Multi-view Supervision for Single-view Reconstruction via Differentiable Ray Consistency

Shubham Tulsiani, Tinghui Zhou, Alexei A. Efros, Jitendra Malik

Abstract—We study the notion of consistency between a 3D shape and a 2D observation and propose a differentiable formulation which allows computing gradients of the 3D shape given an observation from an arbitrary view. We do so by reformulating view consistency using a differentiable ray consistency (DRC) term. We show that this formulation can be incorporated in a learning framework to leverage different types of multi-view observations *e.g.* foreground masks, depth, color images, semantics *etc.* as supervision for learning single-view 3D prediction. We present empirical analysis of our technique in a controlled setting. We also show that this approach allows us to improve over existing techniques for single-view reconstruction of objects from the PASCAL VOC dataset.

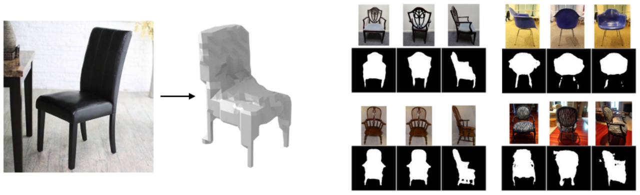
Index Terms—3D Reconstruction, Multi-view Supervision, Ray Consistency

1 Introduction

Consider the flat, two-dimensional image of a chair in Figure 1(a). A human observer cannot help but perceive its 3D structure. Even though we may have never seen this particular chair before, we can readily infer, from this single image, its likely 3D shape and orientation. To make this inference, we must rely on our knowledge about the 3D structure of other, previously seen chairs. But how did we acquire this knowledge? And can we build computational systems that learn about 3D in a similar manner?

Humans are moving organisms: our ecological supervision [1] comprises of observing the world and the objects in it from different perspectives, and these multiple views inform us of the underlying geometry. This insight has been successfully leveraged by a long line of geometry-based reconstruction techniques. While these structure from motion or multi-view stereo methods work for specific instances, they do not, unlike humans, generalize to predict the 3D shape of a novel instance given a single view. Recent learning-based methods have attempted to address this single-view 3D inference task. However, these approaches rely on full 3D supervision and require known 3D shape for each training image. Not only is this form of supervision ecologically implausible, it is also practically tedious to acquire and difficult to scale. Instead, as depicted in Figure 1(b), our goal is to learn 3D prediction using the more naturally plausible multi-view supervision.

We therefore aim to combine aspects of classical multi-view reconstruction with learning based prediction. Akin to the classical geometry-based approaches, we rely on multi-view supervisory signal, while being able to generalize to novel instances and infer their 3D structure from a single view. Our approach is to learn shape prediction by enforcing *geometric consistency* between the predicted 3D and the available multi-view data. Concretely, given one image of an object instance, we predict a corresponding shape, and enforce that this predicted shape is consistent with the multiple views of this instance.



a) Task: 3D shape from a 2D image.

b) Supervision: Multi-view observations of objects.

Fig. 1: We learn to predict 3D shape from a single input view. Our framework can leverage training data of the form of multi-view observations, and learn 3D reconstruction despite the lack of any direct supervision.

A central aspect of this approach is the notion of geometric consistency between a 3D shape and 2D image. In particular, our learning system requires signals for how to improve predicted shapes such that they become more consistent with the available observations. One way this problem has been traditionally addressed is by space carving [2]. Rays are projected out from pixels into the 3D space and each ray that is known not to intersect the object removes the volume in its path, thereby making the carved-out shape consistent with the observed image.

However, to leverage it in a learning-based system, we want to extend this notion of consistency to a differential setting. That is, instead of deleting chunks of volume all at once, we would like to compute incremental changes to the 3D shape that make it more consistent with the 2D image. In this paper, we present a differentiable ray consistency for-

- *S. Tulsiani, T. Zhou, A. A. Efros and J. Malik are with the Department of Electrical Engineering and Computer Science, University of California at Berkeley. E-mail: {shubhtuls, tinghuiz, efros, malik}@eecs.berkeley.edu*

. Project website with code: https://shubhtuls.github.io/drc/

mulation that allows computing the gradient of a predicted 3D shape of an object, given an observation (depth image, foreground mask, color image *etc.*.) from an arbitrary view. The differentiability of our consistency formulation is what allows its use in a learning framework, such as a neural network. Every new piece of evidence gives gradients for the predicted shape, which, in turn, yields incremental updates for the underlying prediction model. Since this prediction model is shared across object instances, it is able to find and learn from the commonalities across different 3D shapes, requiring only sparse per-instance supervision.

We first describe in Section 3 the formulation of our geometric consistency loss, and then present our approach to leverage it for learning single-view reconstruction via multi-view supervision in Section 4. In Section 5 we demonstrate the applicability of our framework to learn 3D inference across various scenarios.

2 Related Work

Object Reconstruction from Image-based Annotations. Blanz and Vetter [3] demonstrated the use of a morphable model to capture 3D shapes. Cashman and Fitzgibbon [4] learned these models for complex categories like dolphins using object silhouettes and keypoint annotations for training and inference. Tulsiani *et al.* [5] extended similar ideas to more general categories and leveraged recognition systems [6], [7], [8] to automate test-time inference. Wu *et al.* [9], using similar annotations, learned a system to predict sparse 3D by inferring parameters of a shape skeleton. However, since the use of such low-dimensional models restricts expressivity, Vicente *et al.* [10] proposed a non-parametric method by leveraging surrogate instances – but at the cost of requiring annotations at test time. We leverage similar training data but using a CNN-based voxel prediction framework allows test time inference without manual annotations and allows handling large shape variations.

Object Reconstruction from 3D Supervision. The advent of deep learning along with availability of large-scale synthetic training data has resulted in applications for object reconstruction. Choy *et al.* [11] learned a CNN to predict a voxel representation using a single (or multiple) input image(s). Girdhar *et al.* [12] also presented similar results for single-view object reconstruction, while also demonstrating some results on real images by using realistic rendering techniques [13] for generating training data. Several approaches have further improved these voulmetric predictions [14], [15], [16], or pursued alternate 3D representations such as point clouds [17], octrees [18], [19], or meshes [20], [21], [22]. A crucial assumption in the procedure of training these models, however, is that full 3D supervision is available. As a result, these methods primarily train using synthetically rendered data where the underlying 3D shape is available.

While the progress demonstrated by these methods is encouraging and supports the claim for using CNN based learning techniques for reconstruction, the requirement of explicit 3D supervision for training is potentially restrictive. We relax this assumption and show that alternate sources of supervision can be leveraged. It allows us to go beyond reconstructing objects in a synthetic setting, to extend to real datasets which do not have 3D supervision.

Multi-view Instance Reconstruction. Perhaps most closely related to our work in terms of the proposed formulation is the line of work in geometry-based techniques for reconstructing a single instance given multiple views. Visual hull [23] formalizes the notion of consistency between a 3D shape and observed object masks. Techniques based on this concept [24], [25] can obtain reconstructions of objects by space carving using multiple available views. It is also possible, by jointly modeling appearance and occupancy, to recover 3D structure of objects/scenes from multiple images via ray-potential based optimization [26], [27] or inference in a generative model [28], [29]. Ulusoy *et al.* [30] propose a probabilistic framework where marginal distributions can be efficiently computed. More detailed reconstructions can be obtained by incorporating additional signals *e.g.* depth or semantics [31], [32], [33].

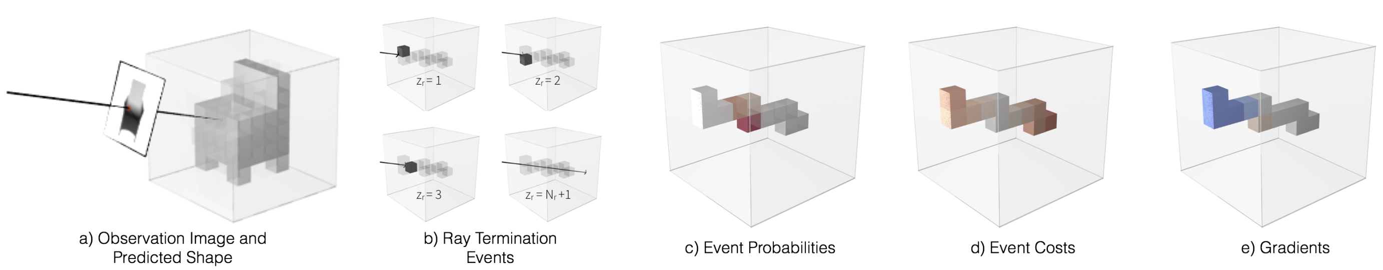
The main goal in these prior works is to reconstruct a specific scene/object from multiple observations and they typically infer a discrete assignment of variables such that it is maximally consistent with the available views. Our insight is that similar cost functions which measure consistency, adapted to treat variables as continuous probabilities, can be used in a learning framework to obtain gradients for the current prediction. Crucially, the multi-view reconstruction approaches typically solve a (large) optimization to reconstruct a particular scene/object instance and require a large number of views. In contrast, we only need to perform a single gradient computation to obtain a learning signal for the CNN and can even work with sparse set of views (possibly even just one view) per instance.

