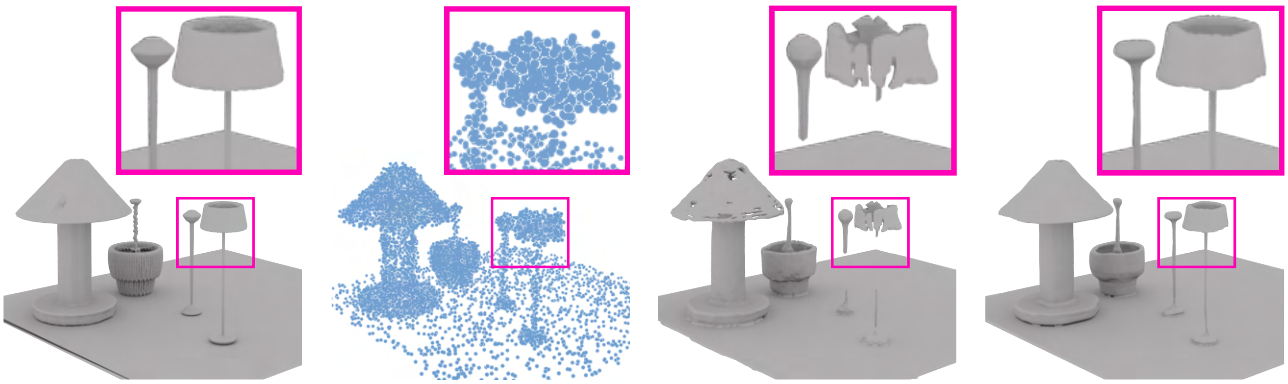
ALTO: Alternating Latent Topologies for Implicit 3D Reconstruction

Zhen Wang[1*] Shijie Zhou[1*] Jeong Joon Park[2] Despoina Paschalidou[2]
Suya You[3] Gordon Wetzstein[2] Leonidas Guibas[2] Achuta Kadambi[1]
[1]University of California, Los Angeles [2]Stanford University [3]DEVCOM Army Research Laboratory



(a) Ground Truth (b) Input (10,000 points) (c) POCO [4] (36.8 sec) (d) **ALTO (10.5 sec)**

Figure 1. **Rethinking latent topologies for fast and detailed implicit 3D reconstructions**. Recent work (POCO CVPR'22 [4]) has used latent encodings for each point to preserve 3D detail. We introduce ALTO, which can alternate between latent topologies like grid latents and point latents to speed up inference and recover more detail, like the 3D reconstruction of a thin lamp-post. Scene from [52].

Abstract

This work introduces alternating latent topologies (ALTO) for high-fidelity reconstruction of implicit 3D surfaces from noisy point clouds. Previous work identifies that the spatial arrangement of latent encodings is important to recover detail. One school of thought is to encode a latent vector for each point (point latents). Another school of thought is to project point latents into a grid (grid latents) which could be a voxel grid or triplane grid. Each school of thought has tradeoffs. Grid latents are coarse and lose high-frequency detail. In contrast, point latents preserve detail. However, point latents are more difficult to decode into a surface, and quality and runtime suffer. In this paper, we propose ALTO to sequentially alternate between geometric representations, before converging to an easy-to-decode latent. We find that this preserves spatial expressiveness and makes decoding lightweight. We validate ALTO on implicit 3D recovery and observe not only a performance improvement over the state-of-the-art, but a runtime improvement of 3-10×. Project website at https://visual.ee.ucla.edu/alto.htm/.

*Equal contribution.

1. Introduction

Reconstructing surfaces from noisy point clouds is an active problem in 3D computer vision. Today, conditional neural fields offer a promising way to learn surfaces from noisy point clouds. Alternatives like voxel regression or mesh estimation are limited by cubic complexity and the requirement of a mesh template, respectively. Recent work has successfully used conditional neural fields to reconstruct 3D surfaces as an occupancy function. A conditional neural field takes as input a query coordinate and conditions this on a latent representation, e.g., feature grids. The spatial expressiveness of the latent representation impacts the overall surface reconstruction quality.

To achieve spatial expression, a neural field is conditioned on a latent space of features (**latents**) from the conditional input. In 3D surface reconstruction the input point cloud is transformed into latents arranged in some topological structure. **Point latents** occur when each point in the input point cloud is assigned a latent vector [4]. **Triplane latents** are formed when point latents are projected into a 3-axis grid [41, 52]. The triplane latent is not as spatially expressive as freeform points, but the lower spatial complexity makes it easier to decode. **Voxel latents** are another type of grid latent where latents are arranged in a feature

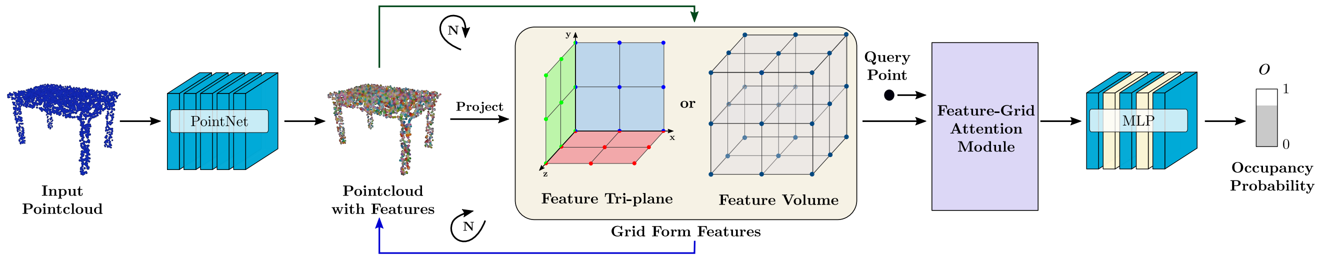


Figure 2. **An overview of our method.** Given input surface points, we obtain an implicit occupancy field with iterative alternation between features in the forms of points and 2D or 3D grids (Sec. 3.2). Then we decode the occupancy values for query points with a learned attention-based interpolation from neighboring grids (Sec. 3.3).

volume [52, 66].

To reconstruct detailed surfaces, recent state-of-the-art methods try to preserve point latents as long as possible. Because point latents are spatially expressive, methods based on point latents are considered state-of-the-art for detailed surface reconstruction [4, 18]. However, using point latents in this way has some tradeoffs. It is difficult to correlate a query with the unstructured topology of a point-based latent space, placing a burden on the decoder. Results from POCO [4] are shown in Fig. 1 where runtime and high-quality detail like thin lampposts remain out of reach.

In this paper, we seek to blend the upside of different latent topologies, while minimizing downside. We present an alternating latent topology (ALTO) method. In contrast to previous work, our method does not stay with either point [4] or grid latents [52], but instead alternates back and forth between point and grid latents before converging to a final grid for ease-of-decoding.

