



HAL
open science

Probabilistic Reconstruction Networks for 3D Shape Inference from a Single Image

Roman Klokov, Jakob Verbeek, Edmond Boyer

► **To cite this version:**

Roman Klokov, Jakob Verbeek, Edmond Boyer. Probabilistic Reconstruction Networks for 3D Shape Inference from a Single Image. BMVC 2019 - British Machine Vision Conference, Sep 2019, Cardiff, United Kingdom. pp.1-13. hal-02268466

HAL Id: hal-02268466

<https://inria.hal.science/hal-02268466>

Submitted on 20 Aug 2019

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Probabilistic Reconstruction Networks for 3D Shape Inference from a Single Image

Roman Klokov
roman.klokov@inria.fr

Jakob Verbeek
jakob.verbeek@inria.fr

Edmond Boyer
edmond.boyer@inria.fr

Univ. Grenoble Alpes, Inria,
CNRS, Grenoble INP*, LJK
38000 Grenoble, France
*Institute of Engineering

Abstract

We study end-to-end learning strategies for 3D shape inference from images, in particular from a single image. Several approaches in this direction have been investigated that explore different shape representations and suitable learning architectures. We focus instead on the underlying probabilistic mechanisms involved and contribute a more principled probabilistic inference-based reconstruction framework, which we coin Probabilistic Reconstruction Networks. This framework expresses image conditioned 3D shape inference through a family of latent variable models, and naturally decouples the choice of shape representations from the inference itself. Moreover, it suggests different options for the image conditioning and allows training in two regimes, using either Monte Carlo or variational approximation of the marginal likelihood. Using our Probabilistic Reconstruction Networks we obtain single image 3D reconstruction results that set a new state of the art on the ShapeNet dataset in terms of the intersection over union and earth mover’s distance evaluation metrics. Interestingly, we obtain these results using a basic voxel grid representation, improving over recent work based on finer point cloud or mesh based representations.

1 Introduction

The overwhelming success of convolutional neural networks on image data [16, 17] instigated the exploration of CNNs for other problems, in particular in 3D visual computing. 3D CNNs for shapes represented with uniform voxel grids have been investigated for recognition [21, 24] and generative modelling tasks [4, 41]. For 3D shape inference, initial works [9, 5] successfully demonstrated the ability of 3D CNNs to produce coherent voxelized shapes given single images. This task has since gained a significant attention, as a result of its vast application field and despite its challenging ill-posed nature.

Further exploring CNNs in this context, recent works have investigated beyond straightforward adaptations of 2D CNNs to 3D voxel grids, notably to overcome the cubic complexity in time and memory associated with it. For instance, sparse representations of large voxel grid have been proposed to reduce complexity while allowing for finer shape details [7, 37]. Other more scalable shape representations suitable for recognition and generation tasks have been investigated, including rendered images [33], geometry images [30], 2D depth maps [32], point clouds [15, 24, 25], and graphs [22, 38]. Importantly, these representations

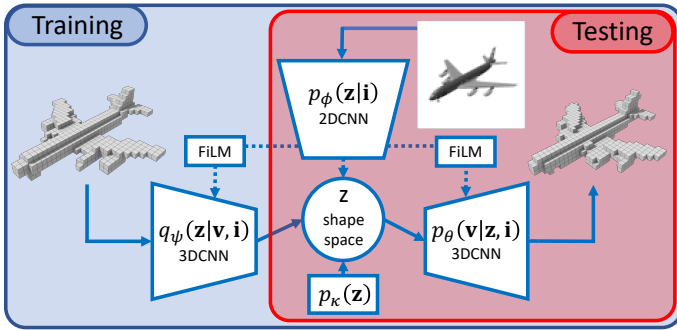


Figure 1: Probabilistic Reconstruction Networks for 3D shape inference from a single image. Arrows show the computational flow through the model, dotted arrows show optional image conditioning. Conditioning between 2D and 3D tensors is achieved by means of FiLM [23] layers. The inference network q_ψ is only used during training for variational inference.

come with specific network architectures and loss functions suited to the corresponding data structures.

The variety of approaches complicates comparisons and identification of the sources of improved performance. In particular, most works do not decouple the problems inherent to the task, and mix new shape representations, along with the associated network architectures and loss functions, with different image conditioning schemes, different probabilistic formulations of the shape prediction task, and in some cases the use of additional training data. This leads to possibly difficult or even unfair comparisons, due to the inability to confidently determine the source of improvements in new models, and emphasizes the need for a more systematic approach.

In this paper we look at the single image 3D shape inference through the prism of a family of generic probabilistic latent variable models, which we term *Probabilistic Reconstruction Networks* (PRN), see Figure 1. The formalism encompassing these models naturally decouples different aspects of the problem including the shape representation, the image conditioning, and the usage of latent shape space for the shape prediction. It also allows to categorize previous models for this task by their structural properties. Without loss of generality, we use voxel grids as shape representations and focus our attention on other aspects. We systematically analyze the impact of several design choices: (i) the dependency structure between the input image, the latent shape variable and the output variables; (ii) the effectiveness of training the model using Monte Carlo sampling or variational inference to approximate the log-likelihood; (iii) the effectiveness of a deterministic version of the model that suppresses any uncertainty associated with the latent variable; (iv) the effect of jointly learning the shape reconstruction model along with a generative shape model, which share their latent shape space.

For our experiments we use the ShapeNet dataset for the single image 3D shape reconstruction. We obtain excellent single image 3D reconstruction results with our Probabilistic Reconstruction Networks, setting new state-of-the-art results in terms of the IoU and EMD performance metrics. Interestingly, our results improve over recent works based on point-cloud and mesh-based shape representations.