Multi-view Supervision for Single-view Depth Prediction. While single-view depth prediction had been dominated by approaches with direct supervision [34], recent approaches based on multi-view supervision have shown promise in achieving similar (and sometimes even better) performance. Garg *et al.* [35] and Godard *et al.* [36] used stereo images to learn a single image depth prediction system by minimizing the inconsistency as measured by pixel-wise reprojection error. Zhou *et al.* [37] further relax the constraint of having calibrated stereo images, and learn a single-view depth model from monocular videos. The motivation of these multi-view supervised depth prediction approaches is similar to ours, but we aim for 3D instead of 2.5D predictions and address the related technical challenges in this work.

3 Formulation

In this section, we formulate a differentiable 'view consistency' loss function which measures the inconsistency between a (predicted) 3D shape and a corresponding observation image with an associated known (or predicted) camera viewpoint. We first formally define our problem setup by instantiating the representation of the 3D shape and the observation image with which the consistency is measured.

Shape Representation. Our 3D shape representation is parametrized as occupancy probabilities of cells in a discretized 3D voxel grid, denoted by the variable x. We use the convention that x_i represents the probability of the i^{th} voxel being empty (we use the term 'occupancy probability'



a) Observation Image and Predicted Shape

b) Ray Termination Events

c) Event Probabilities

d) Event Costs

e) Gradients

Fig. 2: Visualization of various aspects of our Differentiable Ray Consistency formulation. a) Predicted 3D shape represented as probabilistic occupancies and the observation image where we consider consistency between the predicted shape and the ray corresponding to the highlighted pixel. b) Ray termination events (Section 3.2) – the random variable $z_r = i$ corresponds to the event where the ray terminates at the i^{th} voxel on its path, $z_r = N_r + 1$ represents the scenario where the ray escapes the grid. c) Depiction of event probabilities (Section 3.2) where red indicates a high probability of the ray terminating at the corresponding voxel. d) Given the ray observation, we define event costs (Section 3.3). In the example shown, the costs are low (white color) for events where ray terminates in voxels near the observed termination point and high (red color) otherwise. e) The ray consistency loss (Section 3.4) is defined as the expected event cost and our formulation allows us to obtain gradients for occupancies (red indicates that loss decreases if occupancy value increases, blue indicates the opposite). While in this example we consider a depth observation, our formulation allows incorporating diverse kinds of observations by defining the corresponding event cost function as discussed in Section 3.3 and Section 3.5. Best viewed in color.

for simplicity even though it is a misnomer as the variable x is actually 'emptiness probability'). Note that the choice of discretization of the 3D space into voxels need not be a uniform grid – the only assumption we make is that it is possible to trace rays across the voxel grid and compute intersections with cell boundaries.

Observation and Camera. We aim for the shape to be consistent with some available observation O from a camera C. This 'observation' can take various forms *e.g.* a depth image, an object foreground mask, a color image *etc.*– these are treated similarly in our framework. Concretely, we have a observation-camera pair (O, C) where the 'observation' O is from a view defined by (known or predicted) camera C. The camera C is defined via an intrinsic matrix and the extrinsics specifying its rotation and translation in the world coordinate frame.

Our view consistency loss, using the notations mentioned above, is of the form $L(x; (O, C))$. Towards defining this loss, in Section 3.1 we reduce the notion of consistency between the 3D shape and an observation image to consistency between the 3D shape and a ray with associated observations. We then present a differentiable formulation for ray consistency, the various aspects of which are visualized in Figure 2. In Section 3.2, we examine the case of a ray travelling though a probabilistically occupied grid and in Section 3.3, we instantiate costs for each probabilistic ray-termination event. We then combine these to define the consistency cost function in Section 3.4. While we initially only consider the case of the shape being represented by voxel occupancies x, we show in Section 3.5 that it can be extended to incorporate optional per-voxel predictions p. This generalization allows us to incorporate other kinds of observation *e.g.* color images, pixel-wise semantics *etc.*. The generalized consistency loss function is then of the form $L(x, [p]; (O, C))$ where $[p]$ denotes an optional argument. The view consistency loss formulation we present, while differentiable w.r.t the shape x, is not differentiable w.r.t the camera C. In Section 3.6 we present an alternative formulation of this loss that, using a simple re-parametrization, is also differentiable w.r.t C.

3.1 View Consistency as Ray Consistency

Every pixel in the observation image O corresponds to a ray with a recorded observation (depth/color/foreground label/semantic label). Assuming known camera intrinsic parameters (f_u, f_v, u_0, v_0), the image pixel (u, v) corresponds to a ray r originating from the camera centre travelling in direction $(\frac{u-u_0}{f_u}, \frac{v-v_0}{f_v}, 1)$ in the camera coordinate frame. Given the camera extrinsics, the origin and direction of the ray r can also be inferred in the world frame.

Therefore, the available observation-camera pair (O, C) is equivalently a collection of arbitrary rays $\mathcal{R}$ where each $r \in \mathcal{R}$ has a known origin point, direction and an associated observation o_r *e.g.* depth images indicate the distance travelled before hitting a surface, foreground masks inform whether the ray hit the object, semantic labels correspond to observing category of the object the ray terminates in.

Analogous to the common practice in classical multi-view reconstruction approaches [27], [30], [32], [33] which formulate objectives using a set of ray potentials, we can similarly formulate the view consistency loss $L(x; (O, C))$ using per-ray based consistency terms $L_r(x)$. Here, $L_r(x)$ captures if the inferred 3D model x correctly explains the observations associated with the specific ray r. Our view consistency loss is then just the sum of the consistency terms across the rays:

$$L(x; (O, C)) \equiv \sum_{r \in \mathcal{R}} L_r(x) \quad (1)$$

Our task for formulating the view consistency loss is simplified to defining a differentiable ray consistency loss $L_r(x)$.

3.2 Ray-tracing in a Probabilistic Occupancy Grid

With the goal of defining the consistency cost $L_r(x)$, we examine the ray r as it travels across the voxel grid with occupancy probabilities x. The occupancy probabilities in this grid (instantiated by the shape parameters x) induce a distribution over possible terminations for a ray which can be efficiently computed [24], [29]. We denote the various likely ray terminations as *events* that can occur to ray r, and we can define $L_r(x)$ by seeing the incompatibility of these events with available observations o_r.

Ray Termination Events. Since we know the origin and direction for the ray r, we can trace it through the voxel grid - let us assume it passes though N_r voxels. The *events* associated with this ray correspond to it either terminating at one of these N_r voxels or passing through. We use a random variable z_r to correspond to the voxel in which the ray (probabilistically) terminates - with $z_r = N_r + 1$ to represent the case where the ray does not terminate. These events are shown in Figure 2.

Event Probabilities. Given the occupancy probabilities x, we want to infer the probability $q(z_r = i)$. The event $z_r = i$ occurs iff the previous voxels in the path are all unoccupied and the i^{th} voxel is occupied. Assuming an independent distribution of occupancies where the prediction x_i^r corresponds to the probability of the i^{th} voxel on the path of the ray r as being *empty*, we can compute the probability distribution for z_r.

$$q(z_r = i) = \begin{cases} (1 - x_i^r) \prod_{j=1}^{i-1} x_j^r, & \text{if } i \leq N_r \\ \prod_{j=1}^{N_r} x_j^r, & \text{if } i = N_r + 1 \end{cases} \tag{2}$$

3.3 Event Cost Functions

Note that each event ($z_r = i$), induces a prediction *e.g.* if $z_r = i$, we can geometrically compute the distance d_i^r the ray travels before terminating. We can define a cost function between the induced prediction under the event ($z_r = i$) and the available associated observations for ray o_r. We denote this cost function as $\psi_r(i)$ and it assigns a cost to event ($z_r = i$) based on whether it induces predictions inconsistent with o_r. We now show some examples of event cost functions that can incorporate diverse observations o_r and used in various scenarios.

Object Reconstruction from Depth Observations. In this scenario, the available observation o_r corresponds to the observed distance the ray travels d_{gt}^r. We use a simple distance measure between observed distance and event-induced distance to define $\psi_r(i)$.

$$\psi_r^{depth}(i) = |d_i^r - d_{gt}^r| \tag{3}$$

Object Reconstruction from Foreground Masks. We examine the case where we only know the object masks from various views. In this scenario, let $s_r \in \{0, 1\}$ denote the known information regarding each ray - $s_r = 0$ implies the ray r intersects the object *i.e.* corresponds to an image pixel within the mask, $s_r = 1$ indicates otherwise. We can capture this by defining the corresponding cost terms.

$$\psi_r^{mask}(i) = \begin{cases} s_r, & \text{if } i \leq N_r \\ 1 - s_r, & \text{if } i = N_r + 1 \end{cases} \tag{4}$$

We note that some concurrent approaches [38], [39] have also been proposed to specifically address the case of learning object reconstruction from foreground masks. These approaches, either though a learned [38] or fixed [39] re-projection function, minimize the discrepancy between the observed mask and the reprojected predictions. We show in the appendix that our ray consistency based approach effectively minimizes a similar loss using a geometrically derived re-projection function, while also allowing us to handle more general observations.