Our method is general. We can plug-in the ALTO component to existing grid-based conditional models [10, 52] to boost detail recovery. While we have shown that our method can generate occupancy fields, we expect gain of high-fidelity details for other neural fields, such as semantic or affordance fields [32, 70], where similar conditional techniques can be adopted.

We summarize our **contributions** as follows:

- We introduce an iterative technique to blend the strengths of different latent topologies for high-fidelity conditional neural fields generation.
- We propose an attention-based decoder that replaces naive linear interpolation of feature-grids or computationally expensive point-wise attention while keeping compute burden in check.
- We demonstrate performance and runtime improvements over the highest-quality previous method [4], as well as performance improvements over all other baselines.

2. Related Work

3D reconstruction is a very important topic in both computer vision and computational imaging [3]. In this section, we discuss the most relevant literature on learning-based 3D reconstruction methods. Based on their output, existing learning-based approaches can be categorized as implicit or explicit-based representations. In this work, we primarily focus on implicit-based representations as they are closely related to our method.

2.1. Explicit Representations

A common shape representation is 2.5D depth maps [28, 29, 33, 34, 36, 42, 50, 75, 82, 83], which can be inferred using 2D CNNs [73, 74, 81, 84]. However, 2.5D depth maps cannot capture the full 3D geometry. In contrast, voxels [5, 12, 15, 21, 45, 56, 64, 76–78] naturally capture 3D object geometry, by discretizing the shape into a regular grid. As voxel-based methods exhibit cubic space complexity that results in high memory and computation requirements, several works tried to circumvent this with efficient space partitioning techniques [27, 45, 57, 58, 67]. Although these methods allow for increasing the voxel resolution and hence capturing more complex geometries, their application is still limited. Recently, a promising new direction explored learning grid deformations to better capture geometric details [22]. An alternative representation relies on pointclouds. Point-based approaches [1, 20, 31, 55, 69, 80] discretize the 3D space using points and are more lightweight and memory efficient. However, as they lack surface connectivity, they require additional post-processing steps (e.g. using Poisson Surface Reconstruction [38]) for generating the final mesh. Instead, mesh-based methods [8, 16, 24, 26, 35, 40, 48, 71, 80] naturally yield smooth surfaces but they typically require a template mesh [71], which makes scaling them to arbitrarily complex topologies difficult. Other works, proposed to also represent the geometry as an atlas of mappings [17, 26, 44], which can result in non-watertight meshes. To address limitations with learning explicit representations, implicit models [9, 46, 49, 62] emerged as an alternative more compact representation that yield 3D geometries at infinitely high resolutions using iso-surfacing operations (i.e. marching cubes). In this work, we capture 3D geometries implicitly, using an occupancy field [46], as it faithfully can capture complex topologies.

2.2. Neural Implicit Representations

Unlike explicit representations that discretize the output space using voxels, points or mesh vertices, implicit representations represent the 3D shape and appearance implicitly, in the latent vectors of a neural network that learns a mapping between a query point and a context vector to either a signed distance value [2, 25, 47, 49, 65] or a binary occupancy value [9, 46, 63]. However, while these methods typically rely on a single global latent code, they cannot capture local details and struggle scaling to more complicated geometries. To address this, several works [60, 61, 79] explored pixel-aligned implicit representations, that rely on both global and local image features computed along a viewing direction. While, these approaches are able to capture fine-grained geometric details, they rely on features that are computed from images, hence are limited to image-based inputs with known camera poses.

Our work falls in the category of methods that perform 3D reconstructions from points. Among the first to explore this direction were [10, 30, 52]. To increase the expressivity of the underlying representation and to be able to capture complex geometries, instead of conditioning on a global latent code, these works condition on local per-point features. For example, Jiang et al. [30] leverage shape priors by conditioning on a patch-based representation of the point cloud. Other works [41, 52, 66], utilize grid-based convolutional features extracted from feature planes [41], feature volumes [10, 66] or both [52]. An alternative line of work, [72] introduce a test-time optimization mechanism to refine the per-point features predicted on a feature volume. Concurrently, POCO [4], propose to estimate per-point features, which are then refined based on the per-point features of their neighboring points using an attention-based mechanism. A similar idea is also explored in Points2Surf [19] that introduces a patch-based mechanism to decide the sign of the implicit function. Our work is closely related to POCO [4] that can faithfully capture higher-frequency details due its point-wise latent coding. However, the lack of a grid-like structure places extra complexity on the attention-based aggregation module, which results in a higher computation cost. AIR-Net [23] applied local attention to reduce the computation but limits its operating range to objects. In this work, we propose conditioning on a hybrid representation of points and grid latents. In particular, instead of fusing points and grids, we demonstrate that it is the point and grid *alternation* between points and grids that enables recovery of more detail than POCO [4], while reducing compute time by an order of magnitude.

3. Method

There are three insights that motivate our approach: (1) conditioning on the right topology of the latent space is important; (2) previous neural fields for surface reconstruction condition on point or grid latents; (3) both point and grid latents have complementary strengths and weaknesses. Point latents are more spatially expressive but grid latents are easier to decode into a surface.

It might seem like a simple concatenation of point and grid latents would be sufficient. The problem is that point latents remain difficult to decode (even if concatenated with a grid latent). Therefore, our insight is to alternate between grid and point latents, and converge to a grid latent. For feature triplane latents, the alternation also permits communication between the individual planes, which in previous, grid-based works would have been fed into independent hourglass U-Nets [52].

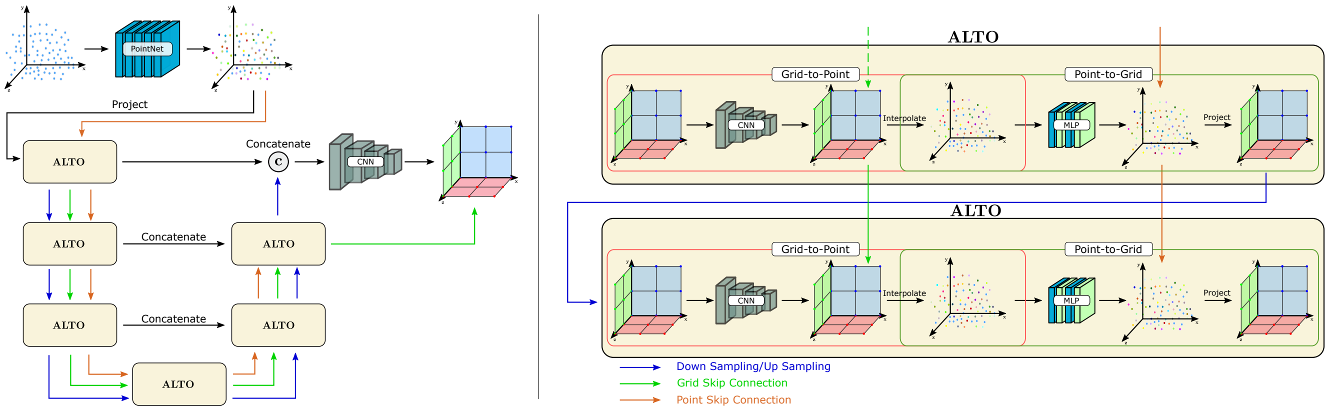
In this section, we introduce a version of ALTO as a point-grid alternating U-Net. An overview of the method is shown in Fig. 2. We demonstrate how the convolutional grid form is learned in Sec. 3.1, our point-grid alternating network in Sec. 3.2, our attention-based decoder in Sec. 3.3 and training and inference in more detail in Sec. 3.4.