In summary, our contributions are:

- a generic latent variable model for the single image 3D shape reconstruction;
- exploration of modeling options in a systematic and comparable manner;

- new state-of-the-art single image reconstruction results on the ShapeNet dataset.

In the following we first introduce our generic latent variable model in Section 2, which is then used to review and categorize previous work on shape inference from a single image in Section 3. We then present our experimental results and comparisons in Section 4.

2 Probabilistic framework for 3D shape reconstruction

Below we present our generic latent variable model in Section 2.1, and detail the network architectures used for our experiments in Section 2.2. Although proposed probabilistic model is agnostic to the underlying shape representation, we present it using voxel grid representation in accordance to our experimental setup.

2.1 Latent variable model for single image 3D shape reconstruction

We consider a shape \mathbf{v} as a uniform voxel occupancy grid of a predefined resolution. Our task is to predict the shape \mathbf{v} given an input image \mathbf{i} , *i.e.* to model $p(\mathbf{v}|\mathbf{i})$. While images and occupancy grids live in different spaces, both are representations of an underlying object that has a 3D shape and an appearance. Using this observation, we assume that an observed shape \mathbf{v} has a latent parametrization \mathbf{z} within a latent shape space of lower dimension, that captures shape variations. We then define our latent variable model for single image 3D shape reconstruction as:

$$p(\mathbf{v}|\mathbf{i}) = \int_{\mathbf{z}} p_{\theta}(\mathbf{v}|\mathbf{z}, \mathbf{i}) p_{\phi}(\mathbf{z}|\mathbf{i}) d\mathbf{z}, \quad (1)$$

where ϕ and θ are parameters of the model. This model consists of two modules: an image informed latent variable prior $p_{\phi}(\mathbf{z}|\mathbf{i})$, and a decoder $p_{\theta}(\mathbf{v}|\mathbf{z}, \mathbf{i})$ that predicts the shape \mathbf{v} given the image and the latent variables. Being a generic latent variable model, it allows us to decouple and study different aspects of the single image 3D shape reconstruction task.

Image conditioning options. In Eq. (1) both latent variable prior and decoder are conditioned on the input image. If we drop the dependence on \mathbf{i} from one module, we maintain dependence of the shape \mathbf{v} on the image \mathbf{i} , and obtain two alternative models:

$$p(\mathbf{v}|\mathbf{i}) = \int_{\mathbf{z}} p_{\theta}(\mathbf{v}|\mathbf{z}) p_{\phi}(\mathbf{z}|\mathbf{i}) d\mathbf{z}, \quad (2)$$

$$p(\mathbf{v}|\mathbf{i}) = \int_{\mathbf{z}} p_{\theta}(\mathbf{v}|\mathbf{z}, \mathbf{i}) p_{\kappa}(\mathbf{z}) d\mathbf{z}. \quad (3)$$

In the first case, we omit the image conditioning in the decoder $p_{\theta}(\mathbf{v}|\mathbf{z})$ and assume that the image conditioned prior $p_{\phi}(\mathbf{z}|\mathbf{i})$ is sufficient to obtain a valid reconstruction by the decoder. This dependency structure corresponds to an assumption of conditional independence of \mathbf{i} and \mathbf{v} given \mathbf{z} . In the second case, we leave the image dependence in the decoder $p_{\theta}(\mathbf{v}|\mathbf{z}, \mathbf{i})$ but use an unconditional prior for the latent variable $p_{\kappa}(\mathbf{z})$. This corresponds to an assumption that \mathbf{i} and \mathbf{z} are a-priori independent. If we drop the image conditioning in both components, the model becomes a generative latent variable shape model: $p(\mathbf{v}) = \int p_{\theta}(\mathbf{v}|\mathbf{z}) p_{\kappa}(\mathbf{z}) d\mathbf{z}$.

Latent variable sampling during training. Due to the non-linear dependencies in the integral in the models defined in equations (1)–(3), exact computation of the log-likelihood

and its gradient is intractable. We consider two alternative approaches to overcome this difficulty. The first is to use a Monte Carlo approximation, as *e.g.* in [8],

$$\ln p(\mathbf{v}|\mathbf{i}) \approx \ln \frac{1}{M} \sum_{m=1}^M p_{\theta}(\mathbf{v}|\mathbf{z}_m), \quad \mathbf{z}_m \sim p_{\phi}(\mathbf{z}|\mathbf{i}), \quad (4)$$

where we make use of the re-parametrization trick [14, 17] to back-propagate the gradient of the log-likelihood *w.r.t.* ϕ through the sampling from $p_{\phi}(\mathbf{z}|\mathbf{i})$.

Alternatively, we can use the variational inference framework [14, 17] to obtain a training signal based on more informed samples from the latent variable. We introduce a variational approximate posterior $q_{\psi}(\mathbf{z}|\mathbf{v}, \mathbf{i})$, which is learned jointly with the prior and decoder by the maximization of the variational lower bound on the log-likelihood:

$$\mathcal{L}(\phi, \psi, \theta, \mathbf{v}, \mathbf{i}) = \mathbb{E}_{q_{\psi}(\mathbf{z}|\mathbf{v}, \mathbf{i})}[\log p_{\theta}(\mathbf{v}|\mathbf{z}, \mathbf{i})] - \mathcal{D}_{\text{KL}}(q_{\psi}(\mathbf{z}|\mathbf{v}, \mathbf{i}) || p_{\phi}(\mathbf{z}|\mathbf{i})) \leq \ln p(\mathbf{v}|\mathbf{i}). \quad (5)$$

To evaluate this bound and its gradient, we sample from $q_{\psi}(\mathbf{z}|\mathbf{v}, \mathbf{i})$. Since the samples are conditioned on the shape, unlike the Monte Carlo approximation case, we expect this approach to be more sample efficient. On the one hand, image conditioning in the posterior may be omitted, since the posterior is already conditioned on the shape information. On the other hand, conditioning on the image may in principle further improve the accuracy of the approximate posterior, as it is based on more information.