3.4 Ray-Consistency Loss

We have examined the case of a ray traversing through the probabilistically occupied voxel grid and defined possible ray-termination events occurring with probability distribution specified by $q(z_r)$. For each of these events, we incur a corresponding cost $\psi_r(i)$ which penalizes inconsistency between the event-induced predictions and available observations o_r. The per-ray consistency loss function $L_r(x)$ is simply the expected cost incurred.

$$L_r(x) = \mathbb{E}_{z_r}[\psi_r(z_r)] \tag{5}$$

$$L_r(x) = \sum_{i=1}^{N_r+1} \psi_r(i)\, q(z_r = i) \tag{6}$$

Recall that the event probabilities $q(z_r = i)$ were defined in terms of the voxel occupancies x predicted by the CNN (Eq. 2). Using this, we can compute the derivatives of the loss function $L_r(x)$ w.r.t the CNN predictions (see Appendix for derivation).

$$\frac{\partial\, L_r(x)}{\partial\, x_k^r} = \sum_{i=k}^{N_r} (\psi_r(i+1) - \psi_r(i)) \prod_{1 \leq j \leq i, j \neq k} x_j^r \tag{7}$$

The ray-consistency loss $L_r(x)$ completes our formulation of view consistency loss as the overall loss is defined in terms of $L_r(x)$ as in Eq. 1. The gradients derived from the view consistency loss simply try to adjust the voxel occupancy predictions x, such that events which are inconsistent with the observations occur with lower probabilities.

3.5 Incorporating Additional Labels

We have developed a view consistency formulation for the setting where the shape representation is described as occupancy probabilities x. In the scenario where alternate per-pixel observations (*e.g.* semantics or color) are available, we can modify consistency formulation to account for per-voxel predictions p in the 3D representation. In this scenario, the observation o_r associated with the ray r includes the corresponding pixel label and similarly, the induced prediction under event ($z_r = i$) includes the auxiliary prediction p_i^r for the i^{th} voxel on the ray's path.

Inspired by Savinov *et al.* [32], [33] who address a similar challenge for multi-view reconstruction, we incorporate consistency between these by extending $L_r(x)$ to $L_r(x, [p])$ by using a generalized event-cost term $\psi_r(i, [p_i^r])$ in Eq. 5 and Eq. 6. Examples of the generalized cost term for two scenarios are presented in Eq. 9 and Eq. 10. The gradients for occupancy predictions x_i^r are as previously defined in Eq. 7, but using the generalized cost term $\psi_r(i, [p_i^r])$ instead. The additional per-voxel predictions can also be trained using the derivatives below.

$$\frac{\partial\, L_r(x, [p])}{\partial\, p_r^i} = q(z_r = i) \frac{\partial\, \psi_r(i, [p_r^i])}{\partial\, p_r^i} \tag{8}$$

Note that we can define any event cost function $\psi(i,[p_i^r])$ as long as it is differentiable w.r.t p_i^r. We can interpret Eq. 8 as the additional per-voxel predictions p being updated to match the observed pixel-wise labels, with the gradient being weighted by the probability of the corresponding event.

Scene Reconstruction from Depth and Semantics. In this setting, the observations associated with each ray correspond to an observed depth d_{gt}^r as well as semantic class labels c_r. The event-induced prediction, if $z_r = i$, corresponds to depth d_i^r and class distribution p_i^r and we can define an event cost penalizing the discrepancy in disparity (since absolute depth can have a large variation) and the negative log likelihood of the observed class.

$$\psi_r^{sem}(i,p_i^r) = |\frac{1}{d_i^r} - \frac{1}{d_{gt}^r}| - log(p_i^r(c_r)) \tag{9}$$

Object Reconstruction from Color Images. In this scenario, the observations c_r associated with each ray corresponds to the RGB color values for the corresponding pixel. Assuming additional per voxel color prediction p, the event-induced prediction, if $z_r = i$, yields the color at the corresponding voxel *i.e.* p_i^r. We can define an event cost penalizing the squared error.

$$\psi_r^{color}(i,p_i^r) = \frac{1}{2}\|p_i^r - c_r\|^2 \tag{10}$$

In addition to defining the event cost functions, we also need to instantiate the induced observations for the event of ray escaping. We define $d_{N_r+1}^r$ in Eq. 3 and Eq. 9 to be a fixed large value, and $p_{N_r+1}^r$ in Eq. 9 and Eq. 10 to be uniform distribution and white color respectively. We discuss this further in the appendix.

3.6 Pose-Differentiable Ray Consistency

The loss formulation presented above is differentiable w.r.t the shape x, but not w.r.t the camera parameters C. This is because $\{x_i^r\}$, which represents the occupancy probability of the i^{th} voxel in the ray's path, is not a differentiable function of the camera (since the ordering of voxels on a ray's path is a discrete function). However, in certain scenarios *e.g.* when leveraging predicted camera parameters instead of known ones, it would be necessary to have a formulation where the loss is differentiable w.r.t both, shape and pose.

In order to overcome this, we use an alternate pose-differentiable ray consistency loss formulation, with the corresponding view loss denoted as $\tilde{L}(x;(O,C))$. We do so by redefining the variable $\{x_i^r\}$ to correspond to the occupancy at the i^{th} sample along the ray. Therefore, instead of using probabilities of voxels along a ray, we consider probabilities at point samples along a ray. Concretely, we sample points at a fixed set of $N_r = 80$ depth values $\{d_i | 1 \leq i \leq N\}$ along each ray.

To determine x_i^r, we look at the 3D coordinate of the corresponding point (determined using the camera rotation R, translation t and the intrinsic parameters), and trilinearly sample the shape x to determine the occupancy at this point.

$$l_i \equiv (\frac{u-u_0}{f_u}d_i, \frac{v-v_0}{f_v}d_i, d_i) \tag{11}$$

$$x_i^r = \mathcal{T}(x, R \times (l_i + t)\ ) \tag{12}$$

As the trilinear sampling function $\mathcal{T}$ is differentiable w.r.t its arguments, the sampled occupancy x_i^r, and consequently the alternate view consistency loss $\tilde{L}(x;(O,C))$, is differentiable w.r.t the shape x and the camera C.

We note that although this idea of using samples instead of voxels (similar to [39]) is less physically grounded, it provides us a convenient tool to obtain gradients for the predicted cameras. While we primarily use the original loss formulation for most of our experiments, we leverage this pose-differentiable loss in some scenarios where the associated cameras for observations are also predicted.

4 Learning Single-view Reconstruction

We aim to learn a function f modeled as a parameterized CNN f_θ, which given a single image I corresponding to a novel object, predicts its shape as a voxel occupancy grid. A straightforward learning-based approach would require a training dataset $\{(I_i, \bar{x}_i)\}$ where the target voxel representation $\bar{x}_i$ is known for each training image I_i. However, we are interested in a scenario where the ground-truth 3D models $\{\bar{x}_i\}$ are not available for training f_θ directly, as is often the case for real-world objects/scenes. While collecting the ground-truth 3D is not feasible, it is relatively easy to obtain 2D or 2.5D observations (e.g. depth maps) of the underlying 3D model from other viewpoints. In this scenario we can leverage the 'view consistency' loss function described in Section 3 to train f_θ.

We consider two supervision scenarios for learning f_θ. We first examine the setting where the available multi-view observations have known associated camera poses (*e.g.* as possible for a moving agent that knows its egomotion), and then address the scenario where even the camera poses associated are unknown.

4.1 Learning with Pose Supervision

Training Data. As our training data, corresponding to each training (RGB) image I_i in the training set, we also have access to one or more additional observations of the same instance from other views. The observations, as described in Section 3, can be of varying forms. Concretely, corresponding to image I_i, we have *one or more* observation-camera pairs $\{O_k^i, C_k^i\}$ where the 'observation' O_k^i is from a view defined by camera C_k^i. Note that these observations are required only for training; at test time, the learned CNN f_θ predicts a 3D shape from only a single 2D image.

Predicted 3D Representation. The output of our single-view 3D prediction CNN is $f_\theta(I) \equiv (x,[p])$ where x denotes voxel occupancy probabilities and $[p]$ indicates optional per-voxel predictions (used if corresponding training observations *e.g.* color, semantics are leveraged).

To learn the parameters θ of the single-view 3D prediction CNN, for each training image I_i we train the CNN to minimize the inconsistency between the prediction $f_\theta(I_i)$ and the one or more observation(s) $\{(O_k^i, C_k^i)\}$ corresponding to I_i. This optimization is the same as minimizing the (differentiable) loss function $\sum_i \sum_k L(f_\theta(I_i); (O^i_k, C^i_k))$ *i.e.* the sum of view consistency losses (Eq. 1) for observations across the training set. To allow for faster training, instead of using all rays as defined in Eq. 1, we randomly sample a few rays (about 1000) per view every stochastic gradient descent iteration.

4.2 Learning without Pose Supervision

In this supervision scenario, we do not assume known camera poses associated with the multiple views. Instead, we assume that we have an RGB image I^i_k associated with each observation image, and leverage a predicted camera to enforce geometric consistency. To operationalize this setup, in addition to learning the single-view 3D prediction CNN f_θ, we also jointly learn a pose prediction CNN g_ϕ. We first describe the training data and representations predicted, and then summarize the learning process.

Training Data. Similar to the setup in Section 4.1, we rely on multi-view training data, but without camera pose annotations. Corresponding to image I_i, we have observation-image pairs $\{O^i_k, I^i_k\}$ where the 'observation' O^i_k is associated with an RGB image I^i_k (which we use to predict pose).

Predictions. The output of the shape prediction CNN f_θ is a voxel occupancy grid as in Section 4.1. The pose prediction CNN g_ϕ predicts, from a single input image, the corresponding camera extrinsic parameters: a quaternion to instantiate the rotation, and a translation $\in \mathbb{R}^3$. We assume known camera intrinsics (although these can also be predicted), and therefore the predictions of g_ϕ suffice to instantiate the associated camera.

To jointly learn the shape and pose prediction CNNs f_θ and g_ϕ, we train these CNNs to minimize the inconsistency between a predicted shape prediction $f_\theta(I_i)$ and available observations with their corresponding predicted cameras $\{(O^i_k, g_\phi(I^i_k)\}$. As we also want the consistency loss to be differentiable w.r.t the predicted cameras, we use the formulation defined in Section 3.6, and minimize the loss function $\sum_i \sum_k \tilde{L}(f_\theta(I_i); (O^i_k, g_\phi(I^i_k)))$.

We empirically observe that training both f_θ and g_ϕ jointly from scratch, and without any direct supervision, is challenging. The optimization often gets stuck at a local minima for the camera pose prediction and only predicts a restricted range of poses *e.g.* conflating front and back facing chairs. To overcome this, we incorporate a pose prior as well as allow g_ϕ to predict a distribution of pose hypotheses instead of a single one. These modifications, described in more detail in the appendix, allow us to learn both shape and pose prediction using only multi-view supervision.

5 Experiments

We consider various scenarios where we can learn single-view reconstruction using our differentiable ray consistency (DRC) formulation. First, we examine the ShapeNet dataset where we use synthetically generated images and corresponding multi-view observations to study our framework. We then demonstrate applications on the PASCAL VOC dataset where we train a single-view 3D prediction system using only one observation per training instance. We then explore the application of our framework for scene reconstruction using short driving sequences as supervision. We also show qualitative results for using multiple color image observations as supervision for single-view reconstruction. While these scenarios assume known camera pose for training, we finally examine two settings where we demonstrate that we can learn 3D prediction even without pose supervision.

5.1 Empirical Analysis on ShapeNet

We study the framework presented and demonstrate its applicability with different types of multi-view observations and also analyze the susceptibility to noise in the learning signal. We perform experiments in a controlled setting using synthetically rendered data where the ground-truth 3D information is available for benchmarking.

Setup. The ShapeNet dataset [40] has a collection of textured CAD models and we examine 3 representative categories with large sets of available models : airplanes, cars, and chairs . We create random train/val/test splits and use rendered images with randomly sampled views as input to the single-view 3D prediction CNNs.

Our CNN model is a simple encoder-decoder which predicts occupancies in a voxel grid from the input RGB image (see appendix for details). To perform control experiments, we vary the sources of information available (and correspondingly, different loss functions) for training the CNN. The various control settings are briefly described below (and explained in detail in the appendix):

Ground-truth 3D. We assume that the ground-truth 3D model is available and use a simple cross-entropy loss for training. This provides an upper bound for the performance of a multi-view consistency method.

DRC (Mask/Depth). In this scenario, we assume that (possibly noisy) depth images (or object masks) from 5 random views are available for each training CAD model and minimize the view consistency loss.

Depth Fusion. As an alternate way of using multi-view information, we preprocess the 5 available depth images per CAD model to compute a pseudo-ground-truth 3D model. We then train the CNN with a cross-entropy loss, restricted to voxels where the views provided any information. Note that unlike our method, this is applicable only if depth images are available and is more susceptible to noise in observations. See appendix for further details and discussion.

Evaluation Metric. We use the mean intersection over union (IoU) between the ground-truth 3D occupancies and the predicted 3D occupancies. Since different losses lead to the learned models being calibrated differently, we report mean IoU at the optimal discretization threshold for each method (the threshold is searched at a category level).

Results. We present the results of the experiments in Table 1 and visualize sample predictions in Figure 3. In general, the qualitative and quantitative results in our setting of using only a small set of multi-view observations are encouragingly close to the upper bound of using ground-truth 3D as supervision. While our approach and the alternative way of depth fusion are comparable in the case of perfect depth information, our approach is more robust to noisy training

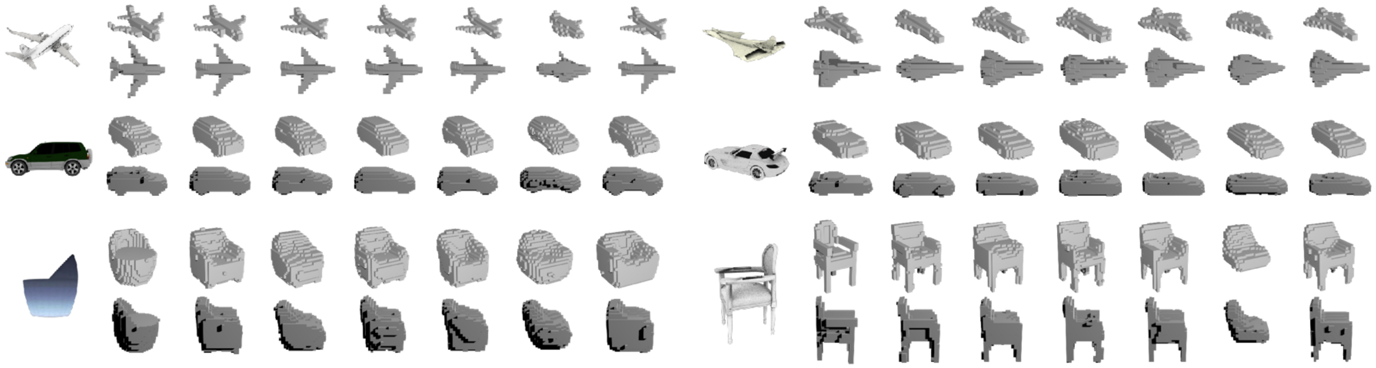


Fig. 3: Reconstructions on the ShapeNet dataset visualized using two representative views. Left to Right : Input, Ground-truth, 3D Training, Ours (Mask), Fusion (Depth), DRC (Depth), Fusion (Noisy Depth), DRC (Noisy Depth).

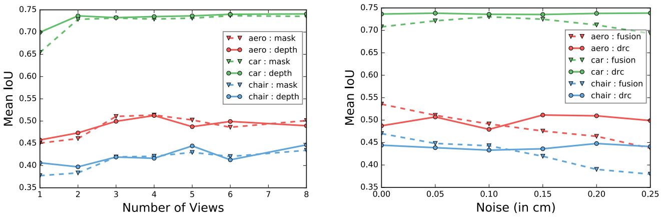


Fig. 4: Analysis of the per-category reconstruction performance. a) As we increase the number of views available per instance for training, the performance initially increases and saturates after few available views. b) As the amount of noise in depth observations used for training increases, the performance of our approach remains relatively consistent.

Training Data	3D	Mask		Depth		Depth (Noisy)	
class		Fusion	DRC	Fusion	DRC	Fusion	DRC
aero	0.57	-	0.50	0.54	0.49	0.46	0.51
car	0.76	-	0.73	0.71	0.74	0.71	0.74
chair	0.47	-	0.43	0.47	0.44	0.39	0.45

TABLE 1: Analysis of our method using mean IoU on ShapeNet.

signal. This is because of the use of a ray potential where the noisy signal only adds a small penalty to the true shape unlike in the case of depth fusion where the noisy signal is used to compute independent unary terms (see appendix for detailed discussion). We observe that even using only object masks leads to comparable performance to using depth but is worse when fewer views are available (Figure 4) and has some systematic errors *e.g.* the chair models cannot learn the concavities present in the seat using foreground mask information.

