Synthesizing 3D Shapes from Silhouette Image Collections using Multi-projection Generative Adversarial Networks

Xiao Li
University of Science and Technology of China

Yue Dong
Microsoft Research Asia

Pieter Peers
College of William & Mary

Xin Tong
Microsoft Research Asia

Abstract

We present a new weakly supervised learning-based method for generating novel category-specific 3D shapes from unoccluded image collections. Our method is weakly supervised and only requires silhouette annotations from unoccluded, category-specific objects. Our method does not require access to the object's 3D shape, multiple observations per object from different views, intra-image pixel-correspondences, or any view annotations. Key to our method is a novel multi-projection generative adversarial network (MP-GAN) that trains a 3D shape generator to be consistent with multiple 2D projections of the 3D shapes, and without direct access to these 3D shapes. This is achieved through multiple discriminators that encode the distribution of 2D projections of the 3D shapes seen from a different views. Additionally, to determine the view information for each silhouette image, we also train a view prediction network on visualizations of 3D shapes synthesized by the generator. We iteratively alternate between training the generator and training the view prediction network. We validate our multi-projection GAN on both synthetic and real image datasets. Furthermore, we also show that multi-projection GANs can aid in learning other high-dimensional distributions from lower dimensional training datasets, such as material-class specific spatially varying reflectance properties from images.

1. Introduction

Learning to synthesize novel 3D shapes from a class of objects is a challenging problem with many applications in computer vision, ranging from single image 3D reconstruction [6, 9, 25, 32, 37], to shape completion [36], to 3D shape analysis [34]. Typically, such methods are trained on reference 3D shapes, or on images from different viewpoints with labeled pixel-correspondences and/or with viewpoint information. Creating such training sets for new object categories is labor-intensive and cumbersome. In this paper we propose a novel weakly supervised method for learning to generate 3D shapes from unoccluded silhouette image-collections without relying on reference 3D shapes, correspondences, or view annotations.

A key challenge in learning to synthesize 3D shapes from image-collections gathered from uncontrolled sources is that we cannot count on having multiple observations of the same object from different viewpoints; we only have samples from the distributions of images of the objects from different views. We overcome this practical problem using a novel generative adversarial network (GAN) architecture that learns the high dimensional distribution of 3D shapes from multiple independently sampled low dimensional distributions of silhouette images of the objects. The relation between each low dimensional distribution of silhouette images and the high dimensional 3D shapes is characterized by a *"projection"*, and the resulting *multi-projection GAN* architecture employs, for each projection, a different discriminator that encodes the characteristics of the corresponding projection's silhouette image distribution. Intuitively, the proposed multi-projection GAN learns the distribution of 3D shapes for which the visualizations for each viewpoint matches the corresponding (independent) distributions of the training silhouette images for that same view.

A second key challenge is that the viewpoint information is often not available for the training images that are gathered "in the wild". To address this challenge, we utilize a view prediction network for inferring the view information for each of the training images. Ideally, the view prediction network should be trained by the types of 3D shapes synthesized by our generator. However, this creates a circular dependency, because the 3D shape generator needs the view prediction network to train from silhouette images. We resolve this dilemma by jointly training both networks in an iterative alternating fashion, resulting in a 3D shape learning pipeline that is robust to non-uniform viewpoint distributions over the training data (i.e., we make no assumptions on the viewpoint distribution).

Our method facilitates training of a 3D shape generator from weakly supervised real-world image collections

with silhouette annotations, thereby greatly reducing the cost for learning a 3D shape generator for new object categories. We demonstrate the strengths of our multi-projection GAN with a jointly trained view prediction network on both synthetic datasets and real-world silhouette image-collections. Furthermore, we demonstrate that multi-projection GANs can be generalized to aid in learning other types of high-dimensional (non-image) distributions from low-dimensional image observations such as learning the distribution of spatially-varying material reflectance parameters.

2. Related Work

In the past few years, various deep learning based methods have been proposed for 3D shape generation. These methods can be roughly categorized in three classes. A first class of methods relies on a large set of reference 3D shapes for training (e.g., for generating 3D voxel shapes with a VAE-GAN [34], for shape completion [36], or for 3D reconstruction from a 2D image [6, 9, 11, 16, 25, 32, 37]). Of particular note is the method of Zhu *et al*. [37] that jointly reconstructs a single 3D voxel shape and camera pose. A second class of methods foregoes the need for reference 3D shapes, and uses annotated images with correspondences [18] or exploit consistency over different views of objects [28, 29, 35]. Of particular note is the method of Tulsiani *et al*. [28] who predict both shape and viewpoint. Closest related to the proposed method is a third class of methods that learns the 3D shape distribution from unannotated images. Gadelha *et al*. [10] train a voxel GAN using a single discriminator with 2D silhouette images from predefined discrete view distributions. Henderson *et al*. [14] utilize a variational auto-encoder (VAE) for generating 3D meshes from unannotated images with uniform distributed viewpoints. In contrast to Gadelha *et al*., we use multiple discriminators for each of the different views, resulting in a better generation quality. Unlike Henderson *et al*., we build on GANs as opposed the VAEs. More importantly, we do not require a uniform viewpoint distribution over the training dataset, which is difficult to enforce in internet-mined datasets. As we will show in section 5, for similar shape representations, the proposed method produces higher quality results compared to both prior methods.

Recently, Bora *et al*. [4] generalized the method of Gadelha *et al*., and showed that a generative model (for images) can be trained from different types of lossy projections. However, Bora *et al*. use a single discriminator, resulting in a suboptimal synthesis for projections with different distributions. To handle multiple projection distributions, our method relies on multiple discriminators, i.e., one for each projection. We are not the first to consider combining multiple discriminators to train GANs [8, 23, 17]. However, all these prior methods require access to the high dimensional training data, and focus on improving either the efficiency or the stability of the training process. In contrast, we do not have access to the high-dimensional distributions (i.e., 3D shapes), and train it directly from the lower-dimensional projections (i.e., images).

