Diverse Plausible Shape Completions from Ambiguous Depth Images

Brad Saund and Dmitry Berenson
Robotics Institute
University of Michigan
bsaund@umich.edu, dmitryb@umich.edu

Abstract: We propose PSSNet, a network architecture for generating diverse plausible 3D reconstructions from a single 2.5D depth image. Existing methods tend to produce only small variations on a single shape, even when multiple shapes are consistent with an observation. To obtain diversity we alter a Variational Auto Encoder by providing a learned shape bounding box feature as side information during training. Since these features are known during training, we are able to add a supervised loss to the encoder and noiseless values to the decoder. To evaluate, we sample a set of completions from a network, construct a set of plausible shape matches for each test observation, and compare using our plausible diversity metric defined over sets of shapes. We perform experiments using Shapenet mugs and partially-occluded YCB objects and find that our method performs comparably in datasets with little ambiguity, and outperforms existing methods when many shapes plausibly fit an observed depth image. We demonstrate one use for PSSNet on a physical robot when grasping objects in occlusion and clutter.

1 Introduction

You look into a cabinet and see a coffee mug on the shelf. Though you only observe the front of the shell you have a rich prior of shapes and so can infer the occluded structure of the mug. Now suppose the handle is facing towards the back of the shelf, hidden from view. You may imagine scenarios where the handle is on the left, on the right, straight back, or perhaps there no handle at all. We propose a neural network architecture for generating these diverse samples over plausible completed shapes (Fig. 1).

More specifically, we generate a set of possible 3D shapes from a 2.5D depth image, such as that provided by a Kinect sensor. There is inherent ambiguity in this process as it is impossible to know the true occupancy of occluded space. We thus seek an algorithm which produces a set of plausible 3D shape estimates from the observed data.

Broadly, researchers have attempted two approaches when inferring 3D structure from a 2.5D depth image. Shape matching optimizes a model pose, potentially with uncertainty [1, 2], thus requiring meshes of any potential object, limiting their ability to generalize. Learning-based methods, such as Variational Auto Encoders (VAE) [3], only require meshes during training and generate visually-pleasing shapes, but are optimized and evaluated on a single completion without consideration of other plausible completions.

Rather than operating on a single maximal-likelihood guess of the world, many robotics algorithms model and plan over a belief over worlds [4], thus we propose the Plausible Shape Sampling Network PSSNet, capable of generating diverse shape completions when multiple plausible shapes could fit a depth image. Our *key insight* is a restructuring of a Variational Auto Encoder to incorporate shape-relevant features during training. We use a normalizing flow to map the pose and size of the shape's bounding box into a portion of the latent space of the VAE. During inference the network estimates a distribution over bounding boxes from which a specific box is sampled and used for reconstruction.

4th Conference on Robot Learning (CoRL 2020), Cambridge MA, USA.

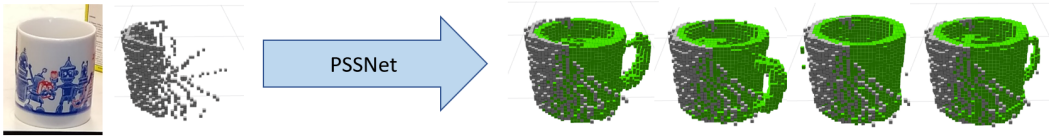

Figure 1: Our proposed PSSNet applied to a noisy segmented Kinect depth image of a mug produces multiple plausible reconstructions

Evaluating the quality of our network presents a dilemma: how do we decide if shapes produced by our network are plausible when they differ from the ground truth? We propose a non-learning method for generating plausible completions for a specific test dataset.

This paper makes the following contributions:

1. A method to generate plausible completions for an evaluation dataset
2. Metrics to evaluate plausible diversity of a black-box shape completer
3. PSSNet: A network for sampling diverse and plausible shape completions

To validate our method, we perform experiments using mugs from shapenet [5] and all YCB objects [6] which show that for ambiguous completions PSSNet produces diverse yet plausible samples, while baselines produce similar and poor quality completions. Without ambiguity PSSNet still retains similar performance to baselines. Finally, we construct physical robot scenarios of grasping objects in occlusion and clutter and show the diversity of PSSNet aids grasping. Code and a video are available at `https://github.com/UM-ARM-Lab/probabilistic_shape_completion` and `https://youtu.be/mY6c8jeZVKU`

2 Related Work

Shape Matching: Robotics has studied the problem of inferring 3D structure from RGB and depth camera images for decades. In the *shape matching* variant the pose or configuration of a target shape is estimated from observations. A classic but powerful non-learning approach uses the Iterative Closest Point (ICP) algorithm to align a target object with the observed pointcloud [7, 8], with some newer methods predicting a pose using neural networks [9, 10, 11]. Uncertainty can be modeled using discrete samples stored in a particle filter [12, 13, 14, 1, 15, 16, 17], where an observation model assigns a likelihood to each proposed shape based on the agreement with the observed depth image. Researchers have hand-crafted likelihood models using sum of squared pixel depth distances [1], outlier rejection [9], gaussian per-pixel error [18], and signed distance [19].

Shape matching requires known meshes for objects, limiting the applicability in an unstructured novel world. Our work uses shape matching to construct an evaluation dataset of plausible shapes and configurations for each given depth image. Using ICP followed by an outlier rejection observation model, we generate plausible particles to evaluate how well PSSNet captures uncertainty. PSSNet does not perform shape matching, nor require models outside of the training process.

