Supplemental: Multi-view Consistency as Supervisory Signal for Learning Shape and Pose Prediction

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

A1. Loss Formulation

We briefly described, in the main text, the formulation of a view consistency loss $L(\bar{x}, C; V)$ that measures the inconsistency between a shape $\bar{x}$ viewed according to camera C and a depth/mask image V. Crucially, this loss was differentiable w.r.t both, pose and shape. As indicated in the main text, our formulation builds upon previously proposed differentiable ray consistency formulation [3] with some innovations to make it differentiable w.r.t pose. For presentation clarity, we first present our full formulation, and later discuss its relation to the previous techniques (a similar discussion can also be found in the main text).

Notation. The (predicted) shape representation $\bar{x}$ is parametrized as occupancy probabilities of cells in a 3D grid. We use the convention that a particular value in the tensor x corresponds to the probability of the corresponding voxel being $empty$. The verification image V that we consider can be a depth or foreground mask image. Finally, the camera C is parametrized via the intrinsic matrix K, and extrinsic matrix defined using a translation t and rotation R.

Per-pixel Error as Ray Consistency Cost. We consider the verification image V one pixel at a time and define the per-pixel error using a (differentiable) ray consistency cost. Each pixel $p \equiv (u, v)$ has an associated value v_p *e.g.* in the case of a depth image, v_p is the recorded depth at the pixel p. Additionally, each pixel corresponds to a ray originating from the camera centre and crossing the image plane at (u, v). Given the camera parameters C and shape $\bar{x}$, we can examine the ray corresponding to this pixel and check whether it is consistent with the observation o_p. We define a ray consistency cost function $L_p(\bar{x}, C; v_p)$ to capture the error associated with the pixel p. The view consistency loss can then be defined as the sum of per-pixel errors $L(\bar{x}, C; V) \equiv \sum_p L_p(\bar{x}, C; v_p)$.

Sampling Occupancies along a Ray. To define the consistency cost function $L_p(\bar{x}, C; v_p)$, we need to consider the ray as it is passing through the probabilistically occupied voxel grid $\bar{x}$. We do so by looking at discrete points sampled along the ray. Concretely, we sample points at a pre-defined set of $N = 80$ depth values $\{d_i | 1 \leq i \leq N\}$ along each ray. We denote by x_i^p the occupancy value at the i^{th} sample along this ray. To determine x_i^p, we look at the 3D coordinate of the corresponding point. Note that this can be determined using the camera parameters. Given the camera intrinsic parameters (f_u, f_v, u_0, v_0), the ray corresponding to the image pixel (u, v) travels along the direction $(\frac{u-u_0}{f_u}, \frac{v-v_0}{f_v}, 1)$ in the camera frame. Therefore, the i^{th} point along the ray, in the camera coordinate frame, is located at $l_i \equiv (\frac{u-u_0}{f_u} d_i, \frac{v-v_0}{f_v} d_i, d_i)$. Then, given the camera extrinsics (R, t), we can compute the location of his point in the coordinate frame of the predicted shape $\bar{x}$. Finally, we can use trilinear sampling to determine the occupancy at this point by sampling the value at this using the occupancies $\bar{x}$. Denoting by $T(G, pt)$ a function that samples a volumetric grid G at a location pt, we can compute the occupancy sampled at the i^{th} as below.

$$x_i^p = \mathcal{T}(\bar{x}, R \times (l_i + t)\ ); \tag{1}$$

$$l_i \equiv (\frac{u - u_0}{f_u} d_i, \frac{v - v_0}{f_v} d_i, d_i) \tag{2}$$

Note that since the trilinear sampling function T is differentiable w.r.t its arguments, the sampled occupancy x_i^p is differentiable w.r.t the shape $\bar{x}$ and the camera C.