Deterministic shape model. We can obtain deterministic versions of the presented models by considering \mathbf{z} as a function of ϕ and, if required, \mathbf{i} . Although this simplifies the models, it also discards an important property. Typically, $p(\mathbf{v}|\mathbf{z})$ is factorized, with each voxel occupancy being modelled as an independent Bernoulli, *e.g.* [9, 5]. With this factorization, any shape ambiguity given a single image cannot be modelled properly, since self-occluded parts of the shape lack structure in the prediction. A latent variable that conditions the factorized distribution can be used to induce dependencies among the voxel occupancies to reflect the structured ambiguity resulting from partially observed shapes.

In case of the variational training we also use a deterministic posterior, and substitute the KL-divergence in Eq. (5) with a suitable similarity measure. For example, the L_2 -norm of the difference between the output of image encoder p_{ϕ} and the posterior q_{ψ} .

Merging unconditional generation with reconstruction. The models presented above can be trained along with a generative shape model, that may be of interest on its own, or used to regularize the conditional model. To achieve this, we consider a variational bound on the log-likelihood of an unconditional generative model:

$$\mathcal{L}(\kappa, \psi, \theta, \mathbf{v}) = \mathbb{E}_{q_{\psi}(\mathbf{z}|\mathbf{v})}[\log p_{\theta}(\mathbf{v}|\mathbf{z})] - \mathcal{D}_{\text{KL}}(q_{\psi}(\mathbf{z}|\mathbf{v}) || p_{\kappa}(\mathbf{z})) \leq \ln p(\mathbf{v}), \quad (6)$$

where $p(\mathbf{v}) = \int p_{\theta}(\mathbf{v}|\mathbf{z})p_{\kappa}(\mathbf{z})d\mathbf{z}$. Looking at Eq. (5) and Eq. (6), we observe that if we omit image conditioning from the decoder and the posterior in Eq. (5), and share their parameters in both conditional and unconditional models, we can obtain a unified training objective:

$$\begin{aligned} \mathcal{L}(\kappa, \phi, \psi, \theta, \mathbf{v}, \mathbf{i}) = & \mathbb{E}_{q_{\psi}(\mathbf{z}|\mathbf{v})}[\log p_{\theta}(\mathbf{v}|\mathbf{z})] - \frac{1}{2}\mathcal{D}_{\text{KL}}(q_{\psi}(\mathbf{z}|\mathbf{v}) || p_{\kappa}(\mathbf{z})) \\ & - \frac{1}{2}\mathcal{D}_{\text{KL}}(q_{\psi}(\mathbf{z}|\mathbf{v}) || p_{\phi}(\mathbf{z}|\mathbf{i})), \end{aligned} \quad (7)$$

which is the average of the lower bounds on the marginal likelihood on the shape \mathbf{v} with and without conditioning on the image. The term corresponding to the unconditional likelihood

can be viewed as a regularization that encourages \mathbf{z} from the entire latent space to correspond to realistic shapes, instead of just the \mathbf{z} in the support of the conditional distributions $p(\mathbf{z}|\mathbf{i})$ on latent coordinates given an input image. Note that in the generative model neither the decoder $p_\theta(\mathbf{v}|\mathbf{z})$ nor the encoder $q_\psi(\mathbf{z}|\mathbf{v})$ are conditioned on the image. These can thus be shared with the single image reconstruction model if the latter is not conditioned on the image in these components.

2.2 Network architectures

Although our probabilistic model is not representation specific, we focus on the voxel grid shape representation, and leave the comparison to alternative representations for the future work. Thus, we implement the different conditional distributions in our model as 2D and 3D CNNs that output the parameters of distributions on the latent variable \mathbf{z} , or on the voxel occupancies. In particular,

- The unconditional Gaussian latent prior $p_\kappa(\mathbf{z})$ is characterized by κ that consists of means and variances for all latent dimensions implemented as trainable parameters.
- The image conditioned prior $p_\phi(\mathbf{z}|\mathbf{i})$ is a 2D CNN, consisting of six blocks of pairs of convolutions: a standard and a strided one, interleaved with batch normalization and point-wise non-linearities, followed by two fully-connected layers. It processes input images into the means and variances of a factored Gaussian on \mathbf{z} .
- The shape conditioned variational posterior $q_\psi(\mathbf{z}|\mathbf{v}, \mathbf{i})$ is a 3D CNN consisting of an initial 3D convolution and a series of four modified residual blocks [2, 3], each using an additional 1×1 convolution instead of identity and feature map concatenation instead of summation and each followed by $2 \times 2 \times 2$ spatial average pooling. Final convolutional features are fed to two additional fully-connected layers. This encoder processes input shapes into the means and variances of a factored Gaussian on \mathbf{z} .
- The 3D deconvolutional decoder $p_\theta(\mathbf{v}|\mathbf{z}, \mathbf{i})$ is mirrored from the approximate posterior q_ψ , with the pooling being substituted by the $2 \times 2 \times 2$ upscaling by trilinear interpolations, producing the parameters of Bernoulli distributions on the voxel occupancies from a latent variable input.

Image conditioning in the two latter modules is inspired by the FiLM conditioning mechanism [23]. Intermediate 2D feature maps from the first five convolutional blocks of the image encoder $p_\phi(\mathbf{z}|\mathbf{i})$ are averaged spatially, transformed by two additional fully-connected layers into weights and biases, that are used to scale the according five intermediate batch-normalized 3D feature maps in the shape encoder, the latent variable decoder, or both. Instead of affine transformation used in FiLM, we use non-negative scaling weights by predicting them in logarithmic scale, in our experiments this resulted in more stable training and slightly better results. See Figure 1 for a schematic overview of the model architecture.