Ablations. When using muti-view supervision, it is informative to look at the change in performance as the number of available training views is increased. We show this result in Figure 4 and observe a performance gain as number of views initially increase but see the performance saturate after few views. We also note that depth observations are more informative than masks when very small number of views are used. Another aspect studied is the reconstruction performance when varying the amount of noise in depth observations. We observe that our approach is fairly robust to noise unlike the fusion approach. See appendix for further details, discussion and explanations of the trends.

5.2 Object Reconstruction on PASCAL VOC

We demonstrate the application of our DRC formulation on the PASCAL VOC dataset [41] where previous 3D supervised single-view reconstruction methods cannot be used due to lack of ground-truth training data. However, available annotations for segmentation masks and camera pose allow application of our framework.

Training Data. We use annotated pose (in PASCAL 3D [42]) and segmentation masks (from PASCAL VOC) as training signal for object reconstruction. To augment training data, we also use the Imagenet [43] objects from PASCAL 3D (using an off-the shelf instance segmentation method [44] to compute foreground masks on these). These annotations effectively provide an orthographic camera C_i for each training instance. Additionally, the annotated segmentation mask provides us with the observation O_i. We use the proposed view consistency loss on objects from the training set in PASCAL3D – the loss measures consistency of the predicted 3D shape given training RGB image I_i with the single observation-camera pair (O_i, C_i). Despite only one observation per instance, the shared prediction model can learn to predict complete 3D shapes.

Benchmark. PASCAL3D also provides annotations for (approximate) 3D shape of objects using a small set of CAD models (about 10 per category). Similar to previous approaches [5], [11], we use these annotations on the test set for benchmarking purposes. Note that since the same small set of models is shared across training and test objects, using the PASCAL3D models for training is likely to bias the evaluation. This makes our results incomparable to those reported in [11] where a model pretrained on ShapeNet data is fine-tuned on PASCAL3D using shapes from this small set of models as ground-truth. See appendix for further discussion.

Setup. The various baselines/variants studied are described below. Note that for all the learning based methods, we train a single category-agnostic CNN.

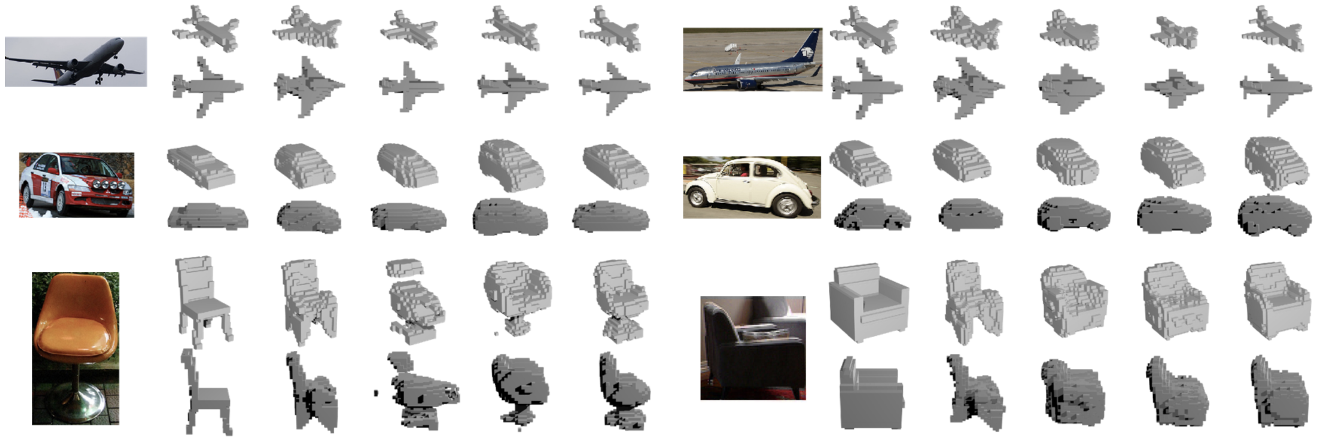


Fig. 5: PASCAL VOC reconstructions visualized using two representative views. Left to Right : Input, Ground-truth (as annotated in PASCAL 3D), Deformable Models [5], DRC (Pascal), Shapenet 3D, DRC (Joint).

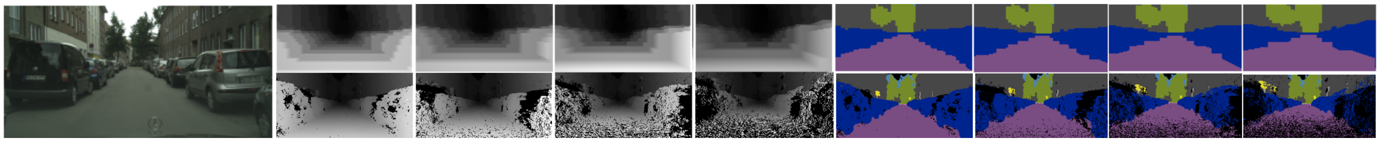


Fig. 6: Sample results on Cityscapes using ego-motion sequences for learning single image 3D reconstruction. Given a single input image (left), our model predicts voxel occupancy probabilities and per-voxel semantic class distribution. We use this prediction to render, in the top row, estimated disparity and semantics for a camera moving forward by 3, 6, 9, 12 metres respectively. The bottom row renders similar output but using a 2.5D representation of ground-truth pixel-wise disparity and pixel-wise semantic labels inferred by [45].

Method	aero	car	chair	mean
CSDM	0.40	0.60	0.29	0.43
DRC (PASCAL)	0.42	0.67	0.25	0.44
Shapenet 3D	0.53	0.67	0.33	0.51
DRC (Joint)	**0.55**	**0.72**	**0.34**	**0.54**

TABLE 2: Mean IoU on PASCAL VOC.

Category-Specific Deformable Models (CSDM). We compare to [5] in a setting where, unlike other methods, it uses ground-truth mask, keypoints to fit deformable 3D models.

ShapeNet 3D (with Realistic Rendering). To emulate the setup used by previous approaches *e.g.* [11], [12], we train a CNN on rendered ShapeNet images using cross entropy loss with the ground-truth CAD model. We attempt to bridge the domain gap by using more realistic renderings via random background/lighting variations [13] and initializing the convolution layers with a pretrained ResNet-18 model [46].

DRC (Pascal). We only use the PASCAL3D instances with pose, object mask annotations to train the CNN with the proposed view consistency loss.

DRC (Joint : ShapeNet 3D + Pascal). We pre-train a model on ShapeNet 3D data as above and finetune it using PASCAL3D using our view consistency loss.

Results. We present the comparisons of our approach to the baselines in Table 2 and visualize sample predictions in Figure 5. We observe that our model when trained using only PASCAL3D data, while being category agnostic and not using ground-truth annotations for testing, performs comparably to [5] which also uses similar training data. We observe that using the PASCAL data via the view consistency loss in addition to the ShapeNet 3D training data allows us to improve across categories as using real images for training removes some error modes that the CNN trained on synthetic data exhibits on real images. Note that the learning signals used in this setup were only approximate – the annotated pose, segmentation masks computed by [44] are not perfect and our method results in improvements despite these.

5.3 3D Scene Reconstruction from Ego-motion

The problem of scene reconstruction is an extremely challenging one. While previous approaches, using direct [34], multi-view [35], [36] or even no supervision [47] predict detailed 2.5D representations (pixelwise depth and/or surface normals), the task of single image 3D prediction has been largely unexplored for scenes. A prominent reason for this is the lack of supervisory data. Even though obtaining full 3D supervision might be difficult, obtaining multi-view observations may be more feasible. We present some preliminary explorations and apply our framework to learn single image 3D reconstruction for scenes by using driving sequences as supervision.

We use the cityscapes dataset [48] which has numerous 30-frame driving sequences with associated disparity images, ego-motion information and semantic labels[1]. We train a CNN to predict, from a single scene image, occupancies

[1] while only sparse frames are annotated, we use a semantic segmentation system [45] trained on these to obtain labels for other frames.

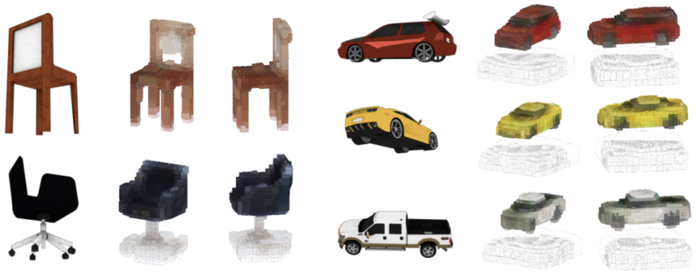


Fig. 7: Sample results on ShapeNet dataset using multiple RGB images as supervision for training. We show the input image (left) and the visualize 3D shape predicted using our learned model from two novel views. Best viewed in color.

and per-voxel semantic labels for a coarse voxel grid. We minimize the consistency loss function corresponding to the event cost in Eq. 9. To account for the large scale of scenes, our voxel grid does not have uniform cells, instead the size of the cells grows as we move away from the camera. See appendix for details, CNN architecture *etc.*.

We show qualitative results in Figure 6 and compare the coarse 3D representation inferred by our method with a detailed 2.5D representation by rendering inferred disparity and semantic segmentation images under simulated forward motion. The 3D representation, while coarse, is able to capture structure not visible in the original image (*e.g.* cars occluding other cars). While this is an encouraging result that demonstrates the possibility of going beyond 2.5D for scenes, there are several challenges that remain *e.g.* the pedestrians/moving cars violate the implicit static scene assumption, the scope of 3D data captured from the multiple views is limited in context of the whole scene and finally, one may never get observations for some aspects *e.g.* multi-view supervision cannot inform us that there is road below the cars parked on the side.

5.4 Object Reconstruction from RGB Supervision

We study the setting where only 2D color images of ShapeNet models are available as supervisory signal. In this scenario, our CNN predicts a per-voxel occupancy as well as a color value. We use the generalized event cost function from Eq. 10 to define the training loss. Some qualitative results are shown in Figure 7. We see the learned model can infer the correct shape as well as color, including the concavities in chairs, shading for hidden parts *etc.*. See appendix for more details and discussion on error modes *e.g.* artifacts below cars.

5.5 ShapeNet Reconstruction without Pose Supervision

We again consider the ShapeNet dataset, but in this scenario to demonstrate our ability to learn without requiring known poses associated with the available observations. We use the loss formulation defined in Section 3.6 and show that we can learn both shape and pose prediction without direct supervision for either.

Dataset. We use the same splits as in Section 5.1. We render the training objects under two settings - a) origin centred (as in Section 5.1), or b) randomly translated around the origin. As the camera is always at a fixed distance away from the origin, the first setting corresponds to training with a known camera translation, but unknown rotation. The second corresponds to training with both translation and rotation unknown. To have a common test set across various control setting, we use the origin centered renderings for our validation and test sets.

Setup. We use the same evaluation setup, hyperparameters, and network architectures as used in Section 5.1, and additionally train a pose CNN which predicts the (unknown) associated camera poses. As we jointly learn both shape ans pose prediction, the obtained reconstructions are in some arbitrary canonical frame different from the ShapeNet canonical frame. Therefore, before evaluating our results, we compute an optimal rotation to best align the predictions to the canonical ShapeNet frame – see appendix for more details.

In addition to evaluating the target setting where we learn without pose supervision, we also report control settings regarding training with known camera poses. This is similar to the setup in Section 5.1, with the difference that we instead use the pose-differentiable loss defined in Section 3.6.

Shape Prediction Results. Our results and the performance under various control settings with stronger supervision is reported in Table 3 and visualized in Figure 8. In general, we observe that the performance degrades gracefully as the amount of supervision available is reduced. This clearly indicates that our approach is able to learn single-view shape prediction despite the lack of either shape or pose information during training. As expected, we also observe that we cannot learn about concavities in chairs via consistency against mask validation images, though we can do so using depth images. We observe a noticeable performance drop in case of mask supervision with unknown translation, as this setting results in scale ambiguities which our evaluation does not account for *e.g.* we learn to predict larger cars, but further away, and this results in a low empirical score.

Pose Estimation Results. The results of our approach are reported in Table 3 and visualized in Figure 9. We report performance using the metrics used in [8] – median angular error and the fraction of instances with error less than a threshold of 30 degrees. We observe a similar trend for the task of pose prediction – that our approach performs comparably to directly supervised learning using ground-truth pose supervision. Interestingly, we often get lower median errors than the supervised setting. We attribute this to the different topologies of the loss functions. The squared L2 loss used in the supervised setting yields small gradients if the pose is almost correct. Our consistency loss however, would want the observation image to perfectly align with the shape via the predicted pose.

5.6 Learning from Online Product Images

Online images of products are a natural source of multi-view observations. While no associated shape or pose supervision is available in such setting, we demonstrate that we can learn 3D prediction systems using such data.

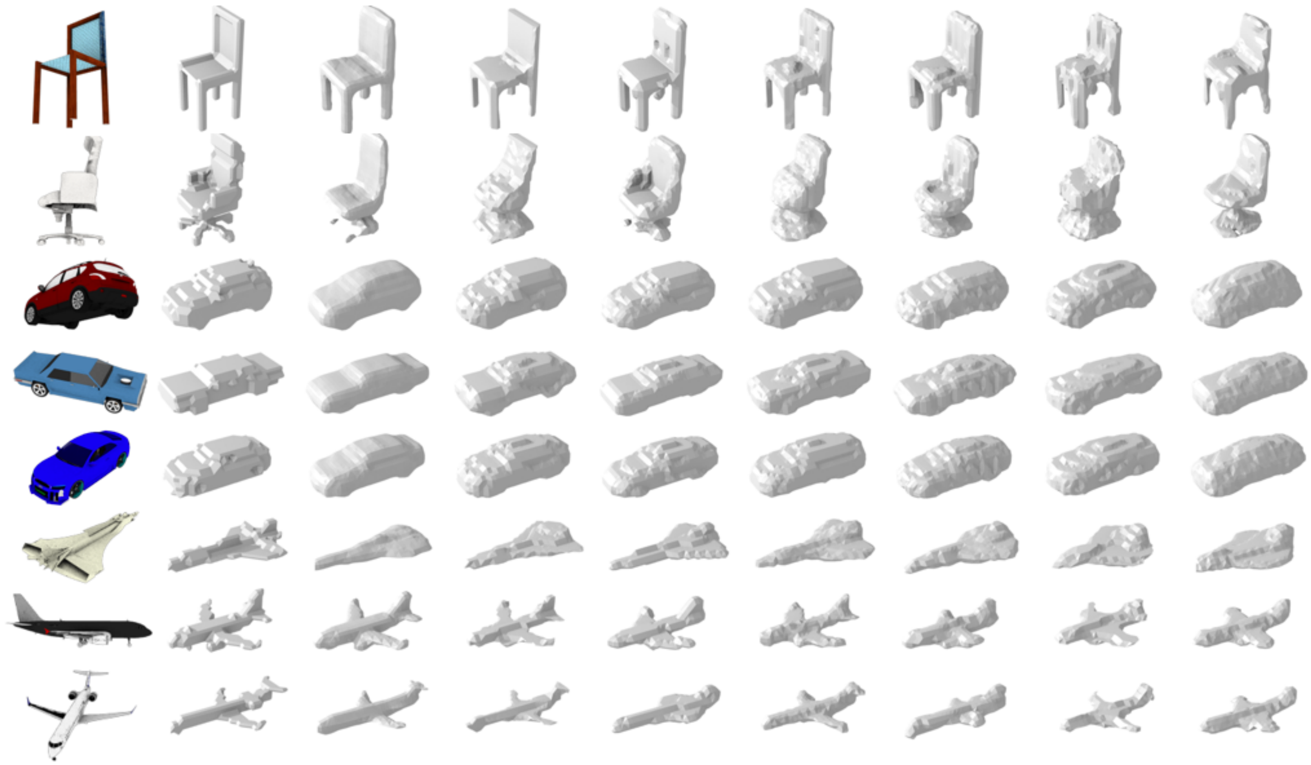


Fig. 8: Shape predictions on the validation set using a single RGB input image. We visualize the voxel occupancies by rendering the corresponding mesh (obtained via marching cubes) from a canonical pose. Left to Right: a) Input Image b) Ground-truth c) 3D Supervised Prediction d,e) Multi-view & Pose Supervision (Mask, Depth) f,g) Mult-view w/o Rotation Supervision (Mask, Depth), and h,i) Mult-view w/o Rotation and Translation Supervision (Mask, Depth).

Training Data	Multi-view & GT Pose		Multi-view w/o Rot		Multi-view w/o Rot & Trans	
class	Mask	Depth	Mask	Depth	Mask	Depth
aero	0.55	0.43	0.52	0.44	0.38	0.37
car	0.75	0.69	0.74	0.71	0.48	0.68
chair	0.42	0.45	0.40	0.43	0.35	0.37
mean	0.57	0.52	0.55	0.53	0.40	0.47

Training Data	GT Pose		MV w/o Rot Mask		MV w/o Rot Depth		MV w/o Rot & Trans Mask		MV w/o Rot & Trans Depth	
class	Acc	Err	Acc	Err	Acc	Err	Acc	Err	Acc	Err
aero	0.79	10.7	0.69	14.3	0.60	21.7	0.53	26.9	0.63	12.3
car	0.90	7.4	0.87	5.2	0.85	4.9	0.53	24.8	0.56	20.6
chair	0.85	11.2	0.81	7.8	0.83	8.6	0.55	24.0	0.62	19.1
mean	0.85	10.0	0.79	9.0	0.76	11.7	0.54	25.1	0.61	17.4

TABLE 3: Analysis of the performance for single-view shape (Left) and pose (Right) prediction. a) Shape Accuracy: Mean IoU on the test set using various supervision settings. b) Pose Accuracy/Error: $\text{Acc}_{\frac{\pi}{6}}$ and Med-Err across different supervision settings.