3.1. Convolutional Feature Grids

The input to our method is a noisy un-oriented point cloud $\mathcal{P} = \{\boldsymbol{p}_i \in \mathbb{R}^3\}_{i=1}^S$, where S is the number of input points. We first use a shallow Point-Net [54] to obtain the initial point features as in ConvONet [52]. These point features are then projected into three 2D grids (feature triplanes) $\{\mathbf{\Pi}_{xy}, \mathbf{\Pi}_{xz}, \mathbf{\Pi}_{yz}\} \in \mathbb{R}^{H\times W\times d}$ or 3D grids (feature volumes) $\mathbf{V} \in \mathbb{R}^{H\times W\times D\times d}$ before feeding into a 2D or 3D convolutional hourglass (U-Net) networks [13, 59]. d is the number of feature channels. For feature volumes, we set $H = W = D = 64$ due to memory overhead of 3D-CNN and for feature triplanes, H and W can be set as high as 128 depending on the task.

3.2. ALTO Latent to Blend Grid and Point Latents

Without loss of generality, we demonstrate ALTO in the context of blending grid and point latents. Note that naive concatenation of latents would not work, as the point latents are difficult to decode. The goal is to use ALTO to blend point latent characteristics into a grid latent via alternation. The alternating block is illustrated in Fig. 3 and incorporated into a U-Net architecture. At each alternation, we first do grid-to-point conversion where convolutional grid features are transformed into point features, followed by point-to-grid conversion where extracted point features are transformed back into grid features for next alternation.



(a) (b)

Figure 3. **An illustration of our ALTO encoder.** (a) As an example, we show the ALTO block instantiated by alternating between two latent topologies: point and triplanes via an "in-network" fashion, i.e. within each level of an hourglass framework U-Net. 'Concatenate' refers to concatenation of the ALTO block output triplane in the downsampling stage and the ALTO block input triplane in the corresponding upsampling stage. (b) We expand on ALTO block to illustrate the sequential grid-to-point and point-to-grid conversion. There are skip connections for both point and grid features between two consecutive levels in the ALTO U-Net.

Grid-to-Point Conversion: At each alternation, to aggregate local neighborhood information, we use convolutional operations for the grid features. We then project each point p orthographically onto the canonical planes and query the feature values through bilinear interpolation for 2D grid and trilinear interpolation for 3D grid. For triplane latents, we sum together the interpolated features from each individual plane.

Point-to-Grid Conversion: At each alternation, given the interpolated point features, we then process point features with an MLP in order to model individual point feature with finer granularity. For feature triplanes grid form, an MLP also gives an additional benefit of having individual plane features communicate with each other. The MLP is implemented with two linear layers and one ReLU nonlinearity. Projected point features falling within the same pixel or voxel cell will be aggregated using average pooling. If using triplane latents with each plane discretized at $H \times W$, this results in planar features with dimensionality $H \times W \times d$ or if using voxel latents we obtain dimensionality $H \times W \times D \times d$, where d is the number of features.

We also adopt skip connections for both point features and grid features between two consecutive ALTO blocks, as illustrated in Fig. 3b. The alternation needs to be implemented carefully to minimize runtime. Naively, we can alternate between triplanes or voxel latents using a U-Net and point latents using MLP, but that would require multiple network passes. Instead, we incorporate the point-grid alternating *inside* each block of a U-Net, i.e. replacing original convolution-only block with ALTO block. We call this single U-Net, the ALTO U-Net. This also enables point and grid features blended at multiple scales and the number of alternation blocks depends on the number of levels of U-Net.

3.3. Decoding ALTO latents using Attention

As discussed in the previous section, ALTO provides a way to get a single latent that blends characteristics of different topologies. The advantage is that the final latent that ALTO converges to (hereafter, ALTO latent) can take on the topology that is easier to decode.

For example, in the case of using ALTO to blend point and grid characteristics, we would like ALTO to converge to a final output in the simpler grid topology. Then, given the ALTO latent in grid form and any query point $\mathbf{q} \in \mathbb{R}^3$ in 3D space, our goal is to decode the learned feature and estimate the occupancy probability of each query point. ALTO benefits from attention on the decoder side. The ALTO latent is in grid form, but has spatial expressivity coming from the blended-in point latents. Standard grid latent decoding, e.g., bi-/tri-linear interpolation used in previous work [41, 52] would not preserve this spatial expressivity.

To decode an ALTO latent, we propose an efficient attention-based mechanism to replace the previous approach of linear interpolation on feature grids. While attention is not new, we leverage *grid latent attention* to avoid heavy runtime issues of *point latent attention* [4] that applies attention over a point-wise 3D neighborhood. As illustrated in Fig. 4, we consider the nearest grids (indices denoted as $\mathcal{N}(\mathbf{q})$), where $|\mathcal{N}| = 9$ for triplane representation and $|\mathcal{N}| = 27$ for volume representation, we call these areas as per-point neighbor feature patches $\mathbf{C}_{\{i \in \mathcal{N}\}}$. We define the query Q, key K, and value V for our attention as

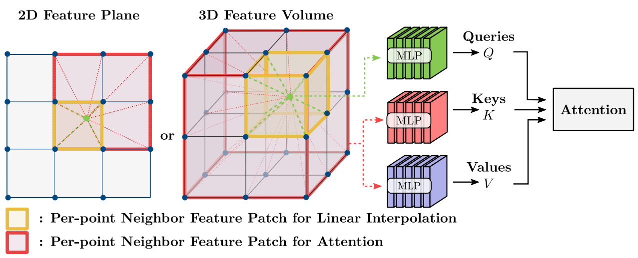


Figure 4. **Attention-based decoder on neighboring grids (2D or 3D).** To obtain features of each query point for decoding occupancy value, we use learned interpolation from neighboring grids that improves occupancy prediction, while being more efficient than expansive point-wise attention mechanism (e.g. POCO [4]).

follows:

$$\begin{aligned} Q &= \text{MLP}(\psi(\mathbf{q})), \\ K &= \text{MLP}(\mathbf{C}_{\{i \in \mathcal{N}(\mathbf{q})\}}), \\ V &= \text{MLP}(\mathbf{C}_{\{i \in \mathcal{N}(\mathbf{q})\}}), \end{aligned} \tag{1}$$

where $\psi(\mathbf{q})$ is the linear interpolated feature value of the query points. Additionally, we compute the displacement vector $\mathbf{d} \in \mathbb{R}^2$ or $\mathbb{R}^3$ which represents the spatial relationship between the projected query point coordinate and the nearest feature grid points. We use the subtraction relation [86] in our attention scoring function:

$$A = \text{softmax}(\text{MLP}((Q - K) + \gamma(\mathbf{d}))), \tag{2}$$

where $\gamma(\mathbf{d})$ works as a learnable positional encoding. In our implementation, γ is an MLP with two fully-connected layers and activated by ReLU. We compute the attention-based interpolated per-point feature F as:

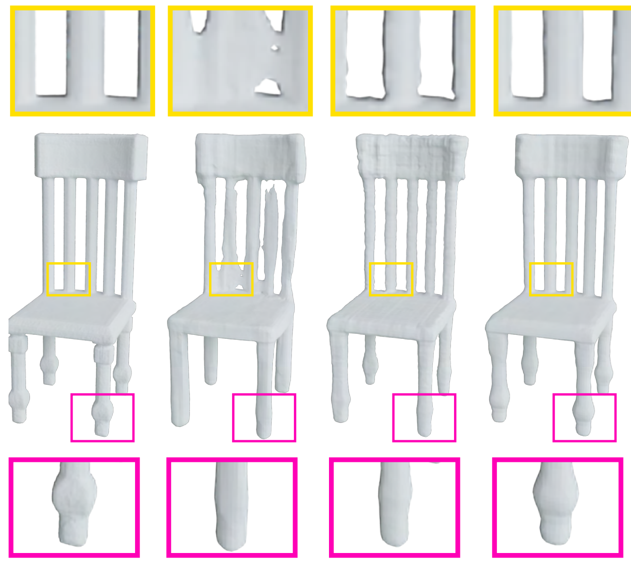
$$F = A \odot (V + \gamma(\mathbf{d})). \tag{3}$$

Note that the same positional encoding from above is added to V and $\odot$ denotes the element-wise product operation. For the case of triplane representation, we use a single-head attention to extract the feature F from each individual plane. The per-triplane features are then concatenated and used for the occupancy prediction. For the case of the volume representation, we use multi-head attention for h independently learned subspaces of Q, K, and V, where h is a hyperparameter varying based on the experiment. Additional details are provided in the supplementary.

Finally, we predict the occupancy of $\mathbf{q}$ using a small fully-connected occupancy network:

$$f_\theta(F) \rightarrow [0, 1]. \tag{4}$$

The network f_θ consists of several ResNet blocks as in [52]. The major difference to the original occupancy decoder in [52] is that we do not bring in the absolute 3D coordinate of $\mathbf{q}$ as input since it theoretically breaks the translational equivalence property.



(a) GT (b) Grid Latent [52] (c) Sec. 3.2 "ALTO (Enc.)" (d) Sec. 3.2+3.3 "ALTO"

Figure 5. **Ablation analysis on ShapeNet.** Note the top inset showing the poles in the chair back (yellow). ALTO (Enc.) is ALTO (Encoder Only) and uses the latent space encoding proposed in Sec. 3.2 with a standard decoder. The full ALTO method includes also the attention-based decoder in Sec. 3.3.

This concludes our description of our latent space encoding and attention decoding. In Fig. 5 observe that the architectures we have proposed progressively improve detail from a standard grid formulation.

3.4. Training and Inference

At training time, we uniformly sample query points $\mathcal{Q}$ and minimize the binary cross-entropy loss between the predicted occupancy value and ground-truth occupancy values written as:

$$\mathcal{L}\left(\hat{o}_{\mathbf{q}}, o_{\mathbf{q}}\right) = -\sum_{\mathbf{q} \in \mathcal{Q}} \left[o_{\mathbf{q}} \log\left(\hat{o}_{\mathbf{q}}\right) + (1 - o_{\mathbf{q}}) \log\left(1 - \hat{o}_{\mathbf{q}}\right)\right] \tag{5}$$

Our model is implemented in PyTorch [51] and uses the Adam optimizer [39] with a learning rate of 10^{-4}. During inference, we use a form of Marching Cubes [43] to obtain the mesh.

4. Experimental Evaluation

4.1. Datasets, Metrics, and Baselines

Object Level Datasets: For evaluation on object-level reconstruction, we use ShapeNet [6]. In particular, ShapeNet [6] contains watertight meshes of object shapes in 13 classes. For fair comparison, we use the same train/val splits and 8500 objects for testing as described in [4, 52].

	input points 3K				input points 1K				input points 300			
Method	IoU ↑	Chamfer-L_1 ↓	NC↑	F-score↑	IoU ↑	Chamfer-L_1 ↓	NC↑	F-score↑	IoU ↑	Chamfer-L_1 ↓	NC↑	F-score↑
ONet [46]	0.761	0.87	0.891	0.785	0.772	0.81	0.894	0.801	0.778	0.80	0.895	0.806
ConvONet [52]	0.884	0.44	0.938	0.942	0.859	0.50	0.929	0.918	0.821	0.59	0.907	0.883
POCO [4]	0.926	**0.30**	0.950	**0.984**	0.884	0.40	0.928	0.950	0.808	0.61	0.892	0.869
ALTO (Encoder Only)	**0.931**	**0.30**	0.950	0.981	0.889	0.39	0.932	0.951	0.842	0.52	0.908	0.903
ALTO	0.930	**0.30**	**0.952**	0.980	**0.905**	**0.35**	**0.940**	**0.964**	**0.863**	**0.47**	**0.922**	**0.924**

Table 1. **Performance on ShapeNet with various point density levels.** Input noisy point cloud with 3K, 1K and 300 input points from left to right. ALTO is our proposed method and ALTO (Encoder only) is an ablation that uses only our encoder with a non-attention based decoder.

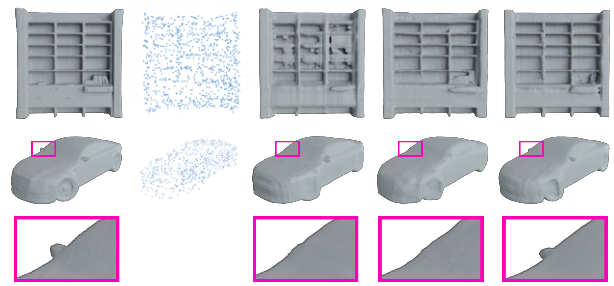
Points are obtained by randomly sampling from each mesh and adding Gaussian noise with zero mean and standard deviation of 0.05.

Scene-Level Datasets: For scene level evaluation, we use Synthetic Rooms dataset [52] and ScanNet-v2 [14]. In total, we use 5000 synthetic room scenes with walls, floors and ShapeNet objects randomly placed together. We use an identical train/val/test split as prepared in prior work [4,52], and Gaussian noise with zero mean and 0.05 standard deviation. ScanNet-v2 contains 1513 scans from real-world scenes that cover a wide range of room types. Since the provided meshes in ScanNet-v2 are not watertight, models are trained on the Synthetic Rooms dataset and tested on ScanNet-v2, which also enables some assessment of Sim2Real performance of various methods.