To ensure fair comparison between the variations of the model we use identical architectures for every component concurring in different models, except for additional fully-connected layers associated with the different image conditioning options. When we include an unconditional generative shape model and optimize Eq. (7), we share the decoder $p_\theta(\mathbf{v}|\mathbf{z})$ and the variational posterior $q_\psi(\mathbf{z}|\mathbf{v})$ between the conditional and unconditional models. Exact architectures, training procedures and their hyperparameters are available on the implementation code page.¹

¹<https://github.com/Regenerator/prns>

Dependencies	Sampling	Deterministic	Discriminator	References
$p(\mathbf{s} \mathbf{z})p(\mathbf{z} \mathbf{i})$	$p(\mathbf{z} \mathbf{i})$	✓		[1, 13, 28, 29, 36, 37, 40, 41, 45]
$p(\mathbf{s} \mathbf{z})p(\mathbf{z} \mathbf{i})$	$p(\mathbf{z} \mathbf{i})$			[10]
$p(\mathbf{s} \mathbf{z}, \mathbf{i})p(\mathbf{z})$	$p(\mathbf{z})$	✓		[39]
$p(\mathbf{s} \mathbf{z}, \mathbf{i})p(\mathbf{z})$	$p(\mathbf{z})$			[34]
$p(\mathbf{s} \mathbf{z})p(\mathbf{z} \mathbf{i})$	$p(\mathbf{z} \mathbf{i})$		✓	[31, 44]
$p(\mathbf{s} \mathbf{z})p(\mathbf{z} \mathbf{i})$	$p(\mathbf{z} \mathbf{i})$	✓	✓	[43]
$p(\mathbf{s} \mathbf{z})p(\mathbf{z} \mathbf{i})$	$q(\mathbf{z} \mathbf{s})$	✓		[6]
$p(\mathbf{s} \mathbf{z})p(\mathbf{z} \mathbf{i})$	$q(\mathbf{z} \mathbf{s})$			[24]

Table 1: Overview of how related work fits into our probabilistic reconstruction framework.

3 Related work

In this section we review related work on single image 3D shape reconstruction, and relate it to our generic latent variable model presented in the previous section.

Representations for 3D shape inference. The majority of works studying the inference-based single image 3D shape reconstruction introduce new shape representations and suitable neural network architectures. The seminal works by Choy *et al.* [4] and Girdhar *et al.* [6] used 3D CNNs to predict voxel occupancy grids. To reduce the computational complexity of the voxel grid representation Tatarchenko *et al.* [36] proposed an architecture to process octrees computed on top of the voxel grids. Richter and Roth [28] proposed to use a set of six depth maps to represent voxel grids and to combine a series of such sets in a nested manner to model non-trivial shapes. Su *et al.* [34] combined 2D CNNs with fully-connected networks to output point clouds, which are learned by optimizing Chamfer distance or differentiable approximation of earth mover’s distance. Wang *et al.* [39] applied graph-convolutional networks [2] to the mesh-based shape representation. Shin *et al.* [29] proposed to predict multiple depth maps and according silhouettes and fuse them into meshes by post-processing with Poisson reconstruction algorithm.

Although related in their overall goals, these approaches are difficult to compare since they use target shapes approximated to different degrees. Ideally, a fair comparison across shape representations should be performed while maintaining the same level of granularity across representations, and for different levels, since it is possible that some representations work better for rough shape reconstruction, while other are best for detailed reconstruction.

Image-shape consistency and additional data. Another significant stream of works originates from the idea of ensuring consistency between input data and target shapes. Initial work by Yan *et al.* [45] introduced the consistency between 3D shapes and their silhouettes produced by different viewpoints in a form of a loss function. Similar ideas were investigated by Wiles and Zisserman [40]. Tulsiani *et al.* [37] expanded this approach by the use of differentiable ray tracing in the loss function, ensuring correspondence of inferred voxelized shapes to foreground segmentation masks and depth images. Wu *et al.* [42] developed the idea even further and introduced a two-step reconstruction framework. The first part of the model is trained to infer 2.5D shape sketches (unions of segmentation, depth and normal maps) from images, while the second is separately trained to predict shapes from 2.5D sketches. Both components are then fine-tuned, using reprojection consistency.

Henderson and Ferrari [11] proposed a probabilistic framework for image generation conditioned on a latent shape variable and an additional latent variable for the shape pose. This framework was used to train an underlying 3D mesh generator with the help of differentiable rendering of 3D meshes into images. Differentiable point clouds [13] closed the

Dependencies	Sampling	Deterministic	Shape model	IoU \uparrow (0.5)	IoU \uparrow (0.4)
$p(\mathbf{v} \mathbf{z})p(\mathbf{z} \mathbf{i})$	$p(\mathbf{z} \mathbf{i})$			63.7	65.0
$p(\mathbf{v} \mathbf{z}, \mathbf{i})p(\mathbf{z} \mathbf{i})$	$p(\mathbf{z} \mathbf{i})$			64.6	65.6
$p(\mathbf{v} \mathbf{z}, \mathbf{i})p(\mathbf{z})$	$p(\mathbf{z})$			64.0	65.0
$p(\mathbf{v} \mathbf{z})p(\mathbf{z} \mathbf{i})$	$q(\mathbf{z} \mathbf{v})$			65.9	66.2
$p(\mathbf{v} \mathbf{z}, \mathbf{i})p(\mathbf{z} \mathbf{i})$	$q(\mathbf{z} \mathbf{v})$			64.8	65.3
$p(\mathbf{v} \mathbf{z})p(\mathbf{z} \mathbf{i})$	$q(\mathbf{z} \mathbf{v}, \mathbf{i})$			65.4	65.8
$p(\mathbf{v} \mathbf{z})p(\mathbf{z} \mathbf{i})$	$q(\mathbf{z} \mathbf{v})$	✓		63.4	63.7
$p(\mathbf{v} \mathbf{z})p(\mathbf{z} \mathbf{i})$	$q(\mathbf{z} \mathbf{v})$		✓	65.6	66.1

Table 2: Evaluation results for variations of PRN. Monte Carlo training uses samples from the unconditional or image-informed prior, while variational training relies on samples from the shape-conditioned approximate posterior. We report IoU under two occupancy probability thresholds τ .

consistency loop between inferred point clouds and input images, by rendering point clouds as images and minimizing loss between such renderings and input images.