Shape Completion: In *shape completion* or *shape reconstruction* the 3D structure is directly predicted from the camera observation. Shape datasets such as shapenet [5] and YCB [6] enable learning on sufficient examples to generate visually compelling results. The most common network architecture learns an encoder to a feature space followed by a decoder to the shape output [20, 21, 22, 3, 23, 24, 25, 26, 27, 28, 29, 30]. In different variants the encoder may accept voxelgrids [20, 23, 21, 29, 31], images [22, 26], or point clouds [32, 27]. Similarly the decoder may produce voxelgrids [20, 23, 21, 29, 31, 26], point clouds [32, 27], meshes [25], octrees [33], or implicit surfaces [34]. Our proposed network encodes to and from voxelgrids, however we expect out contributions to be applicable to other approaches.

In these networks a reconstruction loss such as voxel-independent binary crossentropy guides the optimizer [20, 21, 31, 30], which leads to averaging over possible shapes when there is ambiguity, producing "blurry" completions. Generative Adversarial Networks (GANs) [35] penalize this averaging and are used to produce natural-looking 3D reconstructions [30, 3, 29]. We might hope that by employing VAEs with GANs we could sample substantially different yet plausible completions

for a single input, yet this diversity has not been studied [30, 3, 29]. In our experience VAE-GANs have resulted in visually pleasing samples with low diversity.

Representing Bounding Box Uncertainty: Our proposal for encouraging diversity involves explicitly training the feature space of a VAE to represent means and variances in properties such as position, orientation, and size. The vector representation of these chosen features and their uncertainties must be representable and learnable by a neural network, which is a notorious challenge when representing rotations in SO(3). While new rotation belief representations [2] would be interesting to explore in our framework, we follow the approach of Tremblay et al. [36] and represent pose as a bounding box using 8 3-dimensional points. However, the standard independent Gaussian prior of a VAE is a poor prior for boxes where we expect corner locations to be highly coupled.

Normalizing flows have become popular in image generation as a method to invertably and losslessly map the tightly coupled distribution of pixel values onto an independent Gaussian distribution [37, 38, 39]. However, normalizing flows have also been proposed to model posterior distributions of VAEs [40, 41]. We take a similar, but inverted, approach and learn a normalizing flow as a map from the distribution of bounding boxes to the same independent Gaussian distribution used in our VAE.

3 Problem Formulation and Metrics

We assume a dataset of pairs (x, y) where x is the two voxelgrids (known occupied, known free) for voxelized shape y. In this work we refer to an *object* as a mesh at an unspecified pose and a *shape* as a voxelgrid produced by an object at a specific pose. We assume that for each x there is given a set of plausible completions $\mathcal{P}(x)$. We desire a non-deterministic function $\tilde{y}_i \sim f(x)$ where $\tilde{y}_i$ is a voxelgrid called a *completion* of x. Drawing n samples from $f(x)$ gives a set of completions $\tilde{Y}_x = \{\tilde{y}_1, ..., \tilde{y}_n\}$. Let $d(y_1, y_2)$ be a distance function between two voxelgrids (e.g. Chamfer Distance). We define the Best Accuracy as $M_A(x) = \min_{\tilde{y}_i \in \tilde{Y}_x} d(\tilde{y}_i, y)$. For a given (x, y) pair in our test dataset we additionally evaluate the quality of f using 3 criteria:

1. The coverage of plausible completions:

$$M_C(x) = \frac{1}{|\mathcal{P}(x)|} \sum_{\hat{y} \in \mathcal{P}(x)} \min_{\tilde{y}_i \in \tilde{Y}_x} d(\tilde{y}_i, \hat{y}) \tag{1}$$

2. The average plausibility of completions generated by f:

$$M_P(x) = \frac{1}{|\tilde{Y}_x|} \sum_{\tilde{y}_i \in \tilde{Y}_x} \min_{\hat{y} \in \mathcal{P}(x)} d(\tilde{y}_i, \hat{y}) \tag{2}$$

3. The Plausible Diversity:

$$M_{PD} = M_C + M_P \tag{3}$$

M_A is most similar to metrics used in previous work and is also not dependent on construction of $\mathcal{P}$. M_C penalizes plausible shapes that are not generated by f, whereas M_P penalizes network samples that are far from $\mathcal{P}$. We want to generate diverse samples that are plausible, thus we seek an f that achieves lowest M_{PD}, which is the chamfer distance between the sets $\mathcal{P}$ and $\tilde{Y}$.

4 Method

4.1 Plausible Shape Sampling

Our Plausible Shape Sampling Network, PSSNet, is an adaptation of a variational auto encoder (VAE). During inference PSSNet exactly follows a VAE, with an encoder that predicts a latent mean and variance from which a latent vector is sampled, and a decoder that produces a 3D voxelgrid from this latent vector. During training PSSNet differs from a VAE by replacing a portion of the latent space with a learned representation of an additional input.

Our training data starts with a set of mesh objects at a single pose. For each object we compute an axis-aligned bounding box. We then augment the dataset by applying rotations and translations to