3. Shape Distributions from Silhouette Images

We aim to learn a 3D shape distribution $\mathbf{X}$, in the form of a generator, for a class of objects from a collection of uncorresponded silhouette images that follows a distribution $\mathbf{Y}$. Our solution (Figure 1) consists of two key components: a novel multi-projection GAN (subsection 3.1) that learns the 3D distribution $\mathbf{X}$ using multiple discriminators that ensure that the corresponding 2D "projections" from the 3D distribution follow the distribution of the training silhouette images, and a view prediction network (subsection 3.2) for estimating the viewpoints of the input 2D silhouette training images. The training of both networks depends on the availability of the other. We will therefore first introduce multi-projection GANs assuming that estimates of the viewpoints are available. Next, we introduce our view prediction network and the joint training strategy for training both networks iteratively.

3.1. Multi-Projection GAN

GAN overview A Generative Adversarial Network (GAN) [2, 12, 24] consists of a generator network $\mathbf{G}$ and a discriminator network $\mathbf{D}$. The generator $\mathbf{G}$ takes as input a vector of latent variables z sampled from uniformly distributed noise $\mathbf{Z}$ and generates samples from a learned distribution $\mathbf{X}$. The discriminator $\mathbf{D}$ judges whether a sample belongs to the distribution $\mathbf{X}$ or not. Training both networks is performed in competition, until the discriminator cannot distinguish the generated samples from the data distribution. The loss for the discriminator $\mathbf{D}$ is defined as:

$$\mathbf{L_D}(\mathbf{X},\mathbf{Z}) = \sum_{x\sim\mathbf{X}} \log(\mathbf{D}(x)) + \sum_{z\sim\mathbf{Z}} \log(1-\mathbf{D}(\mathbf{G}(z))), \tag{1}$$

and the loss for the generator is defined as:

$$\mathbf{L_G}(\mathbf{Z}) = \sum_{z\sim\mathbf{Z}} \log(\mathbf{D}(\mathbf{G}(z))). \tag{2}$$

In the case of 3D shape generation, we represent each object in $\mathbf{X}$ by a binary 3D voxel grid.

Projection In order to train a 3D shape GAN, a large collection of exemplars from the 3D shape distribution $\mathbf{X}$ are needed. However, such collections only exist for a few object classes. In our case, we only have silhouette images of the class of objects that follow a distribution $\mathbf{Y}$. The dimensionality of $\mathbf{Y}$ (2D images) is lower than the dimensionality of $\mathbf{X}$ (3D voxel grid). We assume that the relation

between the low dimensional distribution $\mathbf{Y}$ and the high dimensional distribution $\mathbf{X}$ can be modeled by a differentiable (potentially non-linear) *projection*:

$$\mathbf{Y} = \mathbf{P}(\mathbf{X}, \mathbf{\Phi}), \tag{3}$$

where $\mathbf{\Phi}$ are the latent parameters of the projection (e.g., the intrinsic and extrinsic camera parameters), which model any external factors unrelated to the target distribution $\mathbf{X}$.

For learning 3D shape distributions, the projection $\mathbf{P}$ generates a silhouette image from a 3D voxel shape. Practically, we follow the method of Tulsiani *et al*. [29] exactly: Given a 3D voxel shape and viewpoint, we first compute a ray intersection probability for each voxel using ray-casting. Next, the silhouette is computed as the expected value of the intersection probability along the z axis.

Multi-projection GAN Generally, unless the high dimensional distribution $\mathbf{X}$ occupies a low dimensional embedding, a single low dimensional projection will incur a loss of information, and thus cannot unambiguously determine the target distribution $\mathbf{X}$. For example, without additional priors, we cannot reconstruct the distribution of 3D shapes from silhouettes from a single view. Even when considering multiple views (modeled through the latent parameters of the projection $\mathbf{\Phi}$), a single discriminator is unlikely to be able to model the joint distribution over all views with sufficient accuracy. We therefore consider multiple projections $\mathbf{P}_i$ with corresponding projected sample distributions $\mathbf{Y}_i$ and associated discriminators $\mathbf{D}_i$. We define the combined loss function for each discriminator as:

$$\mathbf{L}_{\mathbf{D}_i}(\mathbf{Y}_i, \mathbf{Z}) = \sum_{y \sim \mathbf{Y}_i} \log(\mathbf{D}_i(y)) + \tag{4}$$

$$\sum_{z \sim \mathbf{Z}, \varphi \sim \mathbf{\Phi}_i} \log(1 - \mathbf{D}_i(\mathbf{P}_i(\mathbf{G}(z), \varphi))). \tag{5}$$

Similarly, we define the loss for the generator as:

$$\mathbf{L}_{\mathbf{G}}(\mathbf{Z}) = \sum_i \sum_{z \sim \mathbf{Z}, \varphi \sim \mathbf{\Phi}_i} \log(\mathbf{D}_i(\mathbf{P}_i(\mathbf{G}(z), \varphi))), \tag{6}$$

where we randomly draw the latent variables z from $\mathbf{Z}$, and the latent projection parameters φ from a distribution $\mathbf{\Phi}_i$. In the case of 3D shape generation from silhouette images, the distribution $\mathbf{\Phi}_i$ encompasses the set of viewpoints that project to similar silhouettes (i.e., to account for errors in the view calibration). Figure 1 (a) overviews our 3D shape generation multi-projection GAN, named *MP-GAN*.

Independence of $\mathbf{Y}_i$ An advantage of our multi-projection approach is that the loss function for the i-th discriminator (Equation 5) only depends on samples drawn from $\mathbf{Y}_i$. Hence, each discriminator can be trained from independently draw samples from each $\mathbf{Y}_i$. Consequently, the samples for the different discriminators do not need to correspond to the same objects or latent projection parameters. The choice of which samples to draw for the training for each discriminator does not affect the loss function of the generator either. From a practical perspective, this allows us to train a 3D shape generator with image collections of different objects from different viewpoints, without requiring any explicit correspondences.

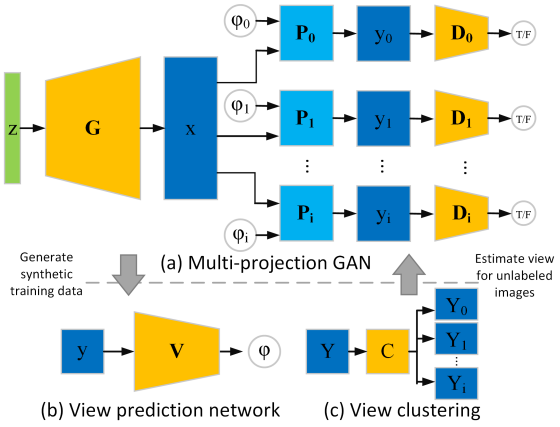

Figure 1. **(a)** A multi-projection GAN consists of a generator $\mathbf{G}$ that takes as input a vector of latent variables z sampled from a uniform distribution $\mathbf{Z}$, and generates samples from the target distribution $\mathbf{X}$. The generator $\mathbf{G}$ is trained in competition with multiple discriminators $\mathbf{D}_i$ that assess if the projection $\mathbf{P}_i$ of a sample belongs to the projected target distribution $\mathbf{Y}_i$. The projection $\mathbf{P}_i$ relates a high dimensional sample from $\mathbf{X}$ (e.g., voxel shape) to a lower-dimensional sample $\mathbf{Y}_i$ (e.g., silhouette image), and which can feature its own latent parameters (φ_i) to model parameters independent of $\mathbf{X}$ (e.g., viewpoint variations). **(b)** The view prediction network $\mathbf{V}$ estimates the viewpoints φ from a silhouette image $\mathbf{Y}$, that are subsequently clustered and merged, **(c)** and assigned to the training set of the corresponding discriminator. We iteratively alternate between training both networks.

Distinct Discriminator Requirement We also observe that a good choice of projections for training multi-projection GANs are those whose projected data distributions are significantly different. Indeed, if two projections $\mathbf{P}_a(\mathbf{X}, \mathbf{\Phi}_a)$ and $\mathbf{P}_b(\mathbf{X}, \mathbf{\Phi}_b)$ are similar to each other, then the corresponding discriminators will learn the same distribution. Hence, according to Equation 5, the two discriminators will similarly affect the training of the generator. Therefore, the two similar projections should act as a single projection:

$$\mathbf{P}(\mathbf{X}, \mathbf{\Phi}) = \mathbf{P}_a(\mathbf{X}, \mathbf{\Phi}_a) \cup \mathbf{P}_b(\mathbf{X}, \mathbf{\Phi}_b), \quad \mathbf{\Phi} = \mathbf{\Phi}_a \cup \mathbf{\Phi}_b. \tag{7}$$

Concretely, for 3D shape modeling, a large difference in viewpoint can yield a significantly different silhouette image distribution. Hence, different discriminators are needed

for large viewpoint differences. In contrast, small viewpoint changes produce very similar silhouette distributions, and thus we can combine discriminators of similar views together, and model the viewpoint variations by the latent projection parameters $\mathbf{\Phi}$. As a result, a multi-projection GAN for 3D shapes does not require perfect viewpoint estimates; we will exploit this property when training the generator and viewpoint predictor (subsection 3.2).

3.2. View Prediction and Clustering

The multiple discriminators $\mathbf{D}_i$ and projections $\mathbf{P}_i$ in *MP-GAN* require knowledge about which silhouette images belong to which distribution $\mathbf{Y}_i$ as well as the corresponding viewpoints modeled by the latent projection parameters $\varphi \sim \mathbf{\Phi}_i$. Prior work in deep viewpoint estimation relied on labeled training data or synthetically rendered images from known 3D shape collections [20, 26, 33], or on multi-view correspondences [28]. None of these methods are directly applicable to our input training dataset.

To estimate the viewpoints, we utilize a view prediction network trained on a large number of reference silhouette images (with viewpoint) obtained by projecting 3D shapes *synthesized* by the generator $\mathbf{G}$. For robustness, we discretize the space of possible viewpoints into 16 predefined *"view-bins"* and treat the view prediction as a classification problem that outputs a vector of view probabilities.

Viewpoint Clustering Ideally, each view-bin corresponds to a projection and an associated discriminator (Equation 5). The silhouette images assigned to the view-bin serve as a training set for the respective discriminator. However, depending on the viewpoint distribution over the training data, not all of the view-bins will contain a sufficient number of training images to train a discriminator. Therefore, instead of using all view-bins, we exploit the observation expressed in Equation 7, and merge nearby views to a pre-determined fixed number of view clusters.

Practically, we perform a K-means clustering on the predicted discretized view probability vectors of the training silhouette images. We use the view probability vectors instead of the estimated viewpoints to better handle ambiguous cases (e.g., front and back view produce identical silhouettes). The result will be a set of viewpoint distributions (as cluster centers) and a unique cluster *id* for each training image. We directly use the estimated viewpoints of the silhouettes images assigned to the i^{th} cluster as the latent projection viewpoint distribution $\mathbf{\Phi}_i$ to ensure that the (synthesized) projected distribution follows the intrinsic distribution of the training silhouette data.

Joint Training Both the multi-projection GAN and the viewpoint prediction classifier require the other for training (i.e., multi-projection GAN requires viewpoint estimates, and the view prediction classifier requires GAN-generated 3D shapes for training). We resolve this conundrum by jointly training both in an iterative fashion where we alternate between training one network while keeping the other fixed. We bootstrap this iterative joint approach by training an initial generator using a single-discriminator 3D shape GAN assuming uniformly distributed viewpoints for training the discriminator. We will refer to the multi-projection GAN trained with joint viewpoint predictions as *VP-MP-GAN*. Figure 2 summarizes our full pipeline.