Similarly to the previous class of models, these methods also enrich available training data by considering different forms of additional data: camera information, 2.5D sketches, *etc.* This, again, makes comparison problematic, since it is not always clear what the impact of the additional training data is.

Relations to our framework. In addition to the work discussed above, given similarities between VAEs and GANs [12], our work is also related to adversarial approaches involving shape discriminators [61, 81, 83]. In Table 1 we organize related work in terms of how it fits into our generic probabilistic reconstruction framework, abstracting away from the implementation of various components.

We see that most previous works use a dependency structure where the latent variable is inferred from the image, and the shape decoder only depends on the latent variable and not on the image. Moreover, most works rely on deterministic models, except for GAN-based approaches of [61, 81], the point cloud based approaches of [20, 62], and the mesh based method of [11]. Finally, only TL-Networks [8] and 3D-LMNet [20] make use of the variational inference for shape modelling.

4 Experiments

In this section we present the experimental setup, our quantitative and qualitative evaluation results, as well as our analysis of these results.

4.1 Dataset, evaluation metrics, and experimental details

Dataset. We evaluate PRNs on a subset of the ShapeNet dataset [9] introduced by Choy *et al.* [9]. It contains about 44k 3D shapes from 13 major categories of ShapeNet dataset represented as voxel grids of resolution 32^3 , as well as renderings from 24 different randomized viewpoints as 137^2 images. Following Choy *et al.*, we use 80% of the shapes from each category for training and remaining shapes for testing.

Evaluation metrics. We evaluate using the standard intersection-over-union (IoU) metric [9], which averages the per-category IoU metric between the inferred shape and the ground-truth voxel representation. In addition, to allow comparison to recent work based

on point-cloud and mesh based representations, we also report the Chamfer distance (CD) and earth mover’s distance (EMD), computed using the code of [65], where we removed the square root from the distance computations in CD to make it comparable to the related work. In particular, each ground truth and predicted voxel grid is mapped to a point cloud by sampling the surface induced using the marching cubes algorithm [18]. We then compute the CD and EMD on the resulting point clouds.

Training and evaluating. When using either Monte Carlo approximation or a variational objective function, we always use a single sample to compute the gradients during training. A unified training protocol is used for all the models: all components are trained simultaneously (contrary to [6, 20, 43]), with the AMSGrad [26] optimizer with decoupled weight decay regularization [19] with step-like scheduling for learning rate and weight decay parameter, and restarts of gradient moments accumulation at the beginning of each step.

During testing, we use a deterministic approach. In particular, we take the means of the conditional distributions rather than samples from them. We found this to significantly improve the reconstruction quality, compared to sampling.

4.2 Experimental results

Evaluation of PRN variants. To explore the various possibilities of our general latent variable model, we consider three options to condition on the image: (i) using the latent space to carry all image information: $p(\mathbf{v}|\mathbf{z})p(\mathbf{z}|\mathbf{i})$, (ii) using additional conditioning of the decoder on the image: $p(\mathbf{v}|\mathbf{z}, \mathbf{i})p(\mathbf{z}|\mathbf{i})$, and (iii) using an uninformed prior on the latent variable: $p(\mathbf{v}|\mathbf{z}, \mathbf{i})p(\mathbf{z})$. To train the models we either approximate the integral in Eq. (1) directly with Monte Carlo samples from the prior on \mathbf{z} , or with a variational lower bound and samples from the variational posterior. We also test a deterministic model, where the distribution on the latent variable is replaced by a deterministic function. Finally, we consider the option to train the model jointly with an unconditional generative model. In Table 2 we present the results, using two thresholds τ on the voxel occupancy probability: the neutral 0.5, and following [9] the looser 0.4 which overall leads to improved IoU scores.

In case of the Monte Carlo approximation (top three rows), additional image conditioning in the decoder improves the results. Conditioning both the latent variable prior and the decoder on the image achieves best results, suggesting that these different pathways to use the image are complementary.

The use of variational training consistently improves the results over the Monte Carlo approximation. In this case, the additional image conditioning of the decoder or the variational posterior, see line five and six, is not effective and even somewhat reduces the performance. This is contrary to the results obtained with Monte Carlo sampling; in the latter case the sampling inefficiency is probably partially compensated by the additional conditioning pathway. Variational inference leads to more accurate samples, which obviates the need for the additional image conditioning (at least for the chosen mechanism of the additional image conditioning).

We also consider a deterministic variant of our best performing model, which resembles the TL-network of [9]. The results show that probabilistic handling of the latent variable reduces overfitting in the model and leads to IoU of 2.5 points higher. Finally, we also tested the training with a joint generative shape model, which TL-Networks used as pre-training. Although we did not observe a significant effect due to the joint training with a generative shape model, it does offer additional functionality by being able to sample shapes,

Model	Image res.	Output	IoU \uparrow	CD \downarrow	EMD \downarrow
3D-R2N2 [4]	127 ²	voxels 32 ³	56.0	7.10 [†]	10.20 [†]
OGN [34]	137 ²	voxel octrees 32 ³	59.6	—	—
PSGN [24]	128 ²	points 1024	64.0	2.50	8.00
AtlasNet [8]	224 ²	points 2500	—	5.11	—
Pixel2Mesh [39]	224 ²	meshes 2466	—	5.91	13.80
3D-LMNet [20]	128 ²	points 2048	—	5.40	7.00
PRN (ours)	137 ²	voxels 32 ³	66.2	4.42	6.32

Table 3: Comparison of PRN to the state-of-the-art. All results are taken from the original papers, except for \dagger , which were provided in [24]. Pixel2Mesh additionally uses camera information and surface normals during training.

or compute their likelihoods under the model.