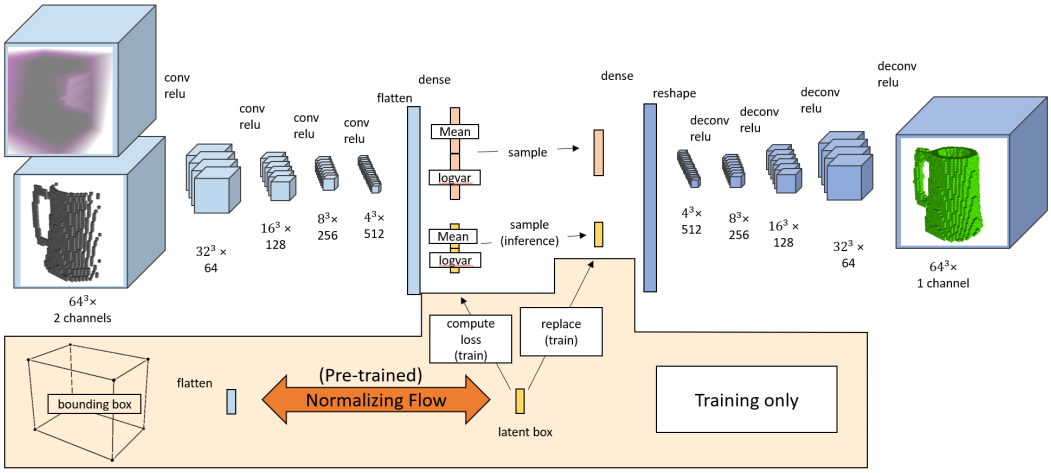

Figure 2: Our network, PSSNet, has the structure of a VAE during inference. During training we separate the latent space into typical learned VAE features and "latent box" feature produced by a learned normalizing flow applied to the ground truth bounding box. These latent box features are used both as a loss on the encoder prediction and as input to the decoder during training.

each object and bounding box. Finally, we compute the voxelized shape y and the known-free and known-occupied voxels x from a fixed view.

We train a normalizing flow on the bounding boxes of the training dataset with a Gaussian prior $\mathcal{N}(0, 1)$. Each bounding box consists of 8 points, thus this flow maps from a 24 dimensional "box" space to a 24 dimensional "latent box" space ψ. The flow consists of 8 RealNVP networks [38], each with 2 hidden layers of size 512. During training batch normalization is performed between every other RealNVP network.

We then use this flow in training PSSNet (Fig. 2). The encoder takes as input x the known-occupied and the known-free 64^3 voxelgrids. $2 \times 2 \times 2$ convolutions with a stride of 2 and relu activation are applied 4 times sequentially using [64, 128, 256, 512] channels. The output is densely connected to a 200D latent-mean and 200D latent log-variance. During inference the network is identical to a VAE, and thus a latent vector is sampled from this mean and log-variance. The decoder inverts the structure of the encoder, with a dense layer reshaped into a 4x4x4x512 tensor followed by "deconvolution", or convolution-transpose layers again with a stride of 2. The output of the decoder, $\tilde{y}$, is a 64^3 voxelgrid that represents the probability of occupancy for each voxel, independently. We threshold this voxelgrid at 0.5 to produce a binary occupancy.

During training, PSSNet differs from a VAE during the latent space sampling. The latent space is partitioned into two vectors: z^f and the 24 dimensional latent box space z^b. During training z^b is replaced by ψ, the latent box produced by the normalizing flow applied to the bounding box, thus z^b has no effect on the final voxelgrid produced. A loss term L^{flow} rewards the log-likelihood of ψ given the latent mean z^b_μ and variance z^b_{logvar} produced by the encoder. Additional loss terms for binary cross-entropy reconstruction loss L^{rec} and L^{VAE} form the Monte Carlo estimate of the Evidence Lower Bound (ELBO) [42] as applied to shape completion [29, 3]. With N as the total number of voxels (64^3), $y[i]$ as the target value $\{0, 1\}$ of the ith voxel, and $\varphi(\mu, \sigma_{logvar})$ is the probability density at μ of a Gaussian with log-variance σ_{logvar}.

$$L^{\text{rec}} = p(y|z) = \frac{1}{N}\sum_{i=1}^{N} -y[i]\log(\tilde{y}[i]) - (1-y[i])\log(\tilde{y}[i]) \tag{4}$$

$$L^{\text{VAE}} = \log(p(z^f)) - \log(p(z^f|x)) = \log(\varphi(z^f, 0)) - \log(\varphi(z^f - z^f_\mu, z^f_{\text{logvar}})) \tag{5}$$

$$L^{\text{flow}} = \log(p(\psi|z^b_\mu, z^b_{\text{logvar}})) = \log(\varphi(\psi - z^b_\mu, z^b_{\text{logvar}})) \tag{6}$$

4.2 Quantifying Plausibility

Many shape completion methods evaluate results using the metric $d(f(x), y)$, which may be appropriate if the ground truth shape is unambiguous given the view from the depth camera. However, given two different shapes y_1, y_2 in the dataset with similar corresponding depth camera image $x_1 \approx x_2$ it is unreasonable to expect f to always generate the correct output. Furthermore, for our application we desire f to output diverse yet plausible shapes.

We propose two criteria to define some y_j as a plausible completion of x_i:

- Observing x_i given y_j must be sufficiently likely given a camera observation model
- The object represented by y_j is in the test database, possibly with a different pose

To address the first criterion we define an observation model $obs(x, y)$ as the likelihood of observing the depth image of the 2.5D view $Im(x)$, given that the true occupied voxels are y. Similar work uses the sum-of-squared depth differences of $Im(x) - Im(y)$ [1], yet we find this model is not sufficiently discriminative. On the other hand, applying a Gaussian belief to each pixel independently [18] is far too discrimative, as a single pixel can alter the likelihood by orders of magnitude. We have had the most success with an outlier rejection model [9].

We define our $obs(x, y)$ as a binary likelihood in Algorithm 1, indicating if x is or is not plausible. We first compute a mask of unreliable depth pixels as any pixel in $Im(y)$ with gradient greater than some threshold δ, and inflate this mask by one pixel (Line 3). We accept x as a plausible depth image of y if every reliable pixel of $||Im(x) - Im(y)||$ is below $\delta = 4$cm. Depending on sensor noise it may be appropriate to allow some outliers. We deem certain pixels in the depth image $Im(y)$ "unreliable" if they are at the boundary of shape, as discretization approximations due to pixelization may assign a vastly different depth value due to a slight translation orthogonal to the camera. We see this effect on physical hardware such as a Kinect as depth values near the boundary of shapes are sometimes far too large, causing points to trail off into the background.