Dataset. We examined the 'chair' object category from the Stanford Online Products Dataset [49] which comprises of automatically downloaded images from eBay.com [50]. Since multiple images (views) of the same product are available, we can leverage our approach to learn from this data. As we also require associated foreground masks for these images, we use an out-of-the-box semantic segmentation system [51] to obtain these. However, the obtained segmentation masks are often incorrect. Additionally, many of the product images were not suited for our setting as they only comprised of a zoom-in of a small portion of the instance (*e.g.* chair wheel). We therefore manually selected images of unoccluded/untruncated instances with a reasonably accurate (though still noisy) predicted segmentation. We then used the object instances with at least 2 valid views for training. This results in a filtered dataset of $N = 282$ instances with $N_i = 3.65$ views on average per instance.

Results. We can apply our approach to learn from this dataset comprising of multiple views with associated (approximate) foreground masks. Since the camera intrinsics are unknown, we assume a default intrinsic matrix (see appendix). We then learn to predict the (unknown) translation and rotation via the pose CNN g_ϕ and the (unknown) shape via the shape CNN f_θ using the available multi-view supervision. Note that the learned CNNs are trained from scratch, and that we use the same architecture/hyperparameters as in the ShapeNet experiments.

Some results (on images of novel instances) using our learned CNN are visualized in Figure 10. We see that we can learn to predict meaningful 3D structure and infer the appropriate shape and pose corresponding to the input image. Since only foreground mask supervision is leveraged, we cannot learn to infer the concavities in shapes. We also observe confusion across poses which result in similar foreground masks. However, we feel that this result using training data derived from a challenging real world setting, concretely demonstrates our method's ability to learn despite the lack of direct shape or pose supervision.

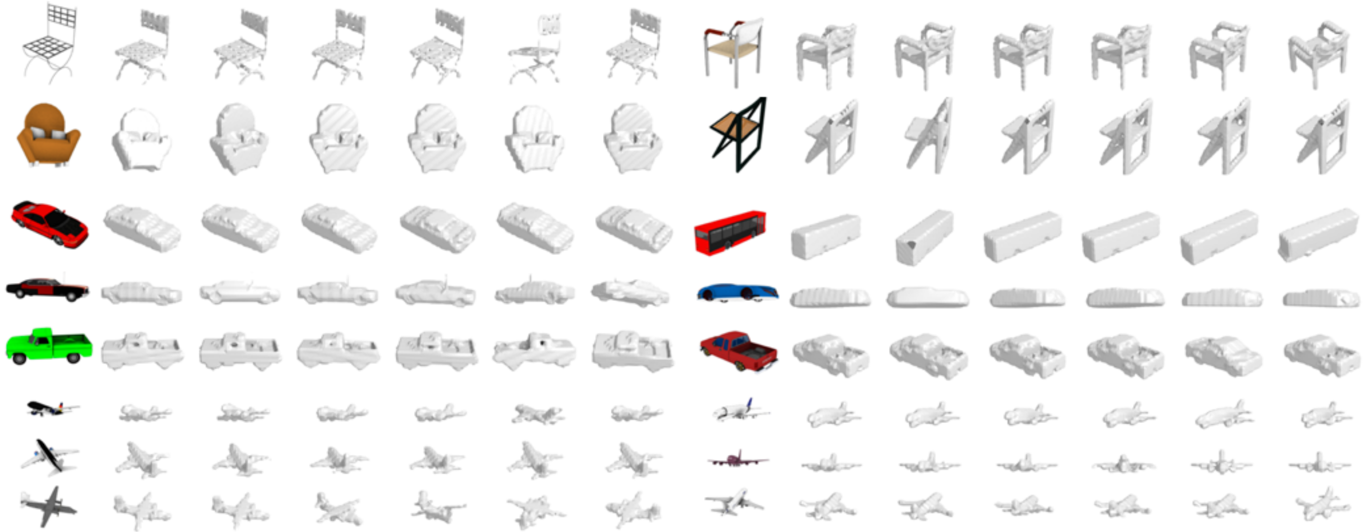


Fig. 9: Rotation predictions on a random subset of the validation images. For visualization, we render the ground-truth voxel occupancies using the corresponding rotation. Left to Right: a) Input Image b) Ground-truth Rotation c) GT Supervised Prediction d,e) Multi-view w/o Rot Supervision (Mask, Depth), and f,g) Multi-view w/o Rot and Trans Supervision (Mask, Depth).

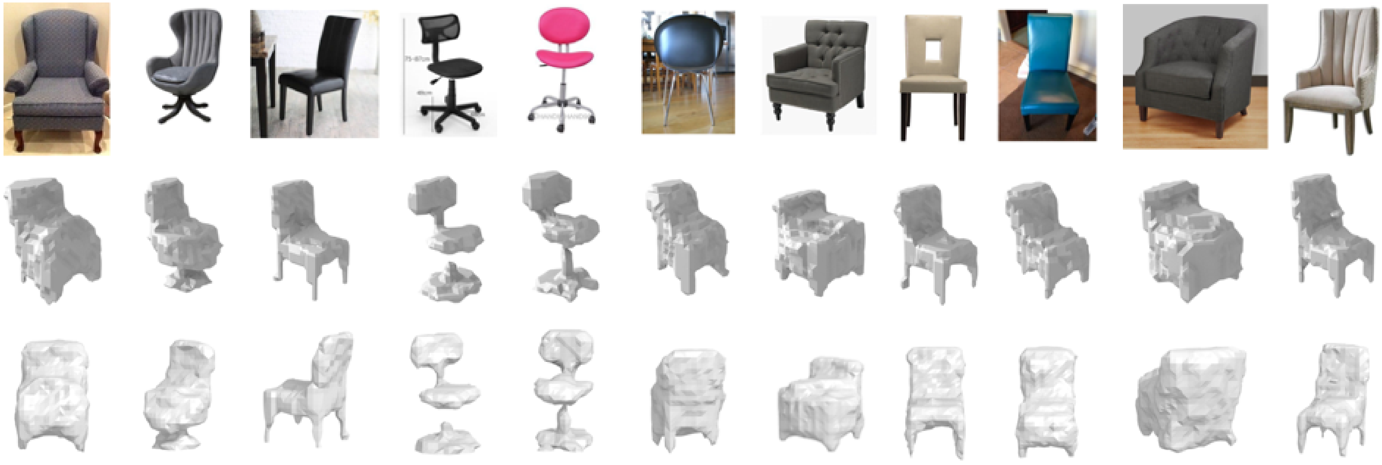


Fig. 10: Visualization of predictions using the Stanford Online Product Dataset. (Top) Input image. (Middle) Predicted shape in the emergent canonical pose. (Bottom) Predicted shape rotated according to the predicted pose.

To the best of our knowledge, this is the first such result and it represents an encouraging step forward.

6 Discussion

We have presented a differentiable formulation for consistency between a 3D shape and a 2D observation and demonstrated its applications for learning single-view reconstruction in various scenarios. These are, however, only the initial steps and a number of challenges are yet to be addressed. Our formulation is applicable to voxel-occupancy based representations and an interesting direction is to extend these ideas to alternate representations which allow finer predictions *e.g.* meshes or octrees. Finally, while our approach allows us to bypass the availability of ground-truth 3D information for training, a benchmark dataset is still required for evaluation which may be challenging for scenarios like scene reconstruction.

Acknowledgments

We thank Saurabh Gupta and David Fouhey for insightful discussions, and the anonymous reviewers for the helpful comments. This work was supported in part by Intel/NSF VEC award IIS-1539099, NSF Award IIS-1212798, and the Berkeley Fellowship to ST. We gratefully acknowledge NVIDIA corporation for the donation of Tesla GPUs used for this research.