Comparison to the state-of-the-art. In Table 3 we compare to earlier state-of-the-art approaches. All methods use the same input images, but use slightly different image preprocessing: 3D-R2N2 uses random cropping, 3D-LMNet central cropping, AtlasNet crops and resizes, while PSGN and Pixel2Mesh resize the image. As OGN, we use original images, but also add a grey-scale version of each image as a fourth input channel.

In Table 3 we report Chamfer distance computed for 1024 predicted and 1024 ground truth points as it was done in most of the related work. Although they all used different numbers of predicted and ground truth points for training, The authors of [8, 20, 39] explicitly state this protocol for evaluation, while there is no information about it in the text of [24]. Judging from the code of [24], we assume that, in their case, the metric was obtained for 1024 predicted and 16384 ground truth points. For the reference, we recomputed CD under the same protocol and obtained a better value: 3.90. This shows that the metric is affected by a negative bias, which decreases with the increasing number of evaluated points, and underlines the need for unified evaluation protocol.

Our PRN obtains excellent results, and significantly improves over previous state-of-the-art results in terms of IoU and EMD, including methods based on point cloud and mesh representations. Point-based approaches use loss functions based on the Chamfer distance, and so naturally perform well in terms of this metric, but this does not per se transfer to better performance in the other metrics. In our case, we do not explicitly optimize for either of these metrics, relying on the binary cross-entropy for the voxel occupancies instead, and yet obtain best results in two of the three metrics (with only one competitor being better in terms of the third metric).

Qualitative reconstruction results. In Figure 2 we provide a selection of qualitative reconstruction results. We show results for the models in rows one, four and seven in Table 2, *i.e.* with the $p(\mathbf{v}|\mathbf{z})p(\mathbf{z}|\mathbf{i})$ dependency structure, using Monte Carlo (PRN MC) and variational (PRN var.) training, and the deterministic version of the latter (PRN var. det.). We show four examples where the variationally trained model is the best, and one case where it is the worst. Overall, variationally trained model output fewer failed reconstructions, as well as more detailed reconstructions, compared to more failures and over simplified reconstructions from the model trained with Monte Carlo. For reference, the average IoU score of the variationally trained model is 66.2 (median 69.9), which corresponds to a fairly accurate reconstruction level, in particular given the challenging nature of the task.


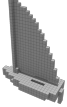

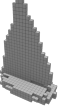
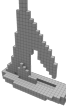

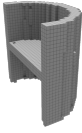
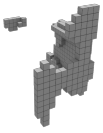
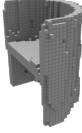
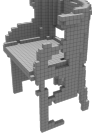
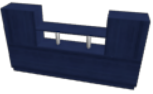
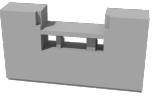

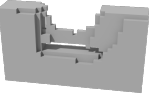


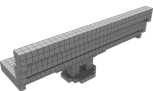
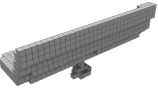
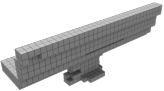
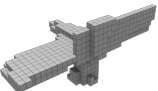





image input	ground truth	PRN MC	PRN var.	PRN var. det.
		 12.8	 74.0	 45.5
		 7.8	 68.5	 17.3
		 85.4	 86.9	 23.0
		 72.7	 87.6	 30.4
		 48.9	 43.9	 47.2

Figure 2: Qualitative reconstruction results for three variants of PRNs.

5 Conclusion

In this paper we presented Probabilistic Reconstruction Networks, a generic probabilistic framework for 3D shape inference from single image. This framework naturally decouples different aspects of the problem, including the shape representation, the image conditioning structure, and the usage of the latent shape space. In our experiments with voxel-grid shape representations, we systematically explored the impact of image conditioning, Monte Carlo *vs.* variational likelihood approximation for training, the stochastic nature of the latent variable, and joint training with a generative shape model. We obtained single image shape reconstruction results that surpass the previous state of the art in terms of the IoU and EMD performance metrics, and outperform recent work based on point-cloud and mesh-based shape representation.

Given the interpretation of the inference-based reconstruction as an instance of conditional generation, future work includes further adaptation of the generative modelling approaches to the task, as well as the investigation of different shape representations within the proposed framework.

Acknowledgements

This work has been partially supported by the grant “Deep in France” (ANR-16-CE23-0006) and LabEx PERSYVAL-Lab (ANR-11-LABX-0025-01).

A Histograms of IoU for selected methods

In Figure 3 we provide a histogram of the IoU scores obtained across the shapes in the ShapeNet test set using three of the model evaluated in Table 2:

- Using $p(\mathbf{v}|\mathbf{z})p(\mathbf{z}|\mathbf{i})$, with Monte Carlo training (Table 2, row 1).
- Using $p(\mathbf{v}|\mathbf{z})p(\mathbf{z}|\mathbf{i})$, with variational training (Table 2, row 4).
- Using $p(\mathbf{v}|\mathbf{z})p(\mathbf{z}|\mathbf{i})$, with deterministic modeling (Table 2, row 7).

For each shape in the test set there are 24 views, giving a total of about 210k shape inferences.