With obs now defined, we generate candidate shapes using objects from the test dataset D_{TEST}. Uniformly sampling poses and objects is infeasibly inefficient, as the vast majority of samples are not plausible. As in some particle filter approaches [43], we sample candidate states and project these onto a manifold of states more likely to be plausible. Algorithm 2 describes our approach. For each $(x_i, y_i) \in D_{TEST}$ we attempt to create a plausible completion using every element $(x_j, y_j) \in D_{TEST}$. We find a transformation T to align the 2.5D voxelgrids x_j to x_i using ICP [44] (Line 3). We then check if the observation is plausible given this aligned shape.

Algorithm 1 Observation Plausible: $obs(x, y)$

```
1: obs_image = Im(x)
2: exp_image = Im(y)
3: mask = ComputeUnreliable(expected_image)
4: for each pixel index i not in mask do
5:     if ||obs_image[i] - exp_image[i]|| > δ then
6:         return False
7: return True
```

Algorithm 2 Compute Plausibles(x_i)

```
1: P(x_i) = ∅
2: for (x_j, y_j) ∈ D_TEST do
3:     T = ICP(x_j, x_i)
4:     if obs(x_i | T y_j) then
5:         P(x_i) = P(x_i) ∪ T y_j
6: return P(x_i)
```

5 Experiments

We present quantitative and qualitative results demonstrating that for non-ambiguous completions PSSNet performs on par with existing methods, and that when there is ambiguity PSSNet performs better. We created datasets from shapenet [5] and YCB [6] such that 2.5D views could have multiple consistent completions. We trained PSSNet as described above as well as a VAE, a VAE with GAN loss similar to [30], and 3D-rec-GAN++ (without super-resolution layers) [29], with networks accepting and producing voxelgrids of size 64^3. We constructed plausible completions $\mathcal{P}$ for each x in our test dataset and evaluated our metrics (Section 3) using $d(y_1, y_2)$, as chamfer distance between voxelgrids converted to pointclouds, as it is a common metric of shape completion quality [3].

Shapenet Mugs: Using the *mugs* category from shapenet we constructed a dataset of 177 train and 37 test meshes. We rotated each mesh and associated bounded box in 5 degree increments about

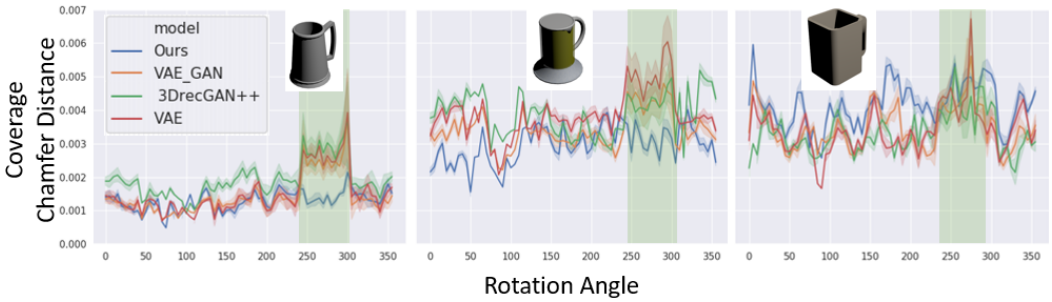

Figure 3: Coverage (Eq. 1) of various methods for shapenet mugs at different rotation angles. Rotations where the mug handle is occluded are highlighted.

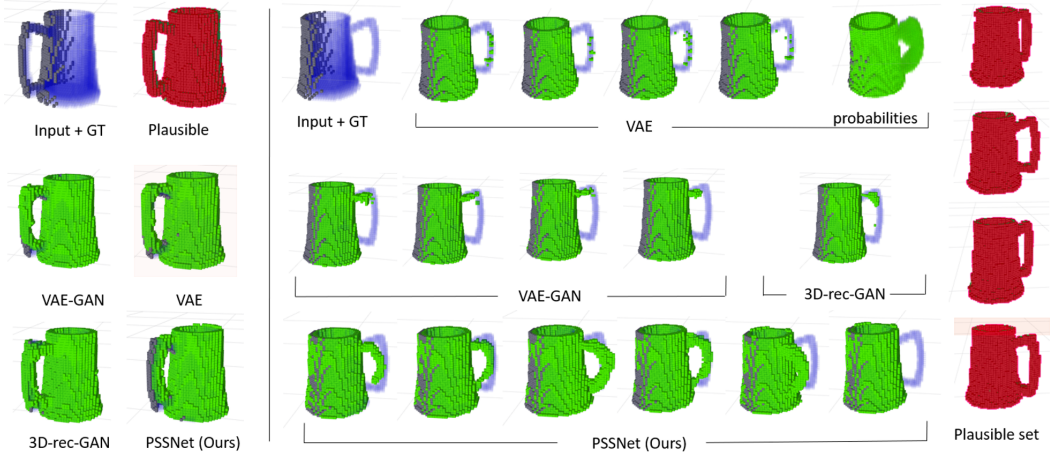

Figure 4: Completions (green) of a mug are sampled from the visible 2.5D view (grey). When the handle is visible (left) all methods produce similar mugs close to the ground truth (GT) (blue). When the handle is occluded (right) sampling from PSSNet yields mugs with different styles of handles in different orientations, with similar variation seen in the plausible set (4 shapes shown).