Evaluation Metrics: The quantitative evaluation metrics used in data tables are standard metrics that enable us to form comparisons to prior work. These include: volumetric IoU, Chamfer-L_1 distance $\times 10^2$, and normal consistency (NC). A detailed definition of each metric can be found in the Occupancy Networks paper [46]. We also include an F-Score metric [68] with threshold value 1%.

Baselines for Comparison: As noted by Boulch et al. [4], baseline methods often perform better in the settings of the original papers. In the same spirit, we thus strictly adapt the protocol of the state-of-the-art (SOTA) paper POCO [4] for evaluation protocol. In addition to POCO, other baselines we include are SPSR [37], ONet [46] and ConvONet [52]. Note that we omit NFK [72] from our evaluations, as they have not made their code publicly available.

Our Method: "Our method" is ALTO. ALTO combines Sec. 3.2 + Sec. 3.3, and is shown in Fig. 5d. Figures/tables use ALTO to denote the proposed method. If we are considering an ablation analysis we will use ALTO with parentheses, e.g., "ALTO (Encoder Only)", and the table caption will specify the ablation. To demonstrate ALTO with different latent topologies , for object-level reconstruction, we use alternation between point and feature triplanes and the resolution of initial individual plane $H = W = 64$. For scene-level reconstruction, we use alternation between point and feature volumes and the resolution $H = W = D = 64$.



(a) GT (b) Input (1K) (c) CONet [52] (d) POCO [4] (e) **ALTO**

Figure 6. **Object-level comparisons on ShapeNet.** On the car, ALTO recovers the detail of having both side mirrors.

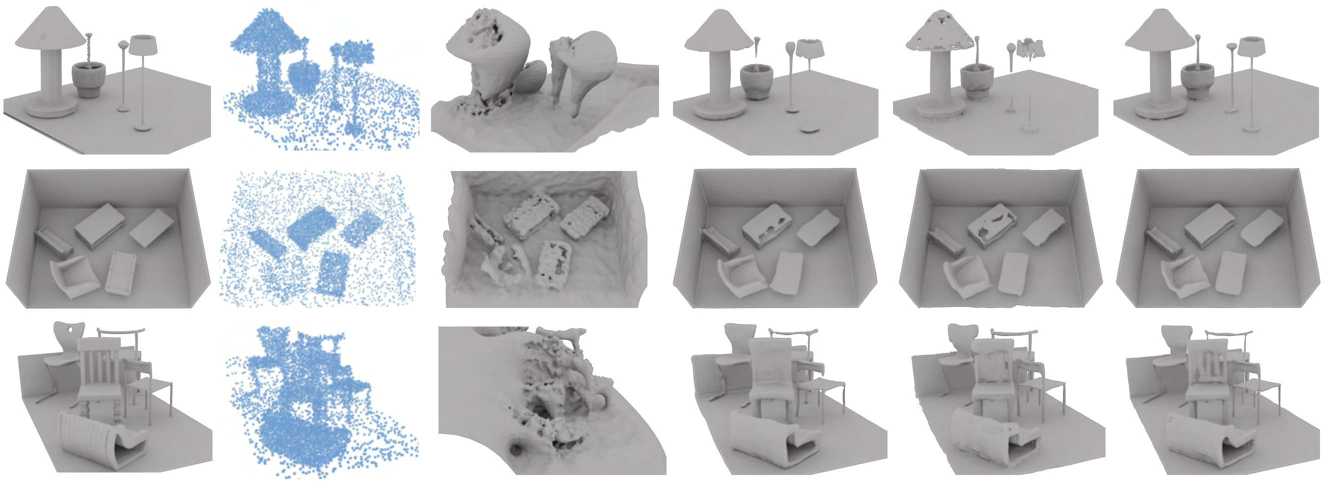
4.2. Results of Object-level Reconstruction

Qualitative Object-level Comparisons: Qualitative results of object-level reconstruction are provided in Fig. 6. We observe that [52] obtains a blurry reconstruction. Note that the width of the bookcase dividers are thicker and there are spurious blobs on the shelves. The SOTA baseline [4] is able to recover some detail, such as the wheel geometry in the car, but loses both side mirrors in the reconstruction. ALTO seems to have a higher fidelity reconstruction due to combining ideas from point and grid latents.

Quantative Object-level Comparisons: ALTO's performance metrics at various point density levels are listed in Tab. 1, for 3K, 1K and 300 input points. When point clouds are sparser, ALTO performs better than POCO on all four metrics. At high point density, ALTO outperforms POCO on three of four metrics. An ablation of just using the ALTO latent encoding and a traditional interpolating decoder [52] is also conducted in the table.

4.3. Scene-level Reconstruction

Qualitative Scene-level Comparisons: ALTO achieves detailed qualitative results compared to baselines. Fig. 7 depicts scene-level reconstruction on the Synthetic Room dataset introduced in [52]. In the first row of Fig. 7 the baselines of ConvONet and POCO both have holes in the coffee table. In the second row of Fig. 7 the high-frequency deatil in the wooden slats of the chair is fully blurred out by Con-



(a) Ground Truth (b) Input Points (c) SPSR [37] (d) ConvONet [52] (e) POCO [4] (f) **ALTO**

Figure 7. **Qualitative comparison on scene-level reconstruction Synthetic Room Dataset.** Learning-based methods are trained and tested on 10K noisy points. ALTO can reconstruct the (top scene) double-deck table and (bottom scene) details in the chair.

Method	IoU ↑	Chamfer-L_1 ↓	NC↑	F-score↑
ONet [46]	0.475	2.03	0.783	0.541
SPSR [37]	-	2.23	0.866	0.810
SPSR trimmed [37]	-	0.69	0.890	0.892
ConvONet [52]	0.849	0.42	0.915	0.964
DP-ConvONet [41]	0.800	0.42	0.912	0.960
POCO [4]	0.884	0.36	0.919	0.980
ALTO	**0.914**	**0.35**	**0.921**	**0.981**

Table 2. **Synthetic Room Dataset.** Input points 10K with noise added. Boldface font represents the preferred results.

Method	IoU ↑	Chamfer-L_1 ↓	NC↑	F-score↑
ConvONet [52]	0.818	0.46	0.906	0.943
POCO [4]	0.801	0.57	0.904	0.812
ALTO	**0.882**	**0.39**	**0.911**	**0.969**

Table 3. **Performance on Synthetic Room Dataset (sparser input point cloud with 3K input points).** Boldface font represents the preferred results.