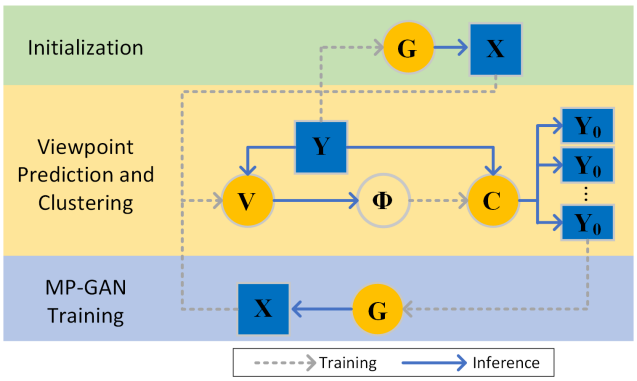

Figure 2. Overview of *VP-MP-GAN* training. Starting from a single discriminator GAN, we iteratively alternate between training a viewpoint classifier (based on training images produced by the generator $\mathbf{G}$), and training *MP-GAN* (based on predicted viewpoints for the silhouette images using the viewpoint classifier $\mathbf{V}$).

4. Implementation and Training

Network Structure We follow the 3D voxel generator network structure of Wu *et al*. [34] which takes a input vector sampled from $N(0, 1)$ and outputs a $64 \times 64 \times 64$ voxel grid via a sequence of 3D convolution and upsample layers. We employ batch normalization after each convolution and upsample layer and use the ReLU activation function. The discriminators take a 64×64 binary silhouette image as input, and output a single scalar probability value. Each discriminator contains 4 convolutional blocks with a stride of 2, followed by a single fully-connected layer. We share the first convolutional layers among the different discriminators, and use spectrum normalization [22] for each layer with the LeakyReLU activation function. The view prediction network shares the same convolutional structure as the discriminators, but with using batch normalization instead of spectrum normalization, and outputting the probability for the 16 discretized view-bins. We refer to the supplemental material for the detailed network structures.

Training Data We demonstrate our method on four different datasets: a synthetic dataset for validation and ablation study, and three different real-world datasets. The syn-

thetic dataset consists of $40,000$ rendered silhouette images for 6,000 3D shapes of chairs from the ShapeNet dataset [5]. All silhouettes are rendered from viewpoints with no elevation angle and uniformly distributed azimuth angles in $[0, 2\pi]$. The three real-world datasets are: $\sim$22,000 chair images mined from internet image repositories and *Pix3D* [27], a dataset of $\sim$36,000 car images mined from internet image repositories plus the Stanford car dataset [19], and a dataset of $\sim$12,000 bird images (*CUB-Birds-200-2011*) [30]. For *Pix3D* and *CUB-Birds-200-2011* we directly use the silhouette masks provided in the database (but we do not use the additional labels). In addition, for *CUB-Birds-200-2011*, we also remove close-up views and occluded images as their silhouettes do not provide complete shape information, as well as images with birds with open wings due to the scarcity of such images. For the images from other sources, we extract the silhouettes with an off-the-shelf segmentation network [13] and manually remove instances with corrupted masks.

Implementation Details We implemented our multi-projection shape GAN framework in TensorFlow [1]. For all experiments, the resulting networks are trained with the ADAM optimizer with a 10^{-4} learning rate, $\beta_1 = 0.5$, $\beta_2 = 0.9$, a $1:1$ training ratio between the generator and the discriminators, and a batch size of 32. For each training iteration, we generate one batch of voxel shapes, and for each generated shape and projection discriminator compute a silhouette image with randomly sampled viewpoint ($\sim \mathbf{\Phi}_i$). In addition, we also sample one batch from the training set corresponding to each projection. During back-propagation, the gradients from each of the discriminators are averaged to drive the generator training (Equation 6).

We limit viewpoint prediction in *VP-MP-GAN* to azimuth angles, since most collected images are dominated by viewpoint changes in the azimuth angle. We split the azimuth range $[0, 2\pi]$ into 16 uniformly distributed view-bins. For the view classifier training, we randomly synthesize 10,000 3D shapes, and generate a silhouette image for a random view within each of the view-bins, yielding a total of 160,000 training data pairs in each epoch. We cluster the view distributions in 8 clusters for all experiments and store the view distributions $\mathbf{\Phi}_i$ for each discriminator in histograms. To avoid outliers, we remove all bins with a probability less than 10% and renormalize the distribution.

Due to the intrinsic ambiguity of silhouette images for many viewpoints (e.g., front and back silhouettes look the same), the generated shapes may not align. Although the multi-projection GAN can still learn the 3D shape distribution without alignment, such ambiguities make the silhouette images from different views less distinct, thus reducing the effectiveness of the multiple discriminators. For datasets with known symmetry, we can leverage this prior knowledge by explicitly modeling the symmetry. In practice, we enforce symmetry by only generating half of the voxel shape and mirroring the remaining half over the symmetry axis. In our experiments, we enforce symmetry for the chair and car dataset; but not for the bird dataset which exhibits non-symmetric poses.

Training *VP-MP-GAN* at a resolution of 64^3 takes on average 40 hours on 4 Nvidia GTX 1080Ti cards.

Table 1. FID score [15] of *MP-GAN* trained on synthetic training data of chairs with reference viewpoint estimates for varying numbers of projections.

Num. of Discriminators	1	2	4	6	8	16	24
FID Score	79.61	49.93	36.22	34.22	33.27	32.45	29.45

5. Experiments

To validate our 3D shape generator, we perform an ablation study to demonstrate the impact of the number of projections and view clusters (subsection 5.1). In addition, we perform a comparison against three related methods (subsection 5.2). Finally, we show that our solution works well on non-synthetic image collections (subsection 5.3).

5.1. Ablation Study

We perform our ablation study on the synthetic chair dataset (section 4) for which we also have reference 3D shapes (not used for training). In this study, we evaluate the quality of the generated results quantitatively using the FID score [15] with an existing voxel classification network [21] trained on the ShapeNet dataset [5] as the feature extractor.

Impact of Number of Projections To analyze the quality of the generator, we train *MP-GAN* on a pre-defined number of view distributions and pre-assign the training images to the correct view clusters created by merging nearby view-bins. Table 1 summarizes the FID scores of MP-GAN for a varying number of projections (and thus discriminators). Note that MP-GAN reverts to a regular single projection GAN for the single discriminator case. We observe that the FID score decreases, and thus the generator quality improves, as the number of projections increases. However, we also observe a diminishing return (e.g., at 16 and 24 projections) when increasing number of projections for a fixed number of training data as the differences between the projection distributions decreases.

Impact of Number of View Clusters We repeat the above experiment, but this time on *VP-MP-GAN* (i.e., with view prediction) on unannotated training silhouette images. Table 2 lists the FID scores for a varying number of view clusters, which also changes the number of projections and discriminators (i.e., each projection is assigned to a view

Table 2. FID score for *VP-MP-GAN* trained on the synthetic training data of chairs (with unknown viewpoints) for varying number of view clusters.

Num. of Clusters	1	2	4	6	8	16	24
FID Score	79.61	53.83	39.23	35.22	34.32	34.10	33.95

Table 3. Evolution of view classification accuracy during training of *VP-MP-GAN* for 8 view clusters. Each (alternating) iteration includes 40,000 GAN training iterations, and 40,000 view predictor training iterations. The "reference" column refers to the view predictor accuracy trained on silhouette images with exact viewpoint estimates.

Num. of Iterations	Ref.	1	2	3	4	5
Accuracy	83.2%	42.7%	66.5%	69.9%	73.3%	75.6%

Table 4. View classification accuracy of *VP-MP-GAN* for a varying number of view clusters after 5 (alternating) iterations. The "reference" column refers to the view predictor accuracy trained on silhouette images with exact viewpoint estimates.

Num. of Clusters	Ref.	1	2	4	6	8
Accuracy	83.2%	42.7%	52.1%	54.5%	66.1%	75.6%

cluster). The upper-bound for the FID scores is set by the *MP-GAN* (Table 1) as these are trained with the exact viewpoint. Compared to the upper-bound, we can see that the scores for *VP-MP-GAN* are similar or slightly larger. As the number of view clusters increases, the inevitable inaccuracies introduced by the view prediction weigh more on the accuracy, resulting in a slightly larger FID score.

As detailed in subsection 3.2, we iteratively refine the viewpoint classifier (on 16 pre-defined bins). Table 3 shows the improvement in view prediction accuracy with each joint training iteration for *VP-MP-GAN* with 8 view clusters. The improvement in view prediction accuracy does not only indicate that the view prediction improves, but also that the learned shape distribution is closer to the target distribution. Table 4 further demonstrates this by listing the accuracy for varying number of view clusters (for 5 iterations); more view clusters result in a more accurate generator, which in turn yields a more accurate view prediction.

Finally, Figure 3 illustrates the view prediction distribution accuracy on the synthetic chair dataset. Note that *VP-MP-GAN* is able to learn the correct distribution with non-uniformly distributed views focused at 8 peaks of the 16 bins. For images gathered from internet repositories, our view prediction also produces plausible results; Figure 4 shows viewpoint classification results for the real-world chair and bird datasets for selected views and images.

5.2. Comparison

Figure 5 compares the results of our *VP-MP-GAN* to *3D-GAN* [34] and *PrGAN* [10] on the synthetic chair dataset.

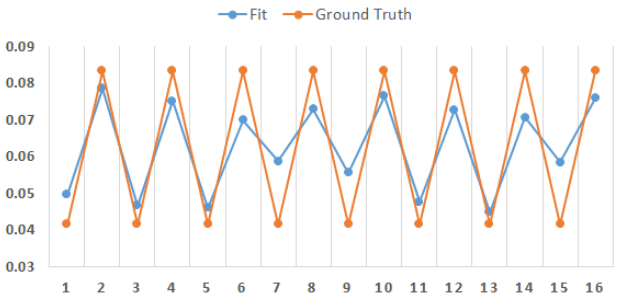

Figure 3. Accuracy of the estimated view distribution (blue) compared to a reference non-uniform distribution of viewpoints (orange) for the synthesized chair dataset.

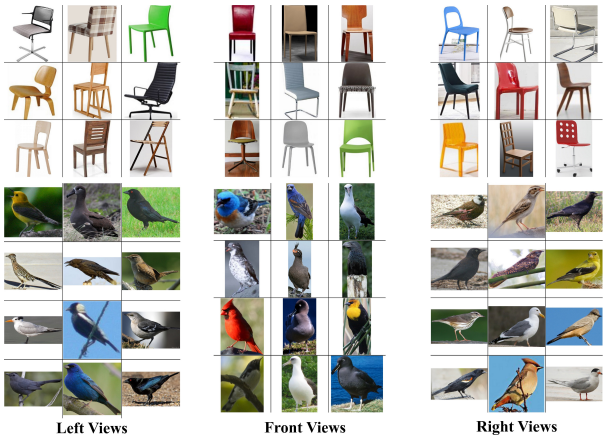

Figure 4. Selection of view-classified training images for the unannotated real-world chair and bird image datasets.