	Shapenet: all mugs				Shapenet: occluded handle			
	best acc	coverage of $\mathcal{P}$	avg. plaus	plausible diversity	best acc	coverage of $\mathcal{P}$	avg. plaus	plausible diversity
PSSNet (ours)	1.9	2.9	2.0	4.9	2.0	3.0	1.9	5.0
VAE	2.0	3.3	1.9	5.2	2.7	3.6	2.2	5.8
3D-rec-GAN	2.2	3.6	2.0	5.6	3.0	3.9	2.3	6.2
VAE-GAN	2.0	3.3	1.9	5.2	2.8	3.7	2.3	5.9
	YCB: 30 pixel wide slit				YCB: 6 pixel narrow slit			
PSSNet (ours)	1.3	1.7	3.2	4.8	2.3	4.5	4.4	8.9
VAE	1.5	3.1	1.8	4.9	3.0	7.8	2.8	10.6
3D-rec-GAN	1.2	3.6	1.2	4.8	4.6	9.6	2.9	12.4
VAE-GAN	1.3	3.3	1.6	5.0	3.1	7.9	2.7	10.6

Table 1: Best sample accuracy, Coverage of the plausible set, Average sample plausibility and Plausible diversity in mm. PSSNet performs best relatively in "Shapenet: occluded handle" and "YCB: narrow slit", as in these datasets there is ambiguity in the full shape given the partial view.

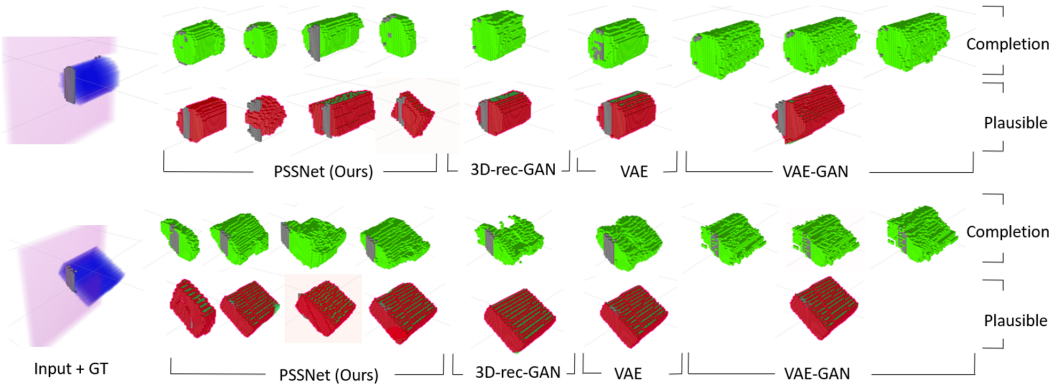

Figure 5: Completions of YCB objects as viewed through a 6 pixel narrow slit with the nearest plausible shape shown for each network sample. PSSNet generates diverse samples where other networks generate only small variations on the same sample.

the vertical axis and voxelized using binvox [45], creating 12744 train and 2664 test shapes. For approximately 1/5 of rotations, the handle is completely occluded from the 2.5D view.

We display the coverage metric for three of these shapes in Fig. 3. The left and middle mugs have a typical handle and when the handle is visible all methods obtain similar coverage. When the handle is occluded other methods perform far worse on M_C, meaning there are plausible completions that significantly differ from any samples produced by the network. PSSNet retains similar coverage even in these occluded regions. The right mug is square and unlike mugs in the training dataset, and the chamfer distance reconstruction error is dominated by the mug body reconstruction.

We visualize samples in Fig. 4 and qualitatively observe the same trends. When visible, all methods accurately reconstruct the mug handle, but when occluded other methods tend to average over plausible mugs and produce poor and non-diverse samples. For the 7 mugs from PSSNet the handles vary in orientation and style while remaining in the occluded region. We find PSSNet generates these diverse plausible handles for many but not all mugs. Qualitatively, we observe similar behavior for PSSNet with live Kinect depth images using a hard-coded segmentation of a mug (Fig. 1).

YCB with slit occlusion: We constructed a training dataset by applying a total of 24 rotations about the vertical and a horizontal axis for each YCB object. During training we occlude left and right portions of the depth image to simulate viewing the object through a vertical slit. We randomly translate the YCB shape and then randomly select a slit of width 5 to 30 pixels (1 pixel $\approx$ 0.6cm) and randomly place this slit so that the target object is visible in at least 5 columns of the image. A full 2.5D view of any YCB object leaves little ambiguity, and this slit simulates viewing occluded objects in a cluttered scene.

We construct two test datasets for a subset of the YCB objects by using the same set of rotations but fix the translations and fix slit widths to 6 and 30 pixels. For each fixed slit width we construct a separate $\mathcal{P}$ by fitting (Alg. 2) each test shape at each orientation and each translation along the slit in 2 pixel increments. 6 pixels is a small portion of each object, thus in this dataset different objects with many different translations tend to match each x. The 30 pixel slit captures most of the object, so there is little ambiguity as to the 3D shape. We visualize completions in Fig. 5.

Metrics averaged over all test datasets are shown in Table 1. PSSNet consistently provides the best coverage and comparably in plausible diversity for the datasets with lower ambiguity and outperforms baselines for datasets with greater ambiguity.