	N_{Train}=10K, N_{Test}=3K		N_{Train}=N_{Test}=3K	
Method	Chamfer-L_1 ↓	F-score↑	Chamfer-L_1 ↓	F-score↑
ConvONet [52]	1.01	0.719	1.16	0.669
POCO [4]	0.93	0.737	1.15	0.667
ALTO	**0.87**	**0.746**	**0.92**	**0.726**

Table 4. **ScanNet-v2.** We test the generalization capability of all the methods on real-world scans ScanNet using models trained on both the same input points of synthetic dataset as test set (N_{Train}=N_{Test}=3K) and different point density level (N_{Train}=10K, N_{Test}=3K). Boldface font represents the preferred results.

vONet. The advantages of ALTO are even more apparent for fine detail, such as the thin lamp-posts shown in Fig. 1. ALTO reduces the quantization effect due to the grid discretization in the grid form by by using iterative alternation between grid and point latents, encoding more fine-grained local features for conditional occupancy field generation.

Quantitative Scene-level Comparisons: ALTO scores higher on quantitative values for scene-level metrics, shown in Tab. 2 and Tab. 3. In the sparse setting, for the baselines methods, we find that ConvONet [52] is quantitatively superior to the SOTA of POCO [4] because oversmoothing tends to improve quantitative results on noisy point clouds. Nonetheless, ALTO performs better than both baselines because ALTO limits spurious noise without resorting to as much oversmoothing.

4.4. Real-world Scene Generalization

A final experiment in the main paper is to assess the performance of our model in real-world scans from ScanNet-v2 [14]. All models are trained on Synthetic Rooms and tested on ScanNet-v2 to demonstrate generalization capability of our method along with baselines. We demonstrate the qualitative results of the setting where models trained on both the same input points of synthetic dataset as ScanNet test set (N_{Train}=N_{Test}=3K) in Fig. 8. Our method is qualitatively superior to SPSR [37], ConvONet [52], and POCO [4]. As in the Synthetic Rooms dataset, we observe that ConvONet oversmooths surfaces (sometimes causing entire objects to disappear, like the conference table in the purple inset of Fig. 8). In contrast, POCO retains some detail but is noisier. The quantitative results in Tab. 4 are consistent with qualitative results. The cross point density test results (N_{Train}=10k, N_{Test}=3k) also demonstrates the superiority of our method on generalization when there are abundant input points in synthetic dataset used for training and low point density in real-world inference.

5. Discussion and Conclusion

In summary, this paper has adopted a different philosophy from the SOTA in surface detail recovery. We rely nei-

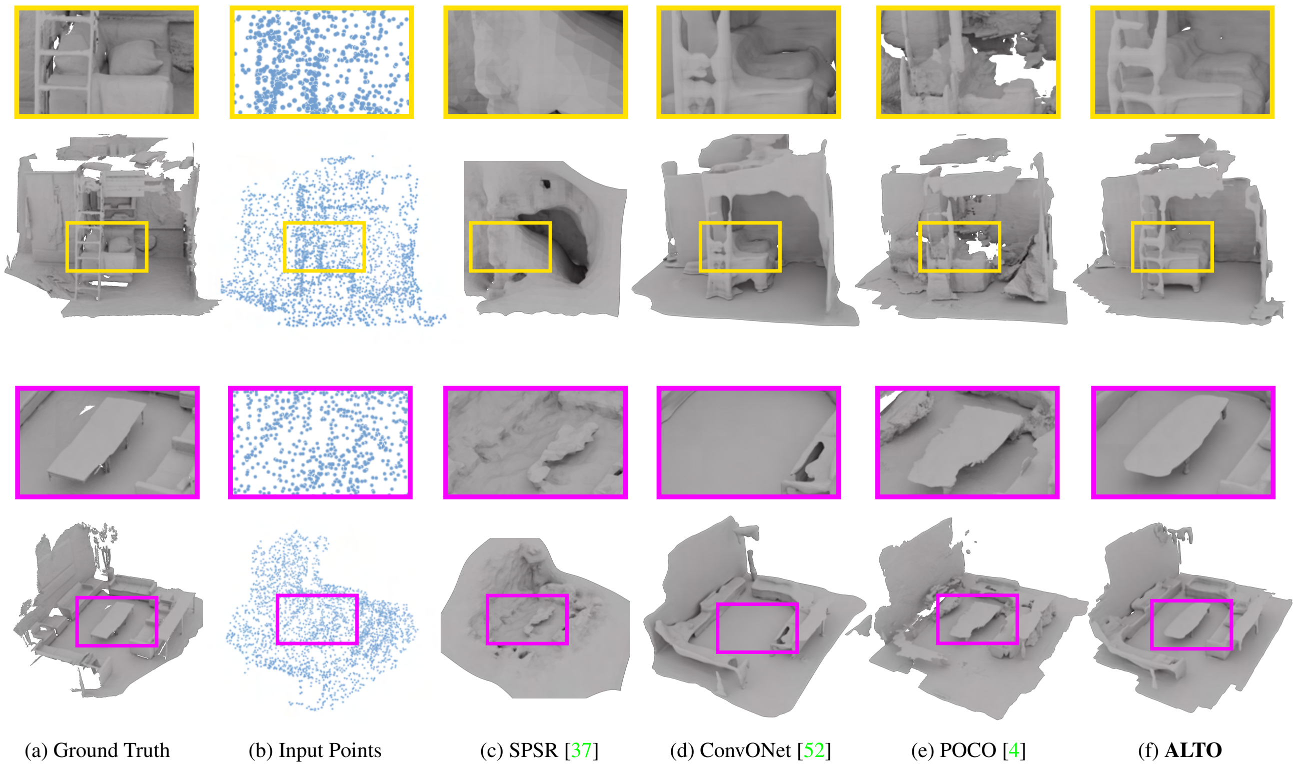


Figure 8. **Cross-dataset evaluation of ALTO and baselines by training on Synthetic Rooms [52] and testing on real-world ScanNet-v2 [14].** Note the large conference-room table is missing in ConvONet [52] (purple inset). The ladder (yellow inset) is a high-frequency surface and we believe our method is qualitatively closest. Please zoom in if browsing with PDF.

Method	# Parameters	Inference time (s)
ConvONet [52]	4,166,657	1.6
POCO [4]	12,790,454	36.1
ALTO	4,787,905	3.6

Table 5. **Runtime comparison.** We report the number of parameters and inference time corresponding to Fig. 8. ALTO is much faster than POCO and recovers more detail [4]. ALTO is also only slightly slower than fast methods that are not as spatially expressive [52].

ther on point latents [4] or grid latents [52] alone, but alternate between topologies. The output of ALTO is a spatially expressive latent that is also topologically easy-to-decode into a 3D surface. This breaks a Pareto tradeoff that previous works have posited between spatial expressiveness and decoding complexity. For this reason, it is not surprising that our method reconstructs more detailed 3D surfaces with faster runtimes than state-of-the-art (Fig. 1 and Tab. 5).

The idea of alternating latent topologies could have implications beyond surface reconstruction. Concurrent research has introduced unusual latent topologies, known as **irregular latents** that show compelling performance benefits for neural fields [85]. One can imagine alternating not only between point, triplane, and voxel latents, but also throwing irregular latents in the mix. Our alternating strategies can be readily applied into other representations such as UDF [11] and 3PSDF [7]. We are also curious to see if alternating topologies can improve performance on a wide range of tasks in neural fields that require spatially expressive latents, such as semantic [70], affordance fields [32] or polarimetric field [53].