References

[1] J. J. Gibson, “The ecological approach to visual perception.” 1979. 1

[2] K. N. Kutulakos and S. M. Seitz, “A theory of shape by space carving,” *IJCV*, 2000. 1

[3] V. Blanz and T. Vetter, “A morphable model for the synthesis of 3d faces,” in *SIGGRAPH*, 1999. 2

[4] T. J. Cashman and A. W. Fitzgibbon, “What shape are dolphins? building 3d morphable models from 2d images,” *TPAMI*, 2013. 2

[5] S. Tulsiani, A. Kar, J. Carreira, and J. Malik, “Learning category-specific deformable 3d models for object reconstruction,” *TPAMI*, 2016. 2, 7, 8

[6] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," in *CVPR*, 2014. 2

[7] B. Hariharan, P. Arbeláez, R. Girshick, and J. Malik, "Hypercolumns for object segmentation and fine-grained localization," in *CVPR*, 2015. 2

[8] S. Tulsiani and J. Malik, "Viewpoints and keypoints," in *CVPR*, 2015. 2, 9

[9] 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. 2

[10] S. Vicente, J. Carreira, L. Agapito, and J. Batista, "Reconstructing pascal voc," in *CVPR*, 2014. 2

[11] 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. 2, 7, 8

[12] R. Girdhar, D. Fouhey, M. Rodriguez, and A. Gupta, "Learning a predictable and generative vector representation for objects," in *ECCV*, 2016. 2, 8

[13] 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*, 2015. 2, 8

[14] R. Zhu, H. Kiani, C. Wang, and S. Lucey, "Rethinking reprojection: Closing the loop for pose-aware shape reconstruction from a single image," in *ICCV*, 2017. 2

[15] J. Wu, Y. Wang, T. Xue, X. Sun, W. T. Freeman, and J. B. Tenenbaum, "MarrNet: 3D Shape Reconstruction via 2.5D Sketches," in *NIPS*, 2017. 2

[16] A. Kar, C. Häne, and J. Malik, "Learning a multi-view stereo machine," in *Advances in Neural Information Processing Systems*, 2017. 2

[17] H. Fan, H. Su, and L. J. Guibas, "A point set generation network for 3d object reconstruction from a single image," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2017. 2

[18] C. Häne, S. Tulsiani, and J. Malik, "Hierarchical surface prediction for 3d object reconstruction," in *3DV*, 2017. 2

[19] M. Tatarchenko, A. Dosovitskiy, and T. Brox, "Octree generating networks: Efficient convolutional architectures for high-resolution 3d outputs," in *ICCV*, 2017. 2

[20] A. Kanazawa, M. J. Black, D. W. Jacobs, and J. Malik, "End-to-end recovery of human shape and pose," in *CVPR*, 2018. 2

[21] A. Kurenkov, J. Ji, A. Garg, V. Mehta, J. Gwak, C. Choy, and S. Savarese, "Deformnet: Free-form deformation network for 3d shape reconstruction from a single image," *arXiv preprint arXiv:1708.04672*, 2017. 2

[22] S. Laine, T. Karras, T. Aila, A. Herva, S. Saito, R. Yu, H. Li, and J. Lehtinen, "Production-level facial performance capture using deep convolutional neural networks," in *Proceedings of the ACM SIGGRAPH/Eurographics Symposium on Computer Animation*. ACM, 2017, p. 10. 2

[23] A. Laurentini, "The visual hull concept for silhouette-based image understanding," *TPAMI*, 1994. 2

[24] A. Broadhurst, T. W. Drummond, and R. Cipolla, "A probabilistic framework for space carving," in *ICCV*, 2001. 2, 3

[25] W. Matusik, C. Buehler, R. Raskar, S. J. Gortler, and L. McMillan, "Image-based visual hulls," in *SIGGRAPH*, 2000. 2

[26] J. De Bonet and P. Viola, "Roxels: Responsibility weighted 3d volume reconstruction," in *ICCV*, 1999. 2

[27] S. Liu and D. B. Cooper, "Ray markov random fields for image-based 3d modeling: model and efficient inference," in *CVPR*, 2010. 2, 3

[28] P. Gargallo, P. Sturm, and S. Pujades, "An occupancy–depth generative model of multi-view images," in *ACCV*, 2007. 2

[29] O. J. Woodford and G. Vogiatzis, "A generative model for online depth fusion," in *ECCV*, 2012. 2, 3

[30] A. O. Ulusoy, A. Geiger, and M. J. Black, "Towards probabilistic volumetric reconstruction using ray potentials," in *3DV*, 2015. 2, 3

[31] A. Kundu, Y. Li, F. Dellaert, F. Li, and J. M. Rehg, "Joint semantic segmentation and 3d reconstruction from monocular video," in *ECCV*, 2014. 2

[32] N. Savinov, C. Hane, L. Ladicky, and M. Pollefeys, "Semantic 3d reconstruction with continuous regularization and ray potentials using a visibility consistency constraint," in *CVPR*, 2016. 2, 3, 4

[33] N. Savinov, C. Häne, M. Pollefeys *et al.*, "Discrete optimization of ray potentials for semantic 3d reconstruction," in *CVPR*, 2015. 2, 3, 4

[34] D. Eigen and R. Fergus, "Predicting depth, surface normals and semantic labels with a common multi-scale convolutional architecture," in *ICCV*, 2015. 2, 8

[35] R. Garg and I. Reid, "Unsupervised cnn for single view depth estimation: Geometry to the rescue," in *ECCV*, 2016. 2, 8

[36] C. Godard, O. Mac Aodha, and G. J. Brostow, "Unsupervised monocular depth estimation with left-right consistency," in *CVPR*, 2017. 2, 8

[37] T. Zhou, M. Brown, N. Snavely, and D. Lowe, "Unsupervised learning of depth and ego-motion from video," in *CVPR*, 2017. 2

[38] D. J. Rezende, S. A. Eslami, S. Mohamed, P. Battaglia, M. Jaderberg, and N. Heess, "Unsupervised learning of 3d structure from images," in *NIPS*, 2016. 4

[39] X. Yan, J. Yang, E. Yumer, Y. Guo, and H. Lee, "Perspective transformer nets: Learning single-view 3d object reconstruction without 3d supervision," in *NIPS*, 2016. 4, 5

[40] A. X. Chang, T. Funkhouser, L. 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," Tech. Rep. arXiv:1512.03012 [cs.GR], 2015. 6

[41] M. Everingham, L. Van Gool, C. K. Williams, J. Winn, and A. Zisserman, "The pascal visual object classes (voc) challenge," *IJCV*, 2010. 7

[42] Y. Xiang, R. Mottaghi, and S. Savarese, "Beyond pascal: A benchmark for 3d object detection in the wild," in *WACV*, 2014. 7

[43] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein *et al.*, "Imagenet large scale visual recognition challenge," *IJCV*, 2015. 7

[44] K. Li, B. Hariharan, and J. Malik, "Iterative instance segmentation," in *CVPR*, 2016. 7, 8

[45] F. Yu and V. Koltun, "Multi-scale context aggregation by dilated convolutions," in *ICLR*, 2016. 8

[46] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *CVPR*, 2016. 8

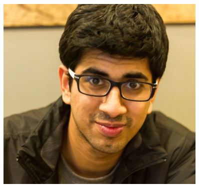
[47] D. F. Fouhey, W. Hussain, A. Gupta, and M. Hebert, "Single image 3D without a single 3D image," in *ICCV*, 2015. 8

[48] M. Cordts, M. Omran, S. Ramos, T. Rehfeld, M. Enzweiler, R. Benenson, U. Franke, S. Roth, and B. Schiele, "The cityscapes dataset for semantic urban scene understanding," in *CVPR*, 2016. 8

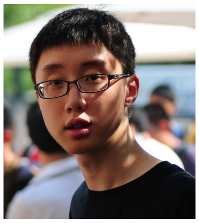
[49] H. O. Song, Y. Xiang, S. Jegelka, and S. Savarese, "Deep metric learning via lifted structured feature embedding," in *CVPR*, 2016. 10

[50] https://www.ebay.com/. 10

[51] L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille, "Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs," *TPAMI*, 2017. 10



Shubham Tulsiani received his PhD degree in computer science from University of California, Berkeley. His research interests lie at the intersection of recognition, pose estimation and 3D reconstruction from a single image.



Tinghui Zhou received his PhD degree in computer science from University of California, Berkeley. His research interests lie at the intersection of computer vision and graphics, with a focus on learning-based approaches for problems where direct labels are difficult or impossible to obtain.

Alexei A. Efros joined UC Berkeley in 2013. Prior to that, he was nine years on the faculty of Carnegie Mellon University, and has also been affiliated with INRIA and University of Oxford. His research is in the area of computer vision and computer graphics, especially at the intersection of the two. He is particularly interested in using data-driven techniques to tackle problems where large quantities of unlabeled visual data are readily available. Efros received his PhD in 2003 from UC Berkeley. He is a recipient of CVPR Best Paper Award (2006), NSF CAREER award (2006), Sloan Fellowship (2008), Guggenheim Fellowship (2008), Okawa Grant (2008), Finmeccanica Career Development Chair (2010), SIGGRAPH Significant New Researcher Award (2010), ECCV Best Paper Honorable Mention (2010), 3 Helmholtz Test-of-Time Prizes (1999,2003,2005), and the ACM Prize in Computing (2016).

Jitendra Malik received the B.Tech degree in Electrical Engineering from Indian Institute of Technology, Kanpur in 1980 and the PhD degree in Computer Science from Stanford University in 1985. In January 1986, he joined the university of California at Berkeley, where he is currently the Arthur J. Chick Professor in the Department of Electrical Engineering and Computer Sciences. He is also on the faculty of the department of Bioengineering, and the Cognitive Science and Vision Science groups. During 2002-2004 he served as the Chair of the Computer Science Division, and as the Department Chair of EECS during 2004-2006 as well as 2016-2017. Since January 2018, he is also Research Director and Site Lead of Facebook AI Research in Menlo Park.