Physical Robot: We constructed two scenarios on a physical robot, shown in Fig. 6 and the accompanying video, where a grasp is chosen based on completions of a mug and the YCB Cheez-it box. Consistent with the simulation experiments we found the baseline methods did not produce reasonable handles for the mug and thus grasps failed, while PSSNet produced multiple plausible handles leading to a successful grasp. Similarly, for the Cheez-it box viewed throw a narrow slit formed from other clutter, baseline methods produced nearly no variation about an incorrect completion, thus the attempted top grasp was not successful. PSSNet produced a variety of completions, some

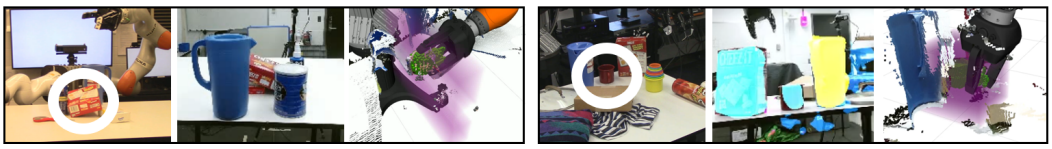

Figure 6: Robot scenarios for grasping Cheez-it box (left) and the mug (right). From left to right: The scene, the robot's view of the scene, and a grasp attempt.

similar to the Cheez-it box and some more similar to other YCB objects. With this ambiguity, the robot executed a side grasp that would capture many of the different possibilities, and successfully grasped the Cheez-it box. Details are further described in the appendix A.2.

6 Discussion

We achieve our goal of creating a network that generates diverse samples, while other networks generate only small variations on a single completion. PSSNet, however, performs worse on M_P, indicating that either PSSNet sometimes produces poor quality samples, or that $\mathcal{P}$ lacks some plausible completions. Subjectively, we see both cases. Given a larger set of test shapes, $\mathcal{P}$ would contain more shapes, and likely M_P would improve. Below we discuss what we see as the main advantages and limitations of our design choices for PSSNet and the plausible set:

Feature replacement: The partial feature replacement in the latent space of the VAE allows proper credit assignment between the encoder and decoder during training of ambiguous samples. For inputs where the reconstruction is inherently ambiguous we desire the encoder to predict variance in the latent space. However given this ambiguity in latent space the reconstruction loss is minimized when the decoder averages over plausible shapes. Replacing these latent box features during training removes some ambiguity so that the reconstruction loss is minimized when the decoder produces a specific object without as much blur.

Normalizing flow: The normalizing flow transforms the box features into the distribution $\mathcal{N}(0, 1)$ of the VAE prior, providing two important properties. First, this maps the arbitrary range of the box features into the correct range for sampling from the VAE without requiring a distance function in latent box space. Second, because the normalizing flow tends to be locally smooth, uncertainty in latent-box space corresponds to rotation, translation, and resizing uncertainty of the bounding box, allowing the VAE prior to model the variance of our dataset.

Computing the Plausible Set: ICP finds local, not global, minima and typically ICP is run many times with different initializations. Our dataset D_{TEST} contains many copies of each object at different rotations, and these copies serve the function of different initializations. However, there are some limitations of our plausible set computation. Our algorithm to compute $\mathcal{P}$ has quadratic complexity which limits the size of the test dataset. In addition, our observation model explicitly ignores small depth errors without considering correlation of errors between pixels, yet small but correlated depth differences could be used to identify larger shapes. Similarly, we explicitly discard depth values on the borders of shapes as independently these pixels tend to be noisy, yet again correlated depth values may provide useful information that is observable even with the independent noise. Our network f may use such features, but $\mathcal{P}$ will not, thus our evaluation may be overly harsh on our network, penalizing it for not generating shapes in $\mathcal{P}$ even when they are not plausible.

7 Conclusion

In this work, we proposed PSSNet, a method for generating diverse yet plausible 3D completions of a 2.5D depth image. A normalizing flow transforms the side information of the true shape bounding box into a feature space, which is used during training to encourage an encoder to generate diverse latent space samples, and to aid the decoder in producing plausible samples. To evaluate this method on a specific dataset we proposed a shape matching method to generate a set of plausible completions, as well as metrics for plausible diversity. In experiment PSSNet generated diverse samples and outperformed existing approaches for depth images with ambiguous reconstructions.

Acknowledgments

This work was supported in part by NSF grant IIS-1750489 and by Toyota Research Institute (TRI). This article solely reflects the opinions of its authors and not TRI or any other Toyota entity.

References

[1] K. Desingh, O. C. Jenkins, L. Reveret, and Z. Sui. Physically plausible scene estimation for manipulation in clutter. *Humanoids*, 2016.

[2] V. Peretroukhin, M. Giamou, D. Rosen, and W. N. Greene. A smooth representation of belief of SO(3) for deep rotation learning with uncertainty. *RSS*, 2020.

[3] J. Wu, C. Zhang, X. Zhang, Z. Zhang, W. T. Freeman, and J. B. Tenenbaum. Learning shape priors for single-view 3D completion and reconstruction. In *ECCV*, 2018.

[4] B. Saund, S. Choudhury, S. Srinivasa, and D. Berenson. The blindfolded robot : A bayesian approach to planning with contact feedback. *ISRR*, 2019.

[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, 2015.

[6] B. Calli, A. Singh, J. Bruce, A. Walsman, K. Konolige, S. Srinivasa, P. Abbeel, and A. M. Dollar. Yale-CMU-Berkeley dataset for robotic manipulation research. *IJRR*, April 2017.

[7] P. J. Besl and N. D. McKay. A method for registration of 3-D shapes. *TPAMI*, 1992.

[8] H. Yang, J. Shi, and L. Carlone. TEASER: Fast and Certifiable Point Cloud Registration. *arXiv*, 2020.