Acknowledgements

The authors thank members of the Visual Machines Group for their feedback and support. This project was supported by the US DoD LUCI (Laboratory University Collaboration Initiative) fellowship and partially supported by ARL grants W911NF-20-2-0158 and W911NF-21-2-0104 under the cooperative A2I2 program. D.P. is supported by the Swiss National Science Foundation under grant number P500PT_206946. G.W. is supported by Samsung, Stanford HAI, and a PECASE from the ARO. L.G. is also supported by an ONR Vannevar Bush Faculty Fellowship. A.K. is also supported by a DARPA Young Faculty Award, NSF CAREER Award IIS-2046737, and Army Young Investigator Award.

References

[1] Panos Achlioptas, Olga Diamanti, Ioannis Mitliagkas, and Leonidas J. Guibas. Learning representations and generative models for 3d point clouds. In *Proc. of the International Conf. on Machine learning (ICML)*, 2018. 2

[2] Matan Atzmon and Yaron Lipman. SAL: sign agnostic learning of shapes from raw data. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, pages 2562–2571, 2020. 3

[3] Ayush Bhandari, Achuta Kadambi, and Ramesh Raskar. *Computational Imaging*. 2022. 2

[4] Alexandre Boulch and Renaud Marlet. Poco: Point convolution for surface reconstruction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6302–6314, 2022. 1, 2, 3, 4, 5, 6, 7, 8

[5] André Brock, Theodore Lim, James M. Ritchie, and Nick Weston. Generative and discriminative voxel modeling with convolutional neural networks. *arXiv.org*, 1608.04236, 2016. 2

[6] Angel X Chang, Thomas Funkhouser, Leonidas Guibas, Pat Hanrahan, Qixing Huang, Zimo Li, Silvio Savarese, Manolis Savva, Shuran Song, Hao Su, et al. Shapenet: An information-rich 3d model repository. *arXiv preprint arXiv:1512.03012*, 2015. 5

[7] Weikai Chen, Cheng Lin, Weiyang Li, and Bo Yang. 3psdf: Three-pole signed distance function for learning surfaces with arbitrary topologies. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18522–18531, 2022. 8

[8] Wenzheng Chen, Huan Ling, Jun Gao, Edward Smith, Jaako Lehtinen, Alec Jacobson, and Sanja Fidler. Learning to predict 3d objects with an interpolation-based differentiable renderer. In *Advances in Neural Information Processing Systems (NIPS)*, 2019. 2

[9] Zhiqin Chen and Hao Zhang. Learning implicit fields for generative shape modeling. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2019. 2, 3

[10] Julian Chibane, Thiemo Alldieck, and Gerard Pons-Moll. Implicit functions in feature space for 3d shape reconstruction and completion. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2020. 2, 3

[11] Julian Chibane, Thiemo Alldieck, and Gerard Pons-Moll. Implicit functions in feature space for 3d shape reconstruction and completion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6970–6981, 2020. 8

[12] Christopher Bongsoo Choy, Danfei Xu, JunYoung Gwak, Kevin Chen, and Silvio Savarese. 3d-r2n2: A unified approach for single and multi-view 3d object reconstruction. In *Proc. of the European Conf. on Computer Vision (ECCV)*, 2016. 2

[13] Özgün Çiçek, Ahmed Abdulkadir, Soeren S Lienkamp, Thomas Brox, and Olaf Ronneberger. 3d u-net: learning dense volumetric segmentation from sparse annotation. In *International conference on medical image computing and computer-assisted intervention*, pages 424–432. Springer, 2016. 3

[14] Angela Dai, Angel X. Chang, Manolis Savva, Maciej Halber, Thomas Funkhouser, and Matthias Niessner. Scannet: Richly-annotated 3d reconstructions of indoor scenes. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2017. 6, 7, 8

[15] Angela Dai, Christian Diller, and Matthias Nießner. Sg-nn: Sparse generative neural networks for self-supervised scene completion of rgb-d scans. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 849–858, 2020. 2

[16] Boyang Deng, Kyle Genova, Soroosh Yazdani, Sofien Bouaziz, Geoffrey Hinton, and Andrea Tagliasacchi. Cvxnets: Learnable convex decomposition. *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2020. 2

[17] Zhantao Deng, Jan Bednařík, Mathieu Salzmann, and Pascal Fua. Better patch stitching for parametric surface reconstruction. 2020. 2

[18] Philipp Erler, Paul Guerrero, Stefan Ohrhallinger, Niloy J Mitra, and Michael Wimmer. Points2surf learning implicit surfaces from point clouds. In *European Conference on Computer Vision*, pages 108–124. Springer, 2020. 2

[19] Philipp Erler, Paul Guerrero, Stefan Ohrhallinger, Niloy J. Mitra, and Michael Wimmer. Points2surf learning implicit surfaces from point clouds. In *Proc. of the European Conf. on Computer Vision (ECCV)*, 2020. 3

[20] Haoqiang Fan, Hao Su, and Leonidas J. Guibas. A point set generation network for 3d object reconstruction from a single image. *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2017. 2

[21] Matheus Gadelha, Subhransu Maji, and Rui Wang. 3d shape induction from 2d views of multiple objects. In *Proc. of the International Conf. on 3D Vision (3DV)*, 2017. 2

[22] Jun Gao, Wenzheng Chen, Tommy Xiang, Alec Jacobson, Morgan McGuire, and Sanja Fidler. Learning deformable tetrahedral meshes for 3d reconstruction. In *Advances in Neural Information Processing Systems (NIPS)*, 2020. 2

[23] Simon Giebenhain and Bastian Goldlücke. Air-nets: An attention-based framework for locally conditioned implicit representations. In *2021 International Conference on 3D Vision (3DV)*, pages 1054–1064. IEEE, 2021. 3

[24] Georgia Gkioxari, Jitendra Malik, and Justin Johnson. Mesh R-CNN. In *Proc. of the IEEE International Conf. on Computer Vision (ICCV)*, 2019. 2

[25] Amos Gropp, Lior Yariv, Niv Haim, Matan Atzmon, and Yaron Lipman. Implicit geometric regularization for learning shapes. In *Proc. of the International Conf. on Machine learning (ICML)*, 2020. 3

[26] Thibault Groueix, Matthew Fisher, Vladimir G. Kim, Bryan C. Russell, and Mathieu Aubry. AtlasNet: A papier-mâché approach to learning 3d surface generation. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2018. 2

[27] Christian Häne, Shubham Tulsiani, and Jitendra Malik. Hierarchical surface prediction for 3d object reconstruction. *arXiv.org*, 1704.00710, 2017. 2

