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

Shubham Tulsiani, Tinghui Zhou, Alexei A. Efros, Jitendra Malik
University of California, Berkeley
{shubhtuls,tinghuiz,efros,malik}@eecs.berkeley.edu

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.*

1. Introduction

When is a solid 3D shape consistent with a 2D image? If it is not, how do we change it to make it more so? One way this problem has been traditionally addressed is by space carving [21]. 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.

But what if we want to extend this notion of consistency to the 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 formulation 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 question of finding a differential formulation for ray consistency is mathematically interesting in and of itself. Luckily, it is also extremely useful as it allows us to connect the concepts in 3D geometry with the latest developments in machine learning. While classic 3D reconstruction methods require large number of 2D views of the same physical 3D object, learning-based methods are able to take advantage of their past experience and thus only require a small number of views for each physical object being trained. Finally, when the system is done learning, it is able to give an estimate of the 3D shape of a novel object from only a *single* image, something that classic methods are incapable of doing. 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.

2. Related Work

Object Reconstruction from Image-based Annotations. Blanz and Vetter [2] 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*. [33] extended similar ideas to more general categories and leveraged recognition systems [15, 18, 34] to automate test-time inference. Wu *et al*. [37], 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*. [36] 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 syn-

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

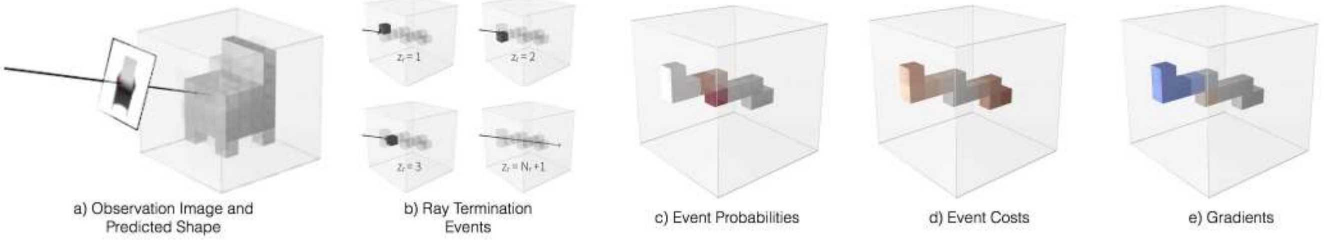


Figure 1: Visualization of various aspects of our Differentiable Ray Consistency formulation. a) Predicted 3D shape represented as probabilistic occupancies and the observation image where we consider consistency between the predicted shape and the ray corresponding to the highlighted pixel. b) Ray termination 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.

thetic training data has resulted in applications for object reconstruction. Choy *et al*. [6] learned a CNN to predict a voxel representation using a single (or multiple) input image(s). Girdhar *et al*. [14] also presented similar results for single-view object reconstruction, while also demonstrating some results on real images by using realistic rendering techniques [31] for generating training data. 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 [22] formalizes the notion of consistency between a 3D shape and observed object masks. Techniques based on this concept [3, 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 [8, 24] or inference in a generative model [13]. Ulusoy *et al*. [35] 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 [20, 29, 30].

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 [9], recent approaches based on multi-view supervision have shown promise in achieving similar (and sometimes even better) performance. Garg *et al*. [12] and Godard *et al*. [16] 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*. [41] 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. 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' 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. We aim for the shape to be consistent with some available observation O. This 'observation' can take various forms *e.g.* a depth image, or an object foreground mask – 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 camera C.

Our view consistency loss, using the notations mentioned above, is of the form $L(x; (O, C))$. 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 proceed to present a differentiable formulation for ray consistency, the various aspects of which are visualized in Figure 1. 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.

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.

We can therefore 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 motivation is that a probabilistic occupancy model (instantiated by the shape parameters x) induces a distribution of *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 1.

Event Probabilities. Given the occupancy probabilities x, we want to infer the probability $p(z_r = i)$. 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 correspnds 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.

$$p(z_r = i) = \begin{cases} (1 - x_i^r) \prod\limits_{j=1}^{i-1} x_j^r, & \text{if } i \leq N_r \\ \prod\limits_{j=1}^{N_r} x_j^r, & \text{if } i = N_r + 1 \end{cases} \quad (2)$$

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_r. We use a simple distance measure between observed distance and event-induced distance to define $\psi_r(i)$.

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

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 [26, 39] have also been proposed to specifically address the case of learning object reconstruction from foreground masks. These approaches, either though a learned [26] or fixed [39] reprojection function, minimize the discrepancy between the observed mask and the reprojected predictions. We show in the appendix [1] 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 $p(z_r)$. 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.

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

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

Recall that the event probabilities $p(z_r = i)$ 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-consisteny 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 for the i^{th} voxel on the ray's path – p_i^r.

To incorporate consistency between these, we can extend $L_r(x)$ 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.

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

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_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.

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

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 [1].

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 .

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 $\sum\limits_i \sum\limits_k L(f_\theta(I_i); (O_k^i, C_k^i))$ *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 SGD iteration.

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. Finally, we show qualitative results for using multiple color image observations as supervision for single-view reconstruction.