[9] V. Narayanan and M. Likhachev. Discriminatively-guided deliberative perception for pose estimation of multiple 3D object instances. In *RSS*, June 2016.

[10] X. Deng, A. Mousavian, Y. Xiang, F. Xia, T. Bretl, and D. Fox. PoseRBPF: A rao-blackwellized particle filter for 6D object pose estimation. In *RSS*, 2019.

[11] T. Hodan, D. Barath, and J. Matas. EPOS: Estimating 6d pose of objects with symmetries. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020.

[12] K. Hausman, S. Niekum, S. Osentoski, and G. S. Sukhatme. Active articulation model estimation through interactive perception. In *ICRA*, 2015.

[13] M. C. Koval, N. S. Pollard, and S. S. Srinivasa. Pose estimation for planar contact manipulation with manifold particle filters. *IJRR*, 2015.

[14] K. Desingh, S. Lu, A. Opipari, and O. C. Jenkins. Efficient nonparametric belief propagation for pose estimation and manipulation of articulated objects. *Science Robotics*, 2019.

[15] K. Desingh, S. Lu, A. Opipari, and O. C. Jenkins. Factored pose estimation of articulated objects using efficient nonparametric belief propagation. In *ICRA*, 2019.

[16] Z. Liu, D. Chen, K. M. Wurm, and G. von Wichert. Table-top scene analysis using knowledge-supervised mcmc. *Robotics and Computer-Integrated Manufacturing*, 2015.

[17] S. Chen, B. Saund, and R. Simmons. The datum particle filter: Localization for objects with coupled geometric datums. In *IROS*, 2017.

[18] M. Wüthrich, P. Pastor, M. Kalakrishnan, J. Bohg, and S. Schaal. Probabilistic object tracking using a range camera. In *IROS*, 2013.

[19] T. Schmidt, R. A. Newcombe, and D. Fox. Dart: Dense articulated real-time tracking. In *Robotics: Science and Systems*, 2014.

[20] Zhirong Wu, S. Song, A. Khosla, Fisher Yu, Linguang Zhang, Xiaoou Tang, and J. Xiao. 3D shapenets: A deep representation for volumetric shapes. In *CVPR*, 2015.

[21] 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.

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

[23] J. Wu, Y. Wang, T. Xue, X. Sun, W. T. Freeman, and J. B. Tenenbaum. MarrNet: 3D Shape Reconstruction via 2.5D Sketches. In *Advances In Neural Information Processing Systems*, 2017.

[24] M. Michalkiewicz, E. Belilovsky, M. Baktashmotlagh, and A. Eriksson. A simple and scalable shape representation for 3D reconstruction. *arXiv*, 2020.

[25] C. Wen, Y. Zhang, Z. Li, and Y. Fu. Pixel2mesh++: Multi-view 3D mesh generation via deformation. *ICCV*, 2019.

[26] H. Xie, H. Yao, X. Sun, S. Zhou, and S. Zhang. Pix2vox: Context-aware 3d reconstruction from single and multi-view images. In *ICCV*, 2019.

[27] H. Fan, H. Su, and L. J. Guibas. A point set generation network for 3D object reconstruction from a single image. *CVPR*, 2017.

[28] Y. Yu, Z. Huang, F. Li, H. Zhang, and X. Le. Point encoder gan: A deep learning model for 3D point cloud inpainting. *Neurocomputing*, 2020.

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

[30] J. Wu, C. Zhang, T. Xue, B. Freeman, and J. Tenenbaum. Learning a probabilistic latent space of object shapes via 3D generative-adversarial modeling. In *NIPS*, 2016.

[31] A. Dai, C. R. Qi, and M. Nießner. Shape completion using 3D-encoder-predictor CNNs and shape synthesis. *CVPR*, 2017.

[32] W. Yuan, T. Khot, D. Held, C. Mertz, and M. Hebert. PCN: Point completion network. In *3DV*, 2018.

[33] G. Riegler, A. O. Ulusoy, and A. Geiger. Octnet: Learning deep 3D representations at high resolutions. In *CVPR*, 2017.

[34] J. J. Park, P. Florence, J. Straub, R. A. Newcombe, and S. Lovegrove. DeepSDF: Learning continuous signed distance functions for shape representation. *CVPR*, 2019.

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

[36] J. Tremblay, T. To, B. Sundaralingam, Y. Xiang, D. Fox, and S. Birchfield. Deep object pose estimation for semantic robotic grasping of household objects. *CoRL*, 2018.

[37] L. Dinh, D. Krueger, and Y. Bengio. Nice: Non-linear independent components estimation. *CoRR*, 2014.

[38] L. Dinh, J. Sohl-Dickstein, and S. Bengio. Density estimation using Real NVP. *ArXiv*, 2017.

[39] D. P. Kingma and P. Dhariwal. Glow: Generative flow with invertible 1x1 convolutions. *NeurIPS*, 2018.

[40] D. J. Rezende and S. Mohamed. Variational inference with normalizing flows. *arXiv*, 2015.

[41] A. Vahdat and J. Kautz. NVAE: A deep hierarchical variational autoencoder. In *arxiv*, 2020.

[42] D. P. Kingma and M. Welling. Auto-encoding variational bayes. In Y. Bengio and Y. LeCun, editors, *ICLR*, 2014.

[43] M. Klingensmith, M. Koval, S. Srinivasa, N. Pollard, and M. Kaess. The manifold particle filter for state estimation on high-dimensional implicit manifolds, April 2016.

[44] R. B. Rusu and S. Cousins. 3D is here: Point Cloud Library (PCL). In *ICRA*, 2011.

[45] P. Min. binvox. `http://www.patrickmin.com/binvox`, 2004 - 2020.