[28] Wilfried Hartmann, Silvano Galliani, Michal Havlena, Luc Van Gool, and Konrad Schindler. Learned multi-patch similarity. In *Proc. of the IEEE International Conf. on Computer Vision (ICCV)*, 2017. 2

[29] Xinyu Huang, Xinjing Cheng, Qichuan Geng, Binbin Cao, Dingfu Zhou, Peng Wang, Yuanqing Lin, and Ruigang Yang. The apolloscape dataset for autonomous driving. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2018. 2

[30] Chiyu Jiang, Avneesh Sud, Ameesh Makadia, Jingwei Huang, Matthias Nießner, and Thomas Funkhouser. Local implicit grid representations for 3d scenes. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2020. 3

[31] Li Jiang, Shaoshuai Shi, Xiaojuan Qi, and Jiaya Jia. GAL: geometric adversarial loss for single-view 3d-object reconstruction. In *Proc. of the European Conf. on Computer Vision (ECCV)*, 2018. 2

[32] Zhenyu Jiang, Yifeng Zhu, Maxwell Svetlik, Kuan Fang, and Yuke Zhu. Synergies between affordance and geometry: 6-dof grasp detection via implicit representations. *arXiv preprint arXiv:2104.01542*, 2021. 2, 8

[33] Achuta Kadambi, Vage Taamazyan, Boxin Shi, and Ramesh Raskar. Polarized 3d: High-quality depth sensing with polarization cues. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 3370–3378, 2015. 2

[34] Achuta Kadambi, Vage Taamazyan, Boxin Shi, and Ramesh Raskar. Depth sensing using geometrically constrained polarization normals. *International Journal of Computer Vision*, 125:34–51, 2017. 2

[35] Angjoo Kanazawa, Shubham Tulsiani, Alexei A. Efros, and Jitendra Malik. Learning category-specific mesh reconstruction from image collections. In *Proc. of the European Conf. on Computer Vision (ECCV)*, 2018. 2

[36] Abhishek Kar, Christian Häne, and Jitendra Malik. Learning a multi-view stereo machine. In *Advances in Neural Information Processing Systems (NIPS)*, 2017. 2

[37] Michael Kazhdan and Hugues Hoppe. Screened poisson surface reconstruction. *ACM Transactions on Graphics (ToG)*, 32(3):1–13, 2013. 6, 7, 8

[38] Michael M. Kazhdan, Matthew Bolitho, and Hugues Hoppe. Poisson surface reconstruction. In *Proceedings of the Fourth Eurographics Symposium on Geometry Processing, Cagliari, Sardinia, Italy, June 26-28, 2006*, volume 256, pages 61–70, 2006. 2

[39] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014. 5

[40] Yiyi Liao, Simon Donne, and Andreas Geiger. Deep marching cubes: Learning explicit surface representations. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2018. 2

[41] Stefan Lionar, Daniil Emtsev, Dusan Svilarkovic, and Songyou Peng. Dynamic plane convolutional occupancy networks. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 1829–1838, 2021. 1, 3, 4, 7

[42] Tian Yu Liu, Parth Agrawal, Allison Chen, Byung-Woo Hong, and Alex Wong. Monitored distillation for positive congruent depth completion. In *Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part II*, pages 35–53. Springer, 2022. 2

[43] William E Lorensen and Harvey E Cline. Marching cubes: A high resolution 3d surface construction algorithm. *ACM siggraph computer graphics*, 21(4):163–169, 1987. 5

[44] Qianli Ma, Shunsuke Saito, Jinlong Yang, Siyu Tang, and Michael J. Black. SCALE: Modeling clothed humans with a surface codec of articulated local elements. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2021. 2

[45] Daniel Maturana and Sebastian Scherer. Voxnet: A 3d convolutional neural network for real-time object recognition. In *Proc. IEEE International Conf. on Intelligent Robots and Systems (IROS)*, 2015. 2

[46] Lars Mescheder, Michael Oechsle, Michael Niemeyer, Sebastian Nowozin, and Andreas Geiger. Occupancy networks: Learning 3d reconstruction in function space. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 4460–4470, 2019. 2, 3, 6, 7

[47] Mateusz Michalkiewicz, Jhony K Pontes, Dominic Jack, Mahsa Baktashmotlagh, and Anders Eriksson. Implicit surface representations as layers in neural networks. In *Proc. of the IEEE International Conf. on Computer Vision (ICCV)*, 2019. 3

[48] Junyi Pan, Xiaoguang Han, Weikai Chen, Jiapeng Tang, and Kui Jia. Deep mesh reconstruction from single RGB images via topology modification networks. In *Proc. of the IEEE International Conf. on Computer Vision (ICCV)*, 2019. 2

[49] Jeong Joon Park, Peter Florence, Julian Straub, Richard A. Newcombe, and Steven Lovegrove. Deepsdf: Learning continuous signed distance functions for shape representation. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2019. 2, 3

[50] Despoina Paschalidou, Ali Osman Ulusoy, Carolin Schmitt, Luc van Gool, and Andreas Geiger. Raynet: Learning volumetric 3d reconstruction with ray potentials. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2018. 2

[51] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. Pytorch: An imperative style, high-performance deep learning library. *Advances in neural information processing systems*, 32, 2019. 5

[52] Songyou Peng, Michael Niemeyer, Lars Mescheder, Marc Pollefeys, and Andreas Geiger. Convolutional occupancy networks. In *European Conference on Computer Vision*, pages 523–540. Springer, 2020. 1, 2, 3, 4, 5, 6, 7, 8

[53] Henry Peters, Yunhao Ba, and Achuta Kadambi. pcon: Polarimetric coordinate networks for neural scene representations. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023. 8

[54] Charles R Qi, Hao Su, Kaichun Mo, and Leonidas J Guibas. Pointnet: Deep learning on point sets for 3d classification

and segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 652–660, 2017. 3

[55] Charles R Qi, Li Yi, Hao Su, and Leonidas J Guibas. Pointnet++: Deep hierarchical feature learning on point sets in a metric space. In *Advances in Neural Information Processing Systems (NIPS)*, 2017. 2

[56] Danilo Jimenez Rezende, S. M. Ali Eslami, Shakir Mohamed, Peter Battaglia, Max Jaderberg, and Nicolas Heess. Unsupervised learning of 3d structure from images. In *Advances in Neural Information Processing Systems (NIPS)*, 2016. 2

[57] Gernot Riegler, Ali Osman Ulusoy, Horst Bischof, and Andreas Geiger. OctNetFusion: Learning depth fusion from data. In *Proc. of the International Conf. on 3D Vision (3DV)*, 2017. 2

[58] Gernot Riegler, Ali Osman Ulusoy, and Andreas Geiger. Octnet: Learning deep 3d representations at high resolutions. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2017. 2

[59] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*, pages 234–241. Springer, 2015. 3

[60] Shunsuke Saito, Zeng Huang, Ryota Natsume, Shigeo Morishima, Angjoo Kanazawa, and Hao Li. Pifu: Pixel-aligned implicit function for high-resolution clothed human digitