: **1)** to make it explicit that we need the manifold constraint to solve such ill-posed problems, **2)** to provide a principled rationale for using an adversarial network (learnable barrier) rather than simply merging the discriminator for reconstruction, **3)** to differentiate the use of the discriminator from that of [45] where the GAN is used "to capture the structural difference of two 3D objects" for feature learning, **4)** to provide a different use of manifold than that of [50] where manifold traversal in the latent space (noise distribution z) of the generators is studied. Rather, we use the manifold in the *discriminator* as a barrier function.

Optimal Learned Penalty Function

However, given a fixed reconstruction function f, the optimum penalty function g cannot discriminate a real object from the reconstruction perfectly if the distribution of the reconstruction $q(\hat{x})$ and the distribution of unlabeled hand-designed or scanned shapes $p(x^\star)$ overlap. In fact, the analysis of the optimal barrier follows that of the discriminator in [17] as the learned penalty function works and trains like a discriminator. Thus, the optimal penalty becomes $g^\star(x) = \frac{p(x)}{p(x)+q(x)}$ where p is the unlabeled 3D shape. Thus, as the reconstruction function generates more realistic shapes, the constraint violation g becomes less important. This behavior works in favor of the reprojection error and the reconstruction function puts more emphasis on minimizing the objective function as the reconstruction gets more realistic.

3.3.3 Raytrace Pooling for Reprojection Error

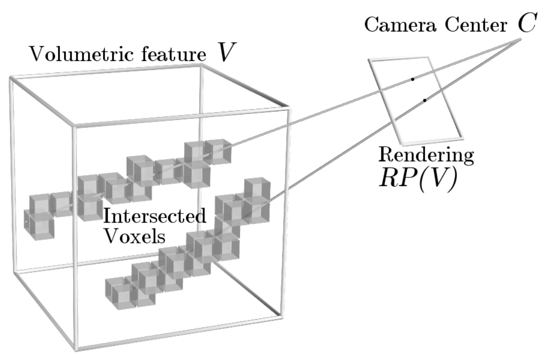

Figure 3.4: Visualization of raytrace pooling. For each pixel of 2D rendering, we calculate the direction of the ray from camera center. Then, we apply pooling function to all hit voxels in 3D grid.

The 2D weak supervisions reside in the image domain whereas the reconstruction is in 3D space. To bridge different domains, we propose a Raytrace Pooling layer (RP-Layer). It takes a 3D volumetric reconstruction x and camera viewpoint c and generates the rendering of the reconstruction x. Here, c consists of the camera center C and camera perspective R. Let a ray emanating from camera center C be $\mathbf{L}_i$ and the intersection of the ray with the image plane be p_i. Then, ray can be parametrized by $u \in \mathbb{R}_+$

$$\mathbf{L}(u) = \mathbf{C} + u\frac{\mathbf{R}^{-1}p - \mathbf{C}}{\|\mathbf{R}^{-1}p - \mathbf{C}\|} \tag{3.6}$$

We aggregate all the voxels v_j that intersect with the ray $\mathbf{L}_i$ using an octree voxel-walking [1] with an efficient ray-box intersection algorithm [43], and compute a single feature for each ray f_i by pooling over the features in the voxels. We visualize the result of the raytracing and aggregated voxels in Fig. 3.4. While multiple types of pooling operations are admissible, we use max pooling in this work. Max pooling along the ray $\mathbf{L}_i$ in an occupancy grid x results in a foreground mask $\tilde{m}$.Finally, we can measure the difference between the predicted foreground mask $\tilde{m} = RP(x, c_j)$ and the ground truth foreground mask m and define a loss $\mathcal{L}_{\text{reproj.}}$:

$$\mathcal{L}_{\text{reproj.}}(x, \mathbf{c}, \mathbf{m}) = \frac{1}{M} \sum_{j}^{M} \mathcal{L}_{\text{s}}(RP(x, c_j), m_j),$$

where M is the number of silhouettes from different viewpoints and c_j is the j-th the camera viewpoint, and $\mathcal{L}_{\text{s}}$ is the mean of per pixel cross-entropy loss. Instead of using raytracing for rendering, a concurrent work in [49] has independently proposed a projection layer based on the Spatial Transformer Network [21]. Since there might be aliasing if the sampling rate is lower than the Nyquist rate [27], the sampling grid from [49] has to be dense and compact. To see the effect of aliasing in sampling-based projection, we compare the performance of [49] and RP-Layer in Sec. 3.4.3. Furthermore, unlike synthetic data where the range of depth is well-controlled, depths of the target objects are unrestricted in real images, which requires dense sampling over a wide range of depth. For our real image reconstruction experiment in Sec. 3.4.4, we determine the range of possible depths over the training data and sample over 512 steps in order to avoid the aliasing effects of [49], far exceeding the 32 steps originally proposed there. On the other hand, RP-Layer mimics the rendering process and does not suffer from aliasing or depth sampling range as it is based on hit-test.