For a fair comparison, we apply the symmetry constraint to both methods and train both networks with our training data. We also list the FID score (in parentheses) of authors' original implementations for reference. *3D-GAN* is directly trained on the reference 3D data, and therefore scores a slightly higher FID score. Nevertheless, our generated shapes exhibit a similar visual quality. Similar to our method, *PrGAN* is also trained on silhouette images without the reference 3D data. However, *PrGAN* assumes known viewpoints and relies on a single discriminator only, resulting in a less accurate shape generator.

Recently, Henderson *et al*. [14] introduced a method for learning 3D shape distributions from shaded images using a VAE approach. Their method uses a 3D mesh representation instead of a voxel grid, making a direct comparison difficult. We therefore, adapt and retrain their method with a voxel generator instead of a mesh generator on silhouette images with a uniform view distribution and added symmetry constraint. Figure 5 shows that *VP-MP-GAN* produces higher quality voxel shapes and exhibits a lower FID score on the synthetic chair dataset.

Figure 5. Quantitative and qualitative comparison of four generators: VAE-based generator [14] (top), 3D-GAN [34] (2nd row), PrGAN [10] (3rd row), and *VP-MP-GAN* (bottom). The corresponding FID scores are shown on the left. We also report the FID score of the authors' original implementation in parenthesis.

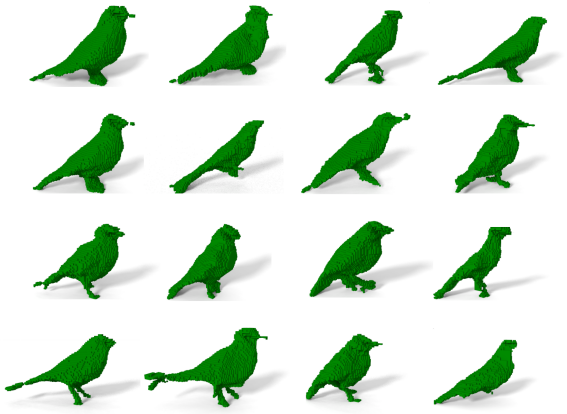

Figure 6. Results generated by *VP-MP-GAN* trained on the *bird* dataset.

5.3. Results on Real-world Datasets

We demonstrate our method's ability to learn the 3D shape distribution from real-world image datasets on three real-world image collections. Figure 6 shows examples of generated bird shapes. Note that there currently does not exist a database of 3D bird shapes, and thus the only way to learn a bird-shape generator is directly from images. Figure 7 and Figure 8 show results of generated chairs and cars, respectively, trained from photographs mined from online photo-collections (without using any synthetic images generated from ShapeNet). As can be seen, in all cases, *VP-MP-GAN* is able to produce high quality voxel shapes from unannotated silhouette images. We also refer to the supplemental material for more results on these datasets.

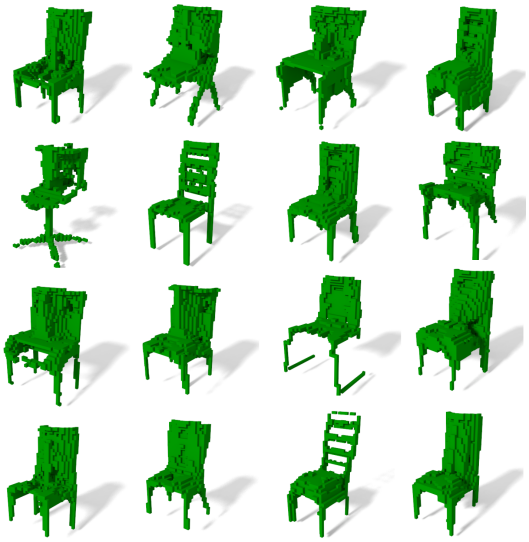

Figure 7. Results generated by *VP-MP-GAN* trained on the internet-mined *chair* dataset.

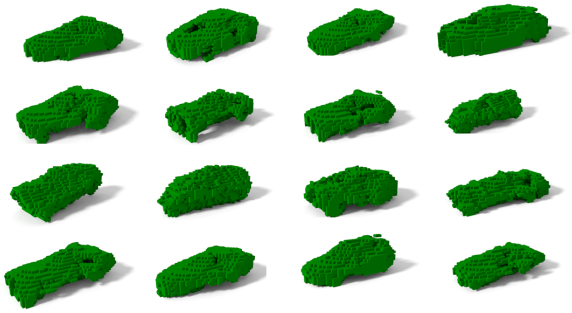

Figure 8. Results generated by *VP-MP-GAN* trained on the *car* dataset.

Limitations: *VP-MP-GAN* infers the distribution of 3D shapes from unoccluded silhouette images. Similar to classic computer vision methods that infer a single shape from silhouette images, our method can also not model concavities, and it requires a complete unoccluded view of the objects in the images. Using depth images instead of silhouette images can resolve the concavity issue. However, we deliberately did not go this route as depth images are not readily available and need to be specially captured.

6. SVBRDF Modeling

Our multi-projection GAN framework is not restricted to 3D shape modeling only, and it can potentially be applied to other applications that model high-dimensional data for which easy access to low-dimensional projections is available. We demonstrate the generality of the multi-projection framework by learning the distribution of spatially-varying material appearance in the form of property maps for spatially-varying bidirectional reflectance distribution functions (SVBRDF) for certain kinds of natural materials such