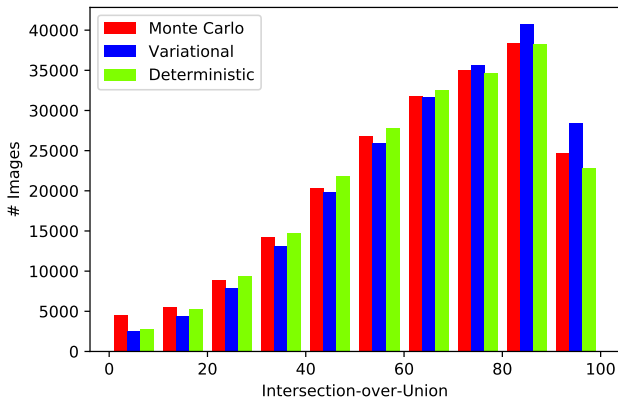


Figure 3: Histogram of IoU values on the ShapeNet test set for the Monte Carlo, variational, and deterministic model. See text for details. Bins of size 10 from 0 to 10, then 10 to 20, *etc.*

The histograms show that the variational learning approach leads to more accurate reconstructions, leading to the largest number of reconstructed shapes in the last three bins for shape with $> 70\%$ IoU. For all other bins of less accurate results, the variational method has the smallest number of shapes.

Compared to the deterministic model, Monte Carlo training leads to more accurate reconstructions, but also to more very poor reconstructions.

B Visualization of shape reconstruction results

In this section we provide visualization of additional shape reconstruction results, similar to the ones presented in Section 4. Contrary to them, we put randomly sampled examples here.

For each example in Figure 4, we show from left to right:

- the input image;
- ground-truth shape;
- inferred shape with Monte Carlo training;
- inferred shape with variational training;
- inferred shape with deterministic model.

These shape inference approaches correspond to rows one, four, and seven of Table 2.


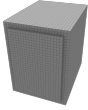
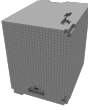
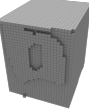
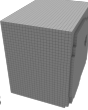

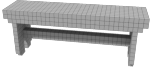
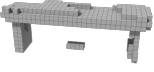
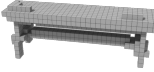
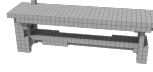

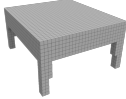
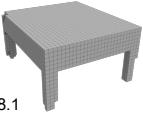
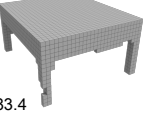
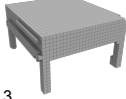









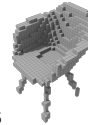

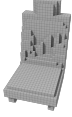
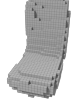




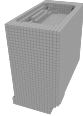

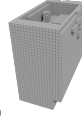

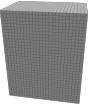
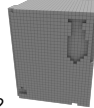
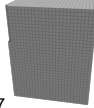
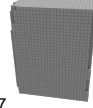
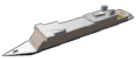
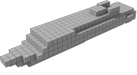
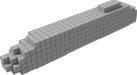
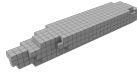
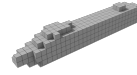


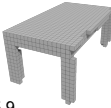
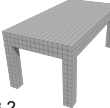
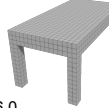
input image	ground truth	PRN MC	PRN var.	PRN var. det
		 95.9	 90.5	 91.8
		 56.4	 50.5	 45.2
		 98.1	 83.4	 73.3
		 72.3	 66.9	 68.0
		 56.0	 57.7	 49.5
		 55.1	 44.0	 51.4
		 88.1	 90.0	 81.9
		 52.2	 99.7	 93.7
		 59.0	 79.5	 65.3
		 85.9	 93.2	 96.0

Figure 4: Reconstruction results for random input images from the test set.

References

- [1] A. Brock, T. Lim, J. Ritchie, and N. Weston. Generative and discriminative voxel modeling with convolutional neural networks. In *NeurIPS 3D deep learning workshop*, 2016.
- [2] M. Bronstein, J. Bruna, A. Szlam, Y. LeCun, and P. Vandergyst. Geometric deep learning: going beyond Euclidean data. *IEEE Signal Processing Magazine*, 34(4):18–42, 2017.
- [3] A. 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. *arXiv preprint*, abs/1512.03012, 2015.
- [4] C. Choy, D. Xu, J.-Y. Gwak, K. Chen, and S. Savarese. 3D-R2N2: A unified approach for single and multi-view 3d object reconstruction. In *ECCV*, 2016.
- [5] R. Girdhar, D. Fouhey, M. Rodriguez, and A. Gupta. Learning a predictable and generative vector representation for objects. In *ECCV*, 2016.
- [6] J. Gordon, J. Bronskill, M. Bauer, S. Nowozin, and R. Turner. Meta-learning probabilistic inference for prediction. In *ICLR*, 2019.
- [7] B. Graham, M. Engelcke, and L. van der Maaten. 3D semantic segmentation with submanifold sparse convolutional networks. In *CVPR*, 2018.
- [8] T. Groueix, M. Fisher, V. Kim, B. Russell, and M. Aubry. A papier-mâché approach to learning 3D surface generation. In *CVPR*, 2018.
- [9] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *CVPR*, 2016.
- [10] K. He, X. Zhang, S. Ren, and J. Sun. Identity mappings in deep residual networks. In *ECCV*, 2016.
- [11] P. Henderson and V. Ferrari. Learning to generate and reconstruct 3d meshes with only 2d supervision. In *BMVC*, 2018.
- [12] Z. Hu, Z. Yang, R. Salakhutdinov, and E. P. Xing. On unifying deep generative models. In *ICLR*, 2018.
- [13] E. Insafutdinov and A. Dosovitskiy. Unsupervised learning of shape and pose with differentiable point clouds. In *NeurIPS*, 2018.
- [14] D. Kingma and M. Welling. Auto-encoding variational Bayes. In *ICLR*, 2014.
- [15] R. Klokov and V. Lempitsky. Escape from cells: Deep Kd-networks for the recognition of 3D point cloud models. In *ICCV*, 2017.
- [16] A. Krizhevsky, I. Sutskever, and G. Hinton. ImageNet classification with deep convolutional neural networks. In *NeurIPS*, 2012.
- [17] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998.