APPENDIX

A Further Experiment details

A.1 Generalizing of shape

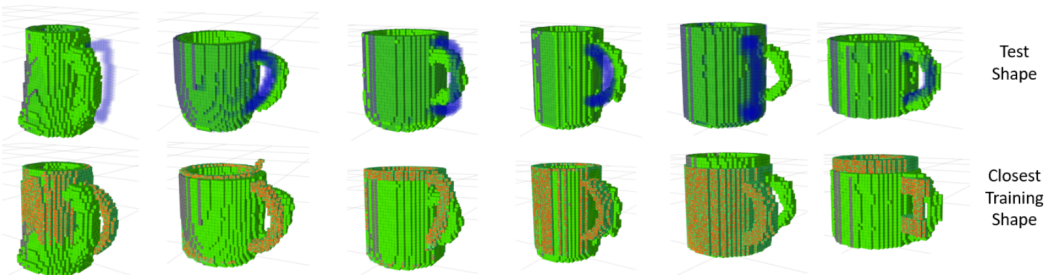

Figure 7: Similarity to the Training shapes. Completions (Green) from a depth view (grey) are shown in both the top and bottom row. In all cases the handle is occluded from the depth view. The top row additionally shows ground truth (transparent blue), while the bottom row additionally shows the training shape (orange) that is closest (chamfer distance) to the ground truth test shape. The completed handle is sometimes closer to the training shape (e.g. left-most), sometimes closer to the test shape (e.g. right-most) and sometimes different from both.

In addition to our main contribution, we ask if these shape completion networks are "completing the shape" or "looking up the closest object from the training set". To evaluate this we examine the quality of the completions of the test shape as compared to the *training* shapes. For each test shape from the Shapenet Mugs dataset we compute the closest (chamfer distance) training shape. We then sample 10 completions from PSSNet using the 2.5D view of the test shape and compute chamfer distance to both the closest-training and test shapes. Over all 26640 samples, the average chamfer distance to the test and closest-train shapes are 2.4mm and 3.8mm respectively. We find that in 2599 (approx 10%) of samples the completion is closer to the training shape. Numerically this indicates PSSNet (and presumably other shape completion networks) are more than searching for the nearest shape.

Qualitatively we notice features, such as the mug handle, sometimes visually appear closer to the closest-training shape. We visualize selected instances in Figure 7. We note that visually these completions represent the diversity we desire, where the completion of an occluded handle can vary.

A.2 Physical Robot Details

We constructed two grasping scenarios on a physical robot, shown in Fig. 8 and the accompanying video. The points from a Kinect were filtered using an image segmenting network to construct known occupied and known free voxelgrids for the target object. 20 completions were sampled from which grasp poses were calculated, and then a grasp was attempted. Our grasping strategies described below are simple but still serve to demonstrate the value of a diverse belief over shapes under ambiguity.

In the mug scenario, kinematics limits and clutter forced the robot to grasp the mug from the occluded handle on the far side from the robot. For each completion a grasp was chosen with a handcoded orientation and grasp point as the furthest back possible grasp to avoid collision with other clutter. The grasp attempted was the average of all valid grasp points with gripper width wide enough to capture all grasp points. VAE-GAN sampled completions that did not have visible handles, resulting in most grasp poses in collision with other clutter. Occasionally stray voxels appeared in VAE-GAN completions that generated valid grasp poses, but when attempted these grasps were not successful. Using PSSNet sampled completions with handles generated valid grasps, which when executed resulted in successful grasps of the true mug.

In the Cheez-it scenario, clutter occluded all but a narrow slit from which only a small portion of the box was visible. Potential grasps were sampled from both a top and side orientation with grasp

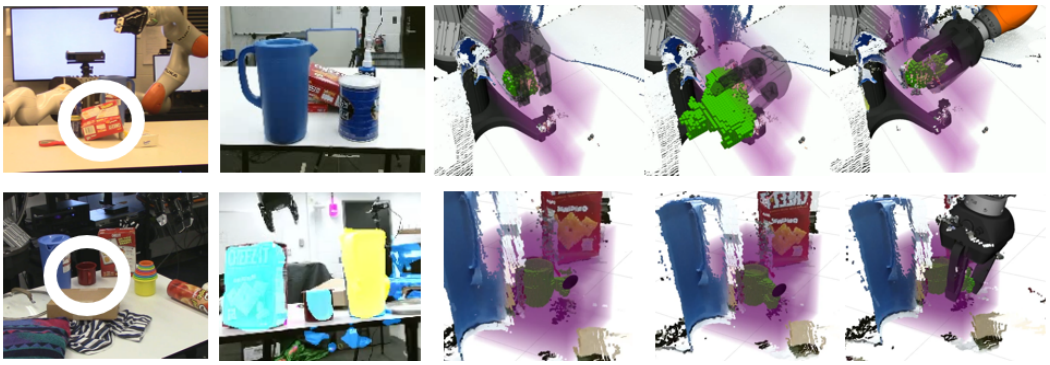

Figure 8: Robot scenarios for grasping Cheez-it box (top) and the mug (bottom). From left to right: The scene, the robot's view of the scene, 2 sampled completions using PSSNet, and a grasp attempt.

point at the centroid of the completed object. Completions from VAE-GAN were consistent, but the completed box was too shallow such that it appeared a top grasp would always be successful. These attempted top grasps were unsuccessful because the gripper collided with the larger-than-expected box. PSSNet again showed diversity with some completions thin and narrow and some as deep as the true box, so that it was unclear if a top grasp would be successful and thus the robot attempted and succeeded at side grasps.