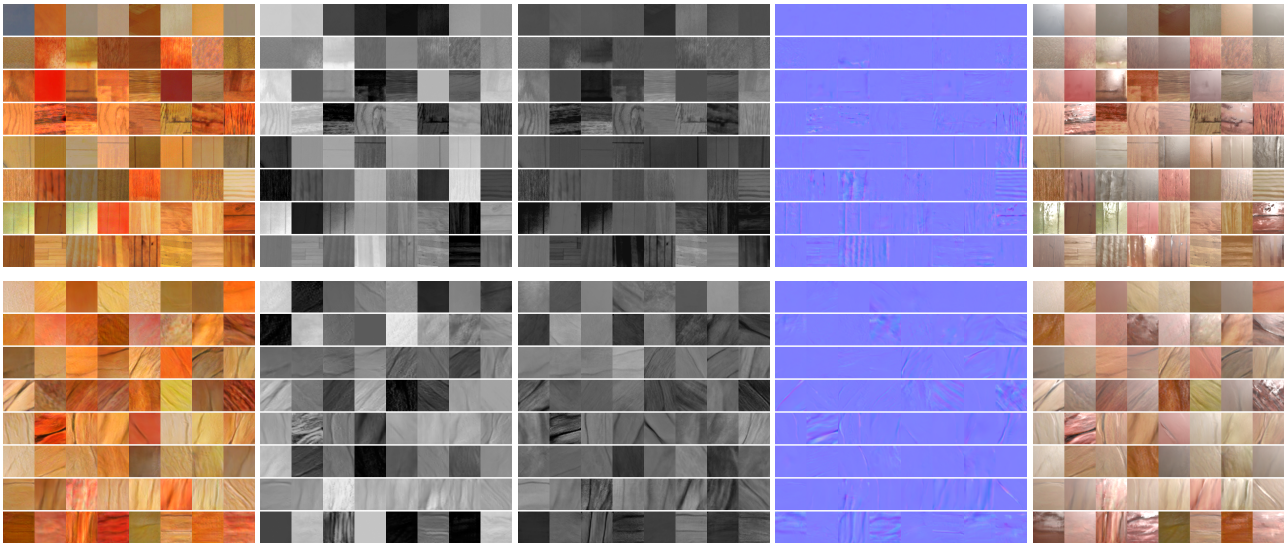

Figure 9. A selection of SVBRDFs generated with our multi-projection GAN (bottom) compared to samples drawn from the training dataset (top).

as wood, metal, and plastics. A SVBRDF is a 6D function that describes how the appearance varies with position, view angle, and incident lighting direction. We model the SVBRDF by a set of four 2D parameter maps that describe the Cook-Torrance BRDF model's [7] reflectance parameters for each surface point (i.e., diffuse albedo, specular albedo, specular roughness, and surface normal).

Unlike textures which can be acquired with a single photograph, accurate characterization of the full SVBRDF and surface normal details of a material is a complex and time-consuming process [31]. However, it is relatively easy to collect uncorresponded example maps of certain parameters (e.g., photometric stereo for normal maps, diffuse albedo via cross-polarized photographs, etc). Furthermore, it is easy to obtain photographs of materials under unknown lighting from internet photo-collections. Both of these types of images represent projections of the 6D SVBRDF. An image of a property map (diffuse or specular albedo, specular roughness, or normal maps) corresponds to a trivial projection that *"selects"* a single property map. A photograph under unknown lighting corresponds to a *"rendering"* projection where the unknown natural lighting distribution is modeled by the latent projection parameter $\mathbf{\Phi}$.

Figure 9 shows (corresponded) generated appearance property maps obtained with our multi-projection GAN, trained on a dataset containing partial appearance property maps as well as real photographs from the OpenSurfaces dataset [3]. The last column in Figure 9 shows a rendering of the generated material under a novel lighting condition. As can be seen, the generated materials produce a plausible *wood* material appearance. We refer to the supplemental material for additional SVBRDF results, a didactic example using the MNIST dataset, as well as the technical details including the training process, network structure, and the collection and preparation of the training data.

7. Conclusion

We proposed a novel weakly supervised method for learning the distribution of 3D shapes for a class of objects from unoccluded silhouette images. Key to our method is a novel multi-projection formulation of GANs that learns a high-dimensional distribution (i.e., voxel grid) from multiple, easier to acquire, lower-dimensional training data consisting of silhouette images from different objects from multiple viewpoints. Our method does not require that the silhouettes from multiple views are corresponded, nor that the viewpoints are known. The generator network is trained with cues from multiple discriminators in parallel. Each discriminator operates on the subset of the training data corresponding to a particular viewpoint. Our second contribution is a novel joint training strategy for training the view prediction network in an iterative fashion with the multi-projection GAN. We demonstrated the effectiveness of our 3D voxel generator on both synthetic and real-world datasets. Furthermore, we showed that our multi-projection framework is more generally applicable than to 3D shape modeling only, and demonstrated this by training a SVBRDF generator from 2D images.

Acknowledgments We would like to thank the reviewers for their constructive feedback. We also thank Baining Guo for discussions and suggestions. Pieter Peers was partially supported by NSF grant IIS-1350323 and gifts from Google, Activision, and Nvidia.

References

[1] M. Abadi, P. Barham, J. Chen, Z. Chen, A. Davis, J. Dean, M. Devin, S. Ghemawat, G. Irving, M. Isard, M. Kudlur, J. Levenberg, R. Monga, S. Moore, D. G. Murray, B. Steiner, P. Tucker, V. Vasudevan, P. Warden, M. Wicke, Y. Yu, and X. Zheng. Tensorflow: A system for large-scale machine learning. In *OSDI*, pages 265–283, 2016. 5

[2] M. Arjovsky, S. Chintala, and L. Bottou. Wasserstein generative adversarial networks. In *ICML*, pages 214–223, 2017. 2

[3] S. Bell, P. Upchurch, N. Snavely, and K. Bala. OpenSurfaces: A richly annotated catalog of surface appearance. *ACM Trans. on Graph.*, 32(4), 2013. 8

[4] A. Bora, E. Price, and A. G. Dimakis. Ambientgan: Generative models from lossy measurements. In *ICLR*, 2018. 2

[5] A. X. Chang, T. A. Funkhouser, L. J. Guibas, P. Hanrahan, Q. Huang, Z. Li, S. Savarese, M. Savva, S. Song, H. Su, J. Xiao, L. Yi, and F. Yu. Shapenet: An information-rich 3d model repository. *arXiv*, 2015. 5