- [18] T. Lewiner, H. Lopes, A. Vieira, and G. Tavares. Efficient implementation of marching cubes' cases with topological guarantees. *J. Graphics, GPU, & Game Tools*, 8(2):1–15, 2003.
- [19] I. Loshchilov and F. Hutter. Decoupled weight decay regularization. In *ICLR*, 2019.
- [20] P. Mandikal, K. Navaneet, M. Agarwal, and R. Babu. 3D-LMNet: Latent embedding matching for accurate and diverse 3D point cloud reconstruction from a single image. In *BMVC*, 2018.
- [21] D. Maturana and S. Scherer. VoxNet: A 3D convolutional neural network for real-time object recognition. In *IROS*, 2015.
- [22] F. Monti, D. Boscaini, J. Masci, E. Rodolà, J. Svoboda, and M. Bronstein. Geometric deep learning on graphs and manifolds using mixture model CNNs. In *CVPR*, 2017.
- [23] E. Perez, F. Strub, H. De Vries, V. Dumoulin, and A. Courville. FiLM: Visual reasoning with a general conditioning layer. In *AAAI*, 2018.
- [24] C. Qi, H. Su, K. Mo, and L. Guibas. Pointnet: Deep learning on point sets for 3D classification and segmentation. In *CVPR*, 2017.
- [25] C. Qi, L. Yi, H. Su, and L. Guibas. Pointnet++: Deep hierarchical feature learning on point sets in a metric space. In *NeurIPS*, 2017.
- [26] S. Reddi, S. Kale, and S. Kumar. On the convergence of Adam and beyond. In *ICLR*, 2018.
- [27] D. Rezende, S. Mohamed, and D. Wierstra. Stochastic backpropagation and approximate inference in deep generative models. In *ICML*, 2014.
- [28] S. R. Richter and S. Roth. Matryoshka Networks: Predicting 3d geometry via nested shape layers. In *CVPR*, 2018.
- [29] D. Shin, C. C. Fowlkes, and D. Hoiem. Pixels, voxels, and views: A study of shape representations for single view 3d object shape prediction. In *CVPR*, 2018.
- [30] A. Sinha, J. Bai, and K. Ramani. Deep learning 3D shape surfaces using geometry images. In *ECCV*, 2016.
- [31] E. Smith and D. Meger. Improved adversarial systems for 3d object generation and reconstruction. In *CoRL*, 2017.
- [32] A. Soltani, H. Huang, J. Wu, T. Kulkarni, and J. Tenenbaum. Synthesizing 3D shapes via modeling multi-view depth maps and silhouettes with deep generative networks. In *CVPR*, 2017.
- [33] H. Su, S. Maji, E. Kalogerakis, and E. Learned-Miller. Multi-view convolutional neural networks for 3D shape recognition. In *ICCV*, 2015.
- [34] H. Su, H. Fan, and L. Guibas. A point set generation network for 3D object reconstruction from a single image. In *CVPR*, 2017.

- [35] X. Sun, J. Wu, X. Zhang, Z. Zhang, C. Zhang, T. Xue, J. Tenenbaum, and W. Freeman. Pix3D: Dataset and methods for single-image 3D shape modeling. In *CVPR*, 2018.
- [36] M. Tatarchenko, A. Dosovitskiy, and T. Brox. Octree generating networks: Efficient convolutional architectures for high-resolution 3D outputs. In *ICCV*, 2017.
- [37] S. Tulsiani, T. Zhou, A. Efros, and J. Malik. Multi-view supervision for single-view reconstruction via differentiable ray consistency. In *CVPR*, 2017.
- [38] N. Verma, E. Boyer, and J. Verbeek. Feastnet: Feature-steered graph convolutions for 3D shape analysis. In *CVPR*, 2018.
- [39] N. Wang, Y. Zhang, Z. Li, Y. Fu, W. Liu, and Y. Jiang. Pixel2Mesh: Generating 3D mesh models from single RGB images. In *ECCV*, 2018.
- [40] O. Wiles and A. Zisserman. SilNet: Single- and multi-view reconstruction by learning from silhouettes. In *BMVC*, 2017.
- [41] J. Wu, C. Zhang, T. Xue, W. Freeman, and J. Tenenbaum. Learning a probabilistic latent space of object shapes via 3D generative-adversarial modeling. In *NeurIPS*, 2016.
- [42] J. Wu, Y. Wang, T. Xue, X. Sun, B. Freeman, and J. Tenenbaum. MarrNet: 3D shape reconstruction via 2.5D sketches. In *NeurIPS*, 2017.
- [43] J. Wu, C. Zhang, X. Zhang, Z. Zhang, W. Freeman, and J. Tenenbaum. Learning shape priors for single-view 3D completion and reconstruction. In *ECCV*, 2018.
- [44] Z. Wu, S. Song, A. Khosla, F. Yu, L. Zhang, X. Tang, and J. Xiao. 3D ShapeNets: A deep representation for volumetric shapes. In *CVPR*, 2015.
- [45] X. Yan, J. Yang, E. Yumer, Y. Guo, and H. Lee. Perspective Transformer Nets: Learning single-view 3D object reconstruction without 3D supervision. In *NeurIPS*, 2016.