3.3.4 McRecon Optimization

Finally, we return to the original problem of Eq. 3.2 and train the weakly supervised reconstruction functions given by $x = f(\mathbf{I}; \theta_f)$.

$$\underset{\hat{x}:=f(\mathbf{I};\theta_f)}{\text{minimize}} \quad \mathop{\mathbb{E}}_{v \in \text{views}} [\mathcal{L}_{\text{reproj.}}(\hat{x}, m_v)] - \frac{1}{t} \log g(\hat{x}) \tag{3.7}$$

Then we train the log barrier so that it regresses to the ideal constraint function $g(x) = 1$ if $x \in \mathcal{M}$ and 0 otherwise.

$$\underset{g}{\text{minimize}} \quad \mathop{\mathbb{E}}_{x^\star \sim p} \log g(x^\star) + \mathop{\mathbb{E}}_{\hat{x} \sim q} \log(1 - g(\hat{x})) \tag{3.8}$$

$p(x)$ is the probability distribution of the unlabeled 3D shapes and q denotes the probability distribution of reconstruction $q(x|\mathbf{I})$. The final algorithm is in Algo. 1.

While convergence properties of such an optimization problem are nontrivial to prove and an active area of research, our empirical results consistently indicate it behaves reasonably well in

Algorithm 1 McRecon: Training

Require: Datasets: $\mathcal{D}_I = \{(\mathbf{I}_i, m_i, c_i)\}_i$, $\mathcal{D}_S = \{x_i^\star\}_i$

function $\textsc{McRecon}(\mathcal{D}_I, \mathcal{D}_S)$
  while not converged **do**
    for all images $(\mathbf{I}_i, m_i, c_i) \in \mathcal{D}$ **do**
      $\hat{x} \leftarrow f(\mathbf{I}_i)$ // 3D reconstruction
      for all camera $c_{i,j}$, s.t. $j \in \{1, \cdots, M\}$ **do**
        $\tilde{m}_{i,j} \leftarrow RP(x, c_{i,j})$ // Reprojection
      end for
      $g \leftarrow \text{UpdatePenalty}(\hat{x}, x^\star)$
      $\mathbb{E}[\mathcal{L}_{\text{reproj.}}] \leftarrow \frac{1}{M}\sum_{j=1}^{M} \mathcal{L}_s(\tilde{m}_{i,j}, m_{i,j})$
      $\mathcal{L}_f \leftarrow \mathbb{E}[\mathcal{L}_{reproj.}] - \frac{1}{t}\log g(x)$
      $\theta_f \leftarrow \theta_f - \alpha \partial \mathcal{L}_f / \partial \theta_f$
    end for
  end while
  return f
end function

Algorithm 2 Penalty Function Update

Require: Datasets: reconstruction $\hat{x}$ and unlabeled 3D shapes $x^\star$

function $\textsc{UpdatePenalty}(\hat{x}, x^\star)$
  $L_g \leftarrow \frac{1}{|\hat{x}|}\sum_{i \in |\hat{x}|} \log g(\hat{x}_i)$
    $+ \frac{1}{|x^\star|}\sum_{i \in |x^\star|} \log(1 - g(x_i^\star))$
  $\theta_g \leftarrow \theta_g - \alpha \partial \mathcal{L}_g / \partial \theta_g$
  return g
end function

practice.

3.4 Experiments

To validate our approach, we design various experiments and use standard datasets. First, we define the baseline methods including recent works (Sec. 3.4.1) and evaluation metrics (Sec. 3.4.2). To compare our approach with baseline methods in a controlled environment, we use a 3D shape dataset and rendering images. We present quantitative ablation study results on Sec. 3.4.3. Next, we test our framework on a real image single-view and a multi-view dataset in Sec. 3.4.4 and Sec. 3.4.5 respectively. To examine the expressive power of the reconstruction function f, we examine the intermediate representation and analyze its semantic content in Sec 3.4.6 similar to [30, 45]. Note that, we can manipulate the output (shape) using a different modality (image) and allow editing in a different domain.

IOU / AP

Level of supervision	Methods	Transportation		Furniture				Mean
		car	airplane	sofa	chair	table	bench	
1 view 2D	VC [24]	0.2605 / 0.2402	0.1092 / 0.0806	0.2627 / 0.2451	0.2035 / 0.1852	0.1735 / 0.1546	0.1303 / 0.1064	0.1986 / 0.1781
	PTN [49]	0.4437 / 0.7725	0.3352 / 0.5568	0.3309 / 0.4947	0.2241 / 0.3178	0.1977 / 0.2800	0.2145 / 0.2884	0.2931 / 0.4620
	RP	0.3791 / 0.7250	0.2508 / 0.4997	0.3427 / 0.5093	0.1930 / 0.3361	0.1821 / 0.2664	0.2188 / 0.3003	0.2577 / 0.4452
1 view 2D + U3D	RP+NN	0.5451 / 0.5582	0.2057 / 0.1560	0.2767 / 0.2285	0.1556 / 0.1056	0.1285 / 0.0872	0.1758 / 0.1183	0.2597 / 0.2267
	McRecon	**0.5622** / **0.8244**	**0.3727** / **0.5911**	**0.3791** / **0.5597**	**0.3503** / **0.4828**	**0.3532** / **0.4582**	**0.2953** / **0.3912**	**0.4036** / **0.5729**
5 views 2D	VC [24]	0.5784 / 0.5430	0.3452 / 0.2936	0.5257 / 0.4941	0.4048 / 0.3509	0.3549 / 0.3011	0.3387 / 0.2788	0.4336 / 0.3857
	PTN [49]	0.6593 / 0.8504	0.4422 / 0.6721	0.5188 / 0.7180	0.3736 / 0.5081	0.3556 / 0.5367	0.3374 / 0.4725	0.4572 / 0.6409
	RP	0.6521 / **0.8713**	0.4344 / 0.6694	0.5242 / 0.7023	0.3717 / 0.5048	0.3197 / 0.4464	0.321 / 0.4377	0.4442 / 0.6123
5 views 2D + U3D	RP+NN	**0.6744** / 0.6508	**0.4671** / 0.4187	0.5467 / 0.5079	0.3449 / 0.2829	0.3081 / 0.2501	0.3116 / 0.2477	0.4465 / 0.3985
	McRecon	0.6142 / 0.8674	0.4523 / **0.6877**	0.5458 / **0.7473**	**0.4365** / **0.6212**	**0.4204** / **0.5741**	**0.4009** / **0.5770**	**0.4849** / **0.6851**
F3D	R2N2 [6]	0.8338 / 0.9631	0.5425 / 0.7747	0.6784 / 0.8582	0.5174 / 0.7266	0.5589 / 0.7754	0.4950 / 0.6982	0.6210 / 0.8123

Table 3.1: Per-category 3D reconstruction Intersection-over-Union(IOU) / Average Precision(AP). Please see Sec. 3.4.1 for details of baseline methods and the level of supervision. McRecon outperforms other baselines by larger margin in classes with more complicated shapes as shown in Fig. 3.5.

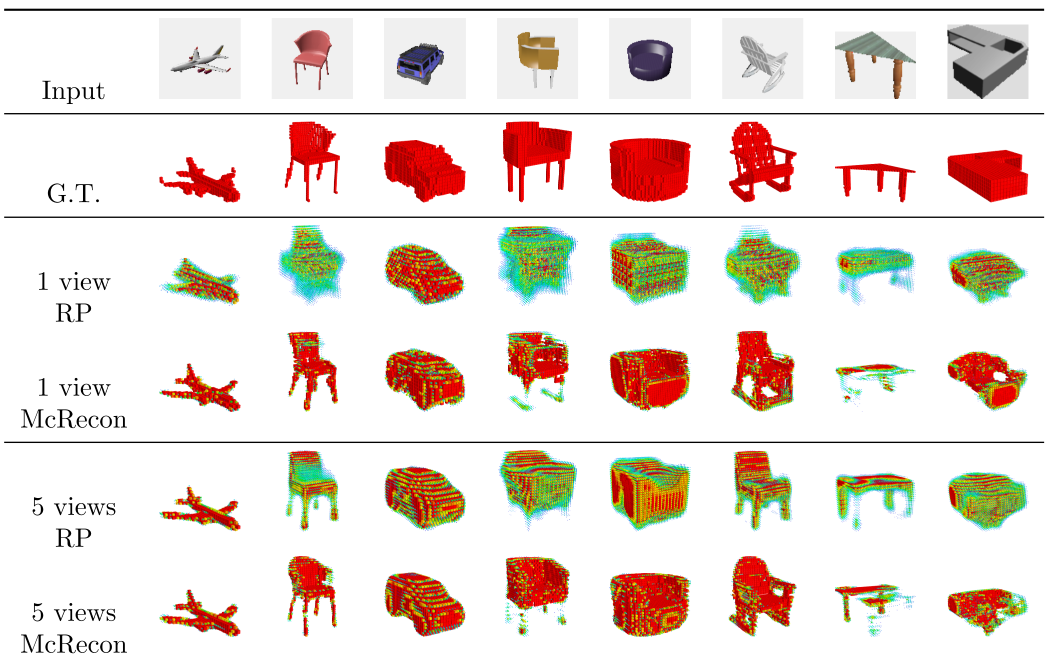

Figure 3.5: Qualitative results of single- or multi-view synthetic image reconstructions on ShapeNet dataset. Compared to RP which only uses 2D weak supervision, McRecon reconstructs complex shapes better. Please refer to Sec. 3.4.2 for details of our visualization method.

3.4.1 Baselines

For an accurate ablation study, we propose various baselines to examine each component in isolation. First, we categorize all the baseline methods into three categories based on the level of supervision: *2D Weak Supervision (2D)*, *2D Weak Supervision + unlabeled 3D Supervision (2D + U3D)*, and *Full 3D Supervision (F3D)*. *2D* has access to 2D silhouettes and viewpoints as supervision; and *2D + U3D* uses silhouettes, viewpoints, and unlabeled 3D shapes for supervision. Finally, *F3D* is supervised with the ground truth 3D reconstruction associated with the images. Given *F3D* supervision, silhouettes do not add any information, thus the performance of a system with full supervision provides an approximate performance upper bound.

Specifically, in the *2D* case, we use Raytrace Pooling (RP) as proposed in Sec .3.3.3 and compare it with Perspective Transformer (PTN) by Yan *et al.* [49]. Next, in the *2D + U3D* case, we use RP + Nearest Neighbor (RP+NN) and McRecon. RP + NN uses unlabeled 3D shapes, by retrieving the 3D shape that is closest to the prediction. Finally, in the *F3D* case, we use R2N2 [6]. We did not include [45, 15] in this experiment since they are restricted to single-view reconstruction and use full 3D supervision which would only provide an additional upper bound. For all neural network based baselines, we used the same base network architecture (encoder and generator) to ascribe performance gain only to the supervision mode. Aside from learning-based methods, we also provide a lower-bound on performance using voxel carving (VC) [24]. We note that voxel carving requires silhouette and camera viewpoint during testing. Kindly refer to the supplementary material for details of baseline methods, implementation, and training.

3.4.2 Metrics and Visualization

The network generates a voxelized reconstruction, and for each voxel, we have occupancy probability (confidence). We use Average Precision (AP) to evaluate the quality and the confidence of the reconstruction. We also binarize the probability and report Intersection-over-Union (IOU) with threshold 0.4, following [6]. This metric gives more accurate evaluation of deterministic methods like voxel carving. For visualization, we use red to indicate voxels with occupancy probability above 0.6 and gradually make it smaller and green until occupancy probability reaches 0.1. When the probability is below 0.1, we did not visualize the voxel.

3.4.3 Ablation Study on ShapeNet [5]

In this section, we perform ablation study and compare McRecon with the baseline methods on the ShapeNet [5] dataset. The synthetic dataset allows us to control external factors such as the number of viewpoints, quality of mask and is ideal for ablation study. Specifically, we use the renderings from [6] since it contains a large number of images from various viewpoints and the camera model has more degree of freedom. In order to train the network on multiple categories while maintaining a semantically meaningful manifold across different classes, we divide the categories into furniture (sofa, chair, bench, table) and vehicles (car, airplane) classes and trained networks separately. We use the alpha channel of the renderings image to generate 2D mask supervisions (finite depth to indicate foreground silhouette). For the unlabeled 3D shapes, we simply voxelized the 3D shapes. To simulate realistic scenario, we divide the dataset into three **disjoint** sets: shapes for 2D weak supervision, shapes for unlabeled 3D shapes, and the test set. Next, we study the impact of the level of supervision, the number of viewpoints, and the object category on the performance.

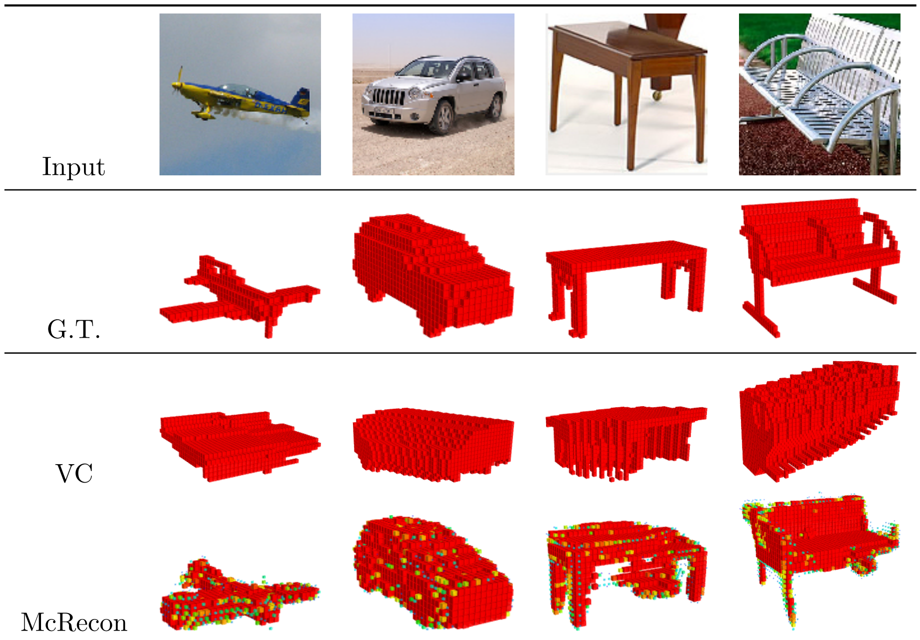

Figure 3.6: Real image single-view reconstructions on ObjectNet3D. Compared to RP which only uses 2D weak supervision, McRecon reconstructs complex shapes better. Please refer to Sec. 3.4.2 for details of our visualization method.

Figure 3.7: Qualitative results of multi-view real image reconstructions on Stanford Online Product dataset [38]. Our network successfully reconstructed real images coordinating different views.

First, we found that more supervision leads to better reconstruction and McRecon make use of the unlabeled 3D shapes effectively (Vertical axis of Tab. 3.1). Compare with the simple nearest

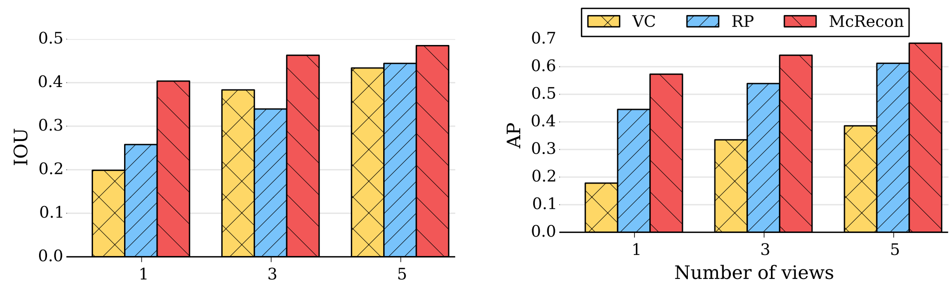

Figure 3.8: Intersection-over-union (IOU) and Average Precision (AP) over the number of masks used for weak supervision. The performance gap between McRecon and the other baselines gets larger as the number of views of masks decreases (i.e. supervision strength gets weaker).

	sofa	chair	table	bench	mean
PTN_16 [49]	0.4753	0.2888	0.2476	0.2576	0.2979
PTN_32 [49]	0.4947	0.3178	0.2800	0.2884	0.3283
PTN_64 [49]	0.5082	0.3377	0.3114	0.3104	0.3509
PTN_128 [49]	0.5217	0.3424	0.3104	0.3146	0.3545
RP	0.5093	0.3361	0.2664	0.3003	0.3308

Table 3.2: AP of 2D weak supervision methods on single-view furniture reconstruction. In order to analyze the effect of aliasing of PTN [49], we varied its disparity sampling density (sampling density N, for all PTN_N) and compare with RP.

neighbor, which also make use of the unlabeled 3D data, McRecon outperforms the simple baseline by a large margin. This hints that the barrier function smoothly interpolates the manifold of 3D shapes and provide strong guidance. Second, McRecon learns to generate better reconstruction even from a small number of 2D weak supervision. In Tab. 3.1 and in Fig. 3.8, we vary the number of 2D silhouettes that we used to train the networks and observe that the performance improvement that we get from exploiting the unlabeled 3D shapes gets larger as we use a fewer number of 2D supervision. Third, we observed that McRecon outperforms other baselines by a larger margin on classes with more complicated shapes such as chair, bench, and table which have concavity that is difficult to recover only using 2D silhouettes. For categories with simpler shapes such as car, the marginal benefit of using the adversarial network is smaller. Similarly, 3D nearest neighbor retrieval improves reconstruction quality only on few categories of a simple shape such as car while it also harms the reconstruction on complex shapes such as chair or table. This is expected since their 3D shapes are close to convex shapes and 2D supervision is enough to recover 3D shapes.

We visualize the reconstructions in Fig. 3.5. We observe that our network can carve concavities, which is difficult to learn solely from mask supervision and demonstrates a qualitative benefit of our manifold constraint. Also, compared to the network trained only using mask supervision, McRecon prefers to binarize the occupancy probability, which seems to be an artifact of the generator fooling

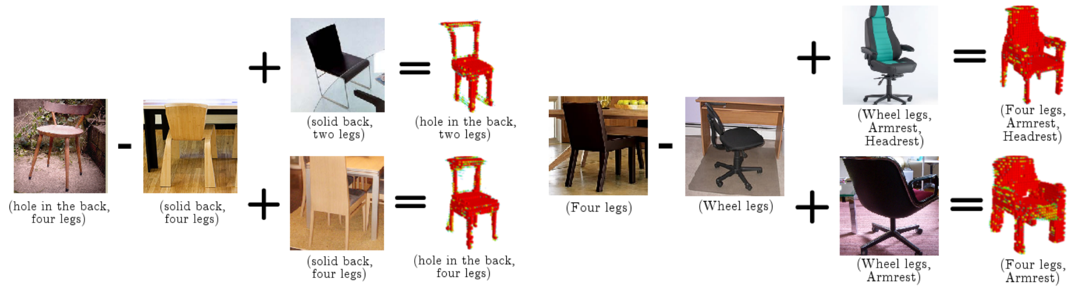

Figure 3.9: Arithmetic on latent variable z of different images. By subtracting latent variables of similar chairs with different properties, we extracted the feature which represents such property. We applied the feature to two other chairs to demonstrate that this is a generic and replicable representation.

the discriminator.

Raytracing Comparison In this section, we compare a raytracing based projection (RP-Layer) and a sampling based projection (PTN [49]) experimentally on ShapeNet single view furniture category. We only vary the projection method and sampling rate along depth but keep the same base network architecture. As shown in Table. 3.2, the reconstruction performance improves as the sampling rate increases as expected in Sec. 3.3.3. We suspect that the trilinear interpolation in PTN played a significant role after it reaches resolution 64 and that implementing a similar scheme using ray length in RP-Layer could potentially improve the result.

3.4.4 Single-view reconst. on ObjectNet3D [47]

In this experiment, we train our network for single real-image reconstruction using the ObjectNet3D [47] dataset. The dataset contains 3D annotations in the form of the closest 3D shape from ShapeNet and viewpoint alignment. Thus, we generate 2D silhouettes using 3D shapes. We split the dataset using the shape index to generate disjoint sets like the previous experiments. Since the dataset consists of at most 1,000 instances per category, we freeze the generator and discriminator and fine-tune only the 2D encoder $E(u)$. We quantitatively evaluate intersection-over-union(IOU) on the reconstruction results as shown in Table 3.3. The numbers indicate that McRecon has better generalization power beyond the issue of ill-conditioned visual hull reconstruction and silhouette-based learning [49] from a single-view mask. Please note that voxel carving, unlike McRecon, requires camera parameters at test time. Qualitative results are presented in Fig. 3.6.

Training with noisy viewpoint estimation In this experiment, we do a noisy estimation of camera parameters instead of using the ground-truth label as an input to RP, training the network only using 2D silhouette. We estimate camera parameters by discretizing azimuth, elevation, and

	sofa	chair	bench	car	airplane
VC [24]	0.304	0.177	0.146	0.481	0.151
PTN [49]	0.276	0.151	0.095	0.421	0.130
McRecon	**0.423**	**0.380**	**0.380**	**0.649**	**0.322**
PTN-NV [49]	0.207	0.128	0.068	0.344	0.100
McRecon-NV	**0.256**	**0.157**	**0.086**	**0.488**	**0.214**

Table 3.3: Per-class real image 3D reconstruction intersection-over-union(IOU) percentage on ObjectNet3D. NV denotes a network trained with noisy viewpoint estimation.

depth of the camera into 10 bins and finding the combination of parameters that minimize the L_2 distance of the rendering of a roughly aligned 3D model [47] with the ground-truth 2D silhouette. We quantitatively evaluate intersection-over-union(IOU) on the reconstruction results as shown in Table 3.3. These results demonstrate that McRecon has stronger generalization ability even with noisy viewpoint labels, deriving benefit from the manifold constraint.

3.4.5 Multi-view Reconst. on OnlineProduct [38]

Stanford Online Product [38] is a large-scale multiview dataset consisting of images of products from e-commerce websites. In this experiment, we test McRecon on multi-view real images using the network trained on the ShapeNet [5] dataset with random background images from PASCAL [11] to make the network robust to the background noise. We visualize the results in Fig. 3.7. The result shows that our network can integrate information across multiple views of real images and reconstruct a reasonable 3D shape.

3.4.6 Representation analysis

In this experiment, we explore the semantic expressiveness of intermediate representation of the reconstruction function f. Specifically, we use the intermediate representation in the recurrent neural network, which we denote as z, as the aggregation of multi-view observations. We use the interpolation and vector arithmetic similar to [10, 30] in the representation space of z. However, unlike the above approaches, we use different modalities for the input and output which are images and 3D shapes respectively. Therefore, we can make high-level manipulation of the representation z from 2D images and modify the output 3D shape.

3.5 Conclusion

We proposed Weakly supervised 3D Reconstruction with Adversarial Constraint (McRecon), a novel framework that makes use of foreground masks for 3D reconstruction by constraining the reconstruction to be in the space of unlabeled real 3D shapes. Additionally, we proposed a raytrace pooling layer to bridge the representation gap between 2D masks and 3D volumes. We analyzed each component of the model through an ablation study on synthetic images. McRecon can successfully generate a high-quality reconstruction from weak 2D supervision, with reconstruction accuracy comparable to prior works that use full 3D supervision. Furthermore, we demonstrated that our model has strong generalization power for single-view real image reconstruction with noisy viewpoint estimation, hinting at better practical utility.

Bibliography

[1] James Arvo. Linear-time voxel walking for octrees. *Ray Tracing News*, 1(2), 1988.

[2] Moez Baccouche, Franck Mamalet, Christian Wolf, Christophe Garcia, and Atilla Baskurt. Sequential deep learning for human action recognition. In *Proceedings of the Second International Conference on Human Behavior Unterstanding*. Springer-Verlag, 2011.

[3] Y Bao, M Chandraker, Y Lin, and S Savarese. Dense object reconstruction using semantic priors. In *2015 IEEE Conference on Computer Vision and Pattern Recognition*, 2013.

[4] Stephen Boyd and Lieven Vandenberghe. *Convex Optimization*. Cambridge University Press, New York, NY, USA, 2004.

[5] Angel X. Chang, Thomas Funkhouser, Leonidas Guibas, Pat Hanrahan, Qixing Huang, Zimo Li, Silvio Savarese, Manolis Savva, Shuran Song, Hao Su, Jianxiong Xiao, Li Yi, and Fisher Yu. ShapeNet: An Information-Rich 3D Model Repository. Technical Report arXiv:1512.03012 [cs.GR], 2015.

[6] 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 *Proceedings of the European Conference on Computer Vision (ECCV)*, 2016.

[7] Angela Dai, Angel X. Chang, Manolis Savva, Maciej Halber, Thomas Funkhouser, and Matthias Nießner. Scannet: Richly-annotated 3d reconstructions of indoor scenes. *arXiv preprint arXiv:1702.04405*, 2017.

[8] Amaury Dame, Victor A Prisacariu, Carl Y Ren, and Ian Reid. Dense reconstruction using 3d object shape priors. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1288–1295, 2013.

[9] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. ImageNet: A Large-Scale Hierarchical Image Database. In *CVPR09*, 2009.

[10] Alexey Dosovitskiy, Jost Tobias Springenberg, and Thomas Brox. Learning to generate chairs with convolutional neural networks. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2015.

[11] M. Everingham, S. M. A. Eslami, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman. The pascal visual object classes challenge: A retrospective. *International Journal of Computer Vision*, 111(1):98–136, Jan. 2015.

[12] Jorge Fuentes-Pacheco, José Ruiz-Ascencio, and Juan Manuel Rendón-Mancha. Visual simultaneous localization and mapping: a survey. *Artificial Intelligence Review*, 43, 2015.

[13] Y. Furukawa, B. Curless, S.M. Seitz, and R. Szeliski. Towards internet-scale multi-view stereo. In *CVPR*, pages 1434–1441, 2010.

[14] Yasutaka Furukawa and Jean Ponce. Accurate, dense and robust multiview stereopsis. *PAMI*, 32(8):1362–1376, 2010.

[15] R. Girdhar, D.F. Fouhey, M. Rodriguez, and A. Gupta. Learning a predictable and generative vector representation for objects. In *ECCV*, 2016.

[16] Michael Goesele, Jens Ackermann, Simon Fuhrmann, Ronny Klowsky, Fabian Langguth, Patrick Müandcke, and Martin Ritz. Scene reconstruction from community photo collections. *IEEE Computer*, 43:48–53, 2010.

[17] 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*, 2014.

[18] Klaus Häming and Gabriele Peters. The structure-from-motion reconstruction pipeline–a survey with focus on short image sequences. *Kybernetika*, 46(5):926–937, 2010.

[19] Carlos Hernández and George Vogiatzis. Shape from photographs: A multi-view stereo pipeline. In *Computer Vision*, volume 285 of *Studies in Computational Intelligence*, pages 281–311. Springer, 2010.

[20] Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A Efros. Image-to-image translation with conditional adversarial networks. *arxiv*, 2016.

[21] Max Jaderberg, Karen Simonyan, Andrew Zisserman, and koray kavukcuoglu. Spatial transformer networks. In C. Cortes, N. D. Lawrence, D. D. Lee, M. Sugiyama, and R. Garnett, editors, *Advances in Neural Information Processing Systems 28*, pages 2017–2025. Curran Associates, Inc., 2015.

[22] S. Ji, W. Xu, M. Yang, and K. Yu. 3d convolutional neural networks for human action recognition. In *ICML*, 2010.

[23] Abhishek Kar, Shubham Tulsiani, Joao Carreira, and Jitendra Malik. Category-specific object reconstruction from a single image. In *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1966–1974. IEEE, 2015.

[24] Kiriakos N. Kutulakos and Steven M. Seitz. A theory of shape by space carving. *International Journal of Computer Vision*, 38(3):199–218, 2000.

[25] D. Maturana and S. Scherer. VoxNet: A 3D Convolutional Neural Network for Real-Time Object Recognition. In *IROS*, 2015.

[26] Wojciech Matusik, Chris Buehler, Ramesh Raskar, Steven J Gortler, and Leonard McMillan. Image-based visual hulls. In *Proceedings of the 27th annual conference on Computer graphics and interactive techniques*, pages 369–374. ACM Press/Addison-Wesley Publishing Co., 2000.

[27] Alan V. Oppenheim and Ronald W. Schafer. *Discrete-Time Signal Processing*. Prentice Hall Press, Upper Saddle River, NJ, USA, 3rd edition, 2009.

[28] Yingwei Pan, Tao Mei, Ting Yao, Houqiang Li, and Yong Rui. Jointly modeling embedding and translation to bridge video and language. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016.

[29] Charles R Qi, Hao Su, Matthias Niessner, Angela Dai, Mengyuan Yan, and Leonidas J Guibas. Volumetric and multi-view cnns for object classification on 3d data. *arXiv preprint arXiv:1604.03265*, 2016.

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

[31] Zhile Ren and Erik B. Sudderth. Three-dimensional object detection and layout prediction using clouds of oriented gradients. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016.

[32] G. Riegler, A. Osman Ulusoy, and A. Geiger. OctNet: Learning Deep 3D Representations at High Resolutions. *ArXiv e-prints*, Nov. 2016.

[33] Jason Rock, Tanmay Gupta, Justin Thorsen, JunYoung Gwak, Daeyun Shin, and Derek Hoiem. Completing 3d object shape from one depth image. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2484–2493, 2015.

[34] Silvio Savarese, Marco Andreetto, Holly Rushmeier, Fausto Bernardini, and Pietro Perona. 3d reconstruction by shadow carving: Theory and practical evaluation. *International Journal of Computer Vision*, 71(3):305–336, 2007.

[35] Nikolay Savinov, Christian Häne, Marc Pollefeys, et al. Discrete optimization of ray potentials for semantic 3d reconstruction. In *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5511–5518. IEEE, 2015.

[36] Steven M Seitz and Charles R Dyer. Photorealistic scene reconstruction by voxel coloring. *International Journal of Computer Vision*, 35(2):151–173, 1999.

[37] Ashish Shrivastava, Tomas Pfister, Oncel Tuzel, Josh Susskind, Wenda Wang, and Russ Webb. Learning from simulated and unsupervised images through adversarial training. *arXiv preprint arXiv:1612.07828*, 2016.

[38] Hyun Oh Song, Yu Xiang, Stefanie Jegelka, and Silvio Savarese. Deep metric learning via lifted structured feature embedding. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[39] Shuran Song and Jianxiong Xiao. Deep Sliding Shapes for amodal 3D object detection in RGB-D images. In *CVPR*, 2016.

[40] Du Tran, Lubomir Bourdev, Rob Fergus, Lorenzo Torresani, and Manohar Paluri. Learning spatiotemporal features with 3d convolutional networks. In *Proceedings of the 2015 IEEE International Conference on Computer Vision (ICCV)*, ICCV '15. IEEE Computer Society, 2015.

[41] Du Tran, Lubomir Bourdev, Rob Fergus, Lorenzo Torresani, and Manohar Paluri. Deep end2end voxel2voxel prediction. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, June 2016.

[42] Shubham Tulsiani, Tinghui Zhou, Alexei A Efros, and Jitendra Malik. Multi-view supervision for single-view reconstruction via differentiable ray consistency. *arXiv preprint arXiv:1704.06254*, 2017.

[43] Amy Williams, Steve Barrus, R Keith Morley, and Peter Shirley. An efficient and robust ray-box intersection algorithm. In *ACM SIGGRAPH 2005 Courses*, page 9. ACM, 2005.

[44] J. Wu, T. Xue, J. J. Lim, Y. Tian, J. B. Tenenbaum, A. Torralba, and W. T. Freeman. Single Image 3D Interpreter Network. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2016.

[45] Jiajun Wu, Chengkai Zhang, Tianfan Xue, William T Freeman, and Joshua B Tenenbaum. Learning a probabilistic latent space of object shapes via 3d generative-adversarial modeling. In *Neural Information Processing Systems (NIPS)*, 2016.

[46] Z. Wu, S. Song, A. Khosla, F. Yu, L. Zhang, X. Tang, and J. Xiao. 3d shapenets: A deep representation for volumetric shapes. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015.

[47] Yu Xiang, Wonhui Kim, Wei Chen, Jingwei Ji, Christopher Choy, Hao Su, Roozbeh Mottaghi, Leonidas Guibas, and Silvio Savarese. Objectnet3d: A large scale database for 3d object recognition. In *European Conference on Computer Vision*, pages 160–176. Springer, 2016.

[48] Yu Xiang, Roozbeh Mottaghi, and Silvio Savarese. Beyond pascal: A benchmark for 3d object detection in the wild. In *IEEE Winter Conference on Applications of Computer Vision*, pages 75–82. IEEE, 2014.

[49] Xinchen Yan, Jimei Yang, Ersin Yumer, Yijie Guo, and Honglak Lee. Learning volumetric 3d object reconstruction from single-view with projective transformations. In *Advances in Neural Information Processing Systems*, 2016.

[50] Jun-Yan Zhu, Philipp Krähenbühl, Eli Shechtman, and Alexei A. Efros. Generative visual manipulation on the natural image manifold. In *Proceedings of European Conference on Computer Vision (ECCV)*, 2016.