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 [5] 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 [1] 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 [1]) :

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

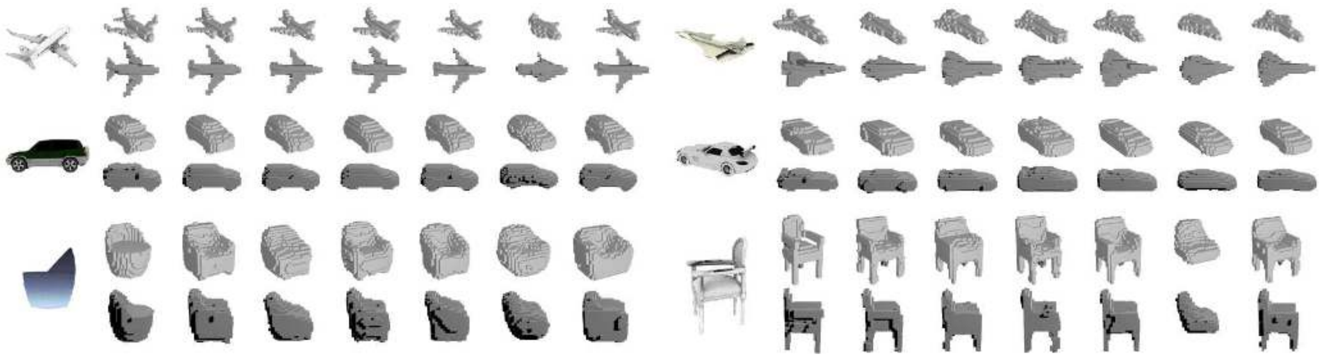
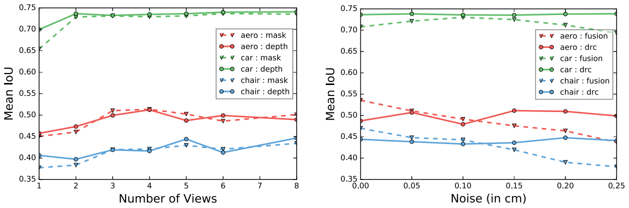


Figure 2: Reconstructions on the ShapeNet dataset visualized using two representative views. Left to Right : Input, Ground-truth, 3D Training, Ours (Mask), Fusion (Depth), DRC (Depth), Fusion (Noisy Depth), DRC (Noisy Depth).



(a) Number of training views **(b)** Amount of noise

Figure 3: 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.

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 [1] 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 2. 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 much more robust to noisy training 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 [1] 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 3) 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 3 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 [1] 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 [10] 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 [38]) and segmentation masks (from PASCAL VOC) as training signal for object reconstruction. To augment

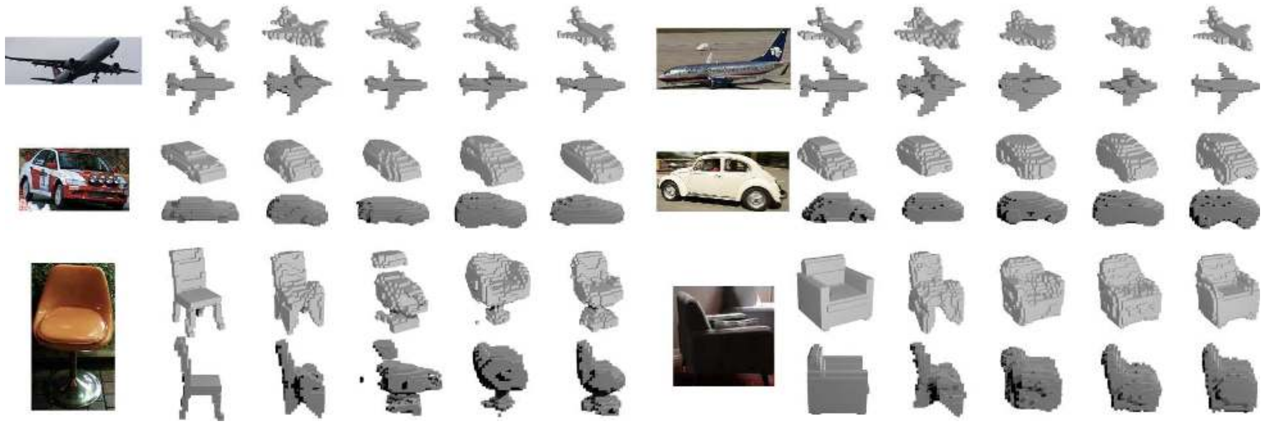


Figure 4: PASCAL VOC reconstructions visualized using two representative views. Left to Right : Input, Ground-truth (as annotated in PASCAL 3D), Deformable Models [33], DRC (Pascal), Shapenet 3D, DRC (Joint).

training data, we also use the Imagenet [28] objects from PASCAL 3D (using an off-the shelf instance segmentation method [23] 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 [6, 33], 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 [6] 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 [1] 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.

Category-Specific Deformable Models (CSDM). We compare to [33] 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.* [6, 14], 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 [31] and initializing the convolution layers with a pretrained ResNet-18 model [19].