Probabilistic Ray Tracing. We have so far considered the ray associated with a pixel p and computed samples with corresponding occupancy probabilities along it. We now trace this ray as it travels forward and use the samples along the ray as checkpoints. In particular, we assume that when the ray reaches the point corresponding to the i^{th} sample, it either travels forward or terminates at that point. Conditioned on the ray reaching this sample, it travels forward with probability x_i^p and terminates with likelihood $(1 - x_i^p)$. We denote by $\mathbf{z}^p \in \{1, \cdots, N+1\}$ a random variable corresponding to the sample index where the ray (probabilistically) terminates, where $z^p = N + 1$ implies that the ray escapes. We call these probabilistic ray terminations as *ray termination events*

and can compute the probability distribution $q(z_p)$ for these.

$$q(z^p = i) = (1 - x_i^p) \prod_{j=1}^{i-1} x_j^p \quad \forall (i \leq N); \tag{3}$$

$$q(z^p = N + 1) = \prod_{j=1}^{N} x_j^p; \tag{4}$$

Event Costs. Each event corresponds to the ray terminating at a particular point. It is possible to assign a cost to each event based on how inconsistent it is to w.r.t the pixel value v_p. If we have a depth observation $v_p \equiv d_p$, we can penalize the event $z^p = i$ by measuring the difference between d_p and d_i. Alternatively, if we have a foreground image observation *i.e.* $v_p \equiv s_p \in \{0, 1\}$ where $s_p = 1$ implies a foreground pixel, we can penalize all events which correspond to a different observation. We can therefore define a cost function $\psi_p(i)$ which computes the cost associated with event $z_p = i$.

$$\psi_p^{depth}(i) = |d_p - d_i|; \tag{5}$$

$$\psi_p^{mask}(i) = |s_p - \mathbb{1}(i \leq N)|; \tag{6}$$

Ray Consistency Cost. We formulated the concept of ray termination events, and associated a probability and a cost to these. The ray consistency cost is then defined as the expected event cost.

$$L_p(\bar{x}, C; v_p) = \mathop{\mathbb{E}}_{z_p} \psi_p(z_p) = \sum_{i=1}^{N} q(z_p = i)\psi_p(i) \tag{7}$$

Note that the probabilities $q(z_p = i)$ are a differentiable function of x_p which, in turn, is a differentiable function of shape $\bar{x}$ and camera C. The view consistency loss, which is simply a sum of multiple ray consistency terms, is therefore also differentiable w.r.t the shape and pose.

Relation to Previous Work. The formulation presented draws upon previous work on differentiable ray consistency [3] and leverages the notions of probabilistic ray termination events and event costs to define the ray consistency loss. A crucial difference however, is that we, using trilinear sampling, compute occupancies for point samples along the ray instead of directly using the occupancies of the voxels in the ray's path. Unlike their formulation, this allows our loss to also be differentiable w.r.t pose which is a crucial requirement for our scenario. Yan *et al*. [4] also use a similar sampling trick but their formulation is restricted to specifically using mask verification images and is additionally not leveraged for learning about pose. Tulsiani *et al*. [3] also discuss how their formulation can be adapted to use more general verification images *e.g*. color, semantics *etc*. using additional per-voxel predictions. While our experiments presented in the main text focus on leveraging mask or depth verification images, a similar generalization is possible for our formulation.

A2. Online Product Images Dataset

We used the 'chair' object category from the Stanford Online Products Dataset [2]. To obtain associated foreground masks for these images, the semantic segmentation system from Chen *et al*. [1], where for each image, the mask was indicated by the pixels with most likely class label as 'chair'. As the obtained segmentation masks were often incorrect, or objects in the images truncated/occluded, we manually selected images of unoccluded/untruncated instances with a reasonably accurate (though still noisy) predicted segmentation. For our training, we only used the object instances with atleast 2 valid views. This resulting dataset is visualized in Figure 1. The result visualizations shown in the main text are using images from the original online products dataset [2], but correspond to objects instances that were not used for our training (due to lack of a sufficient number of valid views).

References

[1] 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. 2

[2] H. O. Song, Y. Xiang, S. Jegelka, and S. Savarese. Deep metric learning via lifted structured feature embedding. In *CVPR*, 2016. 2

[3] S. Tulsiani, T. Zhou, A. A. Efros, and J. Malik. Multi-view supervision for single-view reconstruction via differentiable ray consistency. In *CVPR*, 2017. 1, 2

[4] 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. 2