[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. In *ECCV*, 2016. 1, 2

[7] R. L. Cook and K. E. Torrance. A reflectance model for computer graphics. *ACM Trans. Graph.*, 1(1):7–24, 1982. 8

[8] I. P. Durugkar, I. Gemp, and S. Mahadevan. Generative multi-adversarial networks. *arXiv*, 2016. 2

[9] H. Fan, H. Su, and L. J. Guibas. A point set generation network for 3d object reconstruction from a single image. In *CVPR*, volume 2, page 6, 2017. 1, 2

[10] M. Gadelha, S. Maji, and R. Wang. 3D Shape Induction from 2D Views of Multiple Objects. In *International Conference on 3D Vision*, dec 2017. 2, 6, 7

[11] R. Girdhar, D. F. Fouhey, M. Rodriguez, and A. Gupta. Learning a predictable and generative vector representation for objects. In *ECCV*, pages 484–499, 2016. 2

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

[13] K. He, G. Gkioxari, P. Dollár, and R. Girshick. Mask r-cnn. In *ICCV*, pages 2980–2988, 2017. 5

[14] P. Henderson and V. Ferrari. Learning to generate and reconstruct 3d meshes with only 2d supervision. In *BMVC*, 2018. 2, 6, 7

[15] M. Heusel, H. Ramsauer, T. Unterthiner, B. Nessler, and S. Hochreiter. Gans trained by a two time-scale update rule converge to a local nash equilibrium. In *NIPS*, pages 6626–6637, 2017. 5

[16] L. Jiang, S. Shi, X. Qi, and J. Jia. Gal: Geometric adversarial loss for single-view 3d-object reconstruction. In *ECCV*, pages 820–834, 2018. 2

[17] A. Kanazawa, M. J. Black, D. W. Jacobs, and J. Malik. End-to-end recovery of human shape and pose. In *CVPR*, pages 7122–7131, 2018. 2

[18] A. Kanazawa, S. Tulsiani, A. A. Efros, and J. Malik. Learning category-specific mesh reconstruction from image collections. In *ECCV*, 2018. 2

[19] J. Krause, M. Stark, J. Deng, and L. Fei-Fei. 3d object representations for fine-grained categorization. In *3dRR-13*, 2013. 5

[20] F. Massa, R. Marlet, and M. Aubry. Crafting a multi-task CNN for viewpoint estimation. In *BMVC*, 2016. 4

[21] D. Maturana and S. Scherer. Voxnet: A 3d convolutional neural network for real-time object recognition. In *IROS*, pages 922–928, 2015. 5

[22] T. Miyato, T. Kataoka, M. Koyama, and Y. Yoshida. Spectral normalization for generative adversarial networks. In *ICLR*, 2018. 4

[23] B. Neyshabur, S. Bhojanapalli, and A. Chakrabarti. Stabilizing GAN training with multiple random projections. *arXiv*, 2017. 2

[24] A. Radford, L. Metz, and S. Chintala. Unsupervised representation learning with deep convolutional generative adversarial networks. *arXiv*, 2015. 2

[25] A. Sinha, A. Unmesh, Q. Huang, and K. Ramani. Surfnet: Generating 3d shape surfaces using deep residual networks. In *CVPR*, volume 1, 2017. 1, 2

[26] H. Su, C. R. Qi, Y. Li, and L. J. Guibas. Render for cnn: Viewpoint estimation in images using cnns trained with rendered 3d model views. In *ICCV*, December 2015. 4

[27] X. Sun, J. Wu, X. Zhang, Z. Zhang, C. Zhang, T. Xue, J. B. Tenenbaum, and W. T. Freeman. Pix3d: Dataset and methods for single-image 3d shape modeling. In *CVPR*, 2018. 5

[28] S. Tulsiani, A. A. Efros, and J. Malik. Multi-view consistency as supervisory signal for learning shape and pose prediction. In *CVPR*, 2018. 2, 4

[29] S. Tulsiani, T. Zhou, A. A. Efros, and J. Malik. Multi-view supervision for single-view reconstruction via differentiable ray consistency. In *CVPR*, 2017. 2, 3

[30] C. Wah, S. Branson, P. Welinder, P. Perona, and S. Belongie. The Caltech-UCSD Birds-200-2011 Dataset. Technical Report CNS-TR-2011-001, California Institute of Technology, 2011. 5

[31] M. Weinmann and R. Klein. Advances in geometry and reflectance acquisition. In *ACM SIGGRAPH Asia, Course Notes*, 2015. 8

[32] J. Wu, Y. Wang, T. Xue, X. Sun, B. Freeman, and J. Tenenbaum. Marrnet: 3d shape reconstruction via 2.5d sketches. In *NIPS*. 1, 2

[33] J. Wu, T. Xue, J. J. Lim, Y. Tian, J. B. Tenenbaum, A. Torralba, and W. T. Freeman. Single image 3d interpreter network. In *ECCV*, 2016. 4

[34] 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. In *NIPS*, pages 82–90, 2016. 1, 2, 4, 6, 7

[35] X. Yan, J. Yang, E. Yumer, Y. Guo, and H. Lee. Perspective transformer nets: Learning single-view 3d object reconstruction without 3d supervision. In D. D. Lee, M. Sugiyama, U. V. Luxburg, I. Guyon, and R. Garnett, editors, *NIPS*, pages 1696–1704. 2016. 2

[36] B. Yang, S. Rosa, A. Markham, N. Trigoni, and H. Wen. Dense 3d object reconstruction from a single depth view. In *TPAMI*, 2018. 1, 2

[37] R. Zhu, H. K. Galoogahi, C. Wang, and S. Lucey. Rethinking reprojection: Closing the loop for pose-aware shape reconstruction from a single image. In *ICCV*, pages 57–65, 2017. 1, 2