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.

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 4. 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 [33] 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 [23] 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 di-

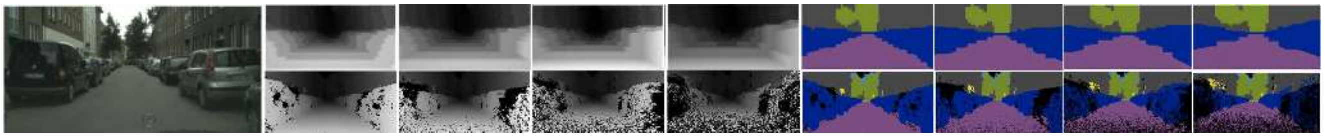


Figure 5: Sample results on Cityscapes using ego-motion sequences for learning single image 3D reconstruction. Given a single input image (left), our model predicts voxel occupancy probabilities and per-voxel semantic class distribution. We use this prediction to render, in the top row, estimated disparity 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 [40].

rect [9], multi-view [12, 16] or even no supervision [11] 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 [7] 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 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 [1] for details, CNN architecture *etc*.

We show qualitative results in Figure 5 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 6. 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 [1] for more details and discussion on error modes *e.g.* artifacts below cars.

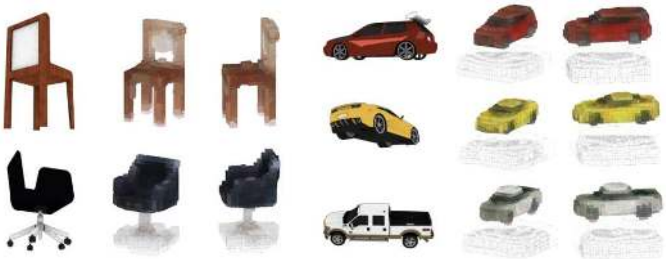


Figure 6: Sample results on ShapeNet dataset using multiple RGB images as supervision for training. We show the input image (left) and the visualize 3D shape predicted using our learned model from two novel views. Best viewed in color.

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.* [17, 27, 32]. Additionally, we assume a known camera transformation across views. While this is a realistic assumption from the perspective of agents, relaxing this might further allow learning from web-scale data. 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.

Acknowledgements. We thank the anonymous reviewers for 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.

[1] while only sparse frames are annotated, we use a semantic segmentation system [40] trained on these to obtain labels for other frames

References

[1] https://shubhtuls.github.io/drc/appendix.pdf. 4, 5, 6, 7, 8

[2] V. Blanz and T. Vetter. A morphable model for the synthesis of 3d faces. In *SIGGRAPH*, 1999. 1

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

[4] T. J. Cashman and A. W. Fitzgibbon. What shape are dolphins? building 3d morphable models from 2d images. *TPAMI*, 2013. 1

[5] 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. Technical Report arXiv:1512.03012 [cs.GR], 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. 2, 7

[7] 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

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

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

[10] M. Everingham, L. Van Gool, C. K. Williams, J. Winn, and A. Zisserman. The pascal visual object classes (voc) challenge. *IJCV*, 2010. 6

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

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

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

[14] R. Girdhar, D. Fouhey, M. Rodriguez, and A. Gupta. Learning a predictable and generative vector representation for objects. In *ECCV*, 2016. 2, 7

[15] R. Girshick, J. Donahue, T. Darrell, and J. Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. In *CVPR*, 2014. 1

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

[17] C. Häne, S. Tulsiani, and J. Malik. Hierarchical surface prediction for 3d object reconstruction. *arXiv preprint arXiv:1704.00710*, 2017. 8

[18] B. Hariharan, P. Arbeláez, R. Girshick, and J. Malik. Hypercolumns for object segmentation and fine-grained localization. In *CVPR*, 2015. 1

[19] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *CVPR*, 2016. 7

[20] 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

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

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

[23] K. Li, B. Hariharan, and J. Malik. Iterative instance segmentation. In *CVPR*, 2016. 7

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

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

[26] 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

[27] G. Riegler, A. O. Ulusoy, H. Bischof, and A. Geiger. Octnetfusion: Learning depth fusion from data. *arXiv preprint arXiv:1704.01047*, 2017. 8

[28] 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

[29] 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

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

[31] 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, 7

[32] M. Tatarchenko, A. Dosovitskiy, and T. Brox. Octree generating networks: Efficient convolutional architectures for high-resolution 3d outputs. *arXiv preprint arXiv:1703.09438*, 2017. 8

[33] S. Tulsiani, A. Kar, J. Carreira, and J. Malik. Learning category-specific deformable 3d models for object reconstruction. *TPAMI*, 2016. 1, 7

[34] S. Tulsiani and J. Malik. Viewpoints and keypoints. In *CVPR*, 2015. 1

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

[36] S. Vicente, J. Carreira, L. Agapito, and J. Batista. Reconstructing pascal voc. In *CVPR*, 2014. 1

[37] 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. 1

[38] Y. Xiang, R. Mottaghi, and S. Savarese. Beyond pascal: A benchmark for 3d object detection in the wild. In *WACV*, 2014. 6

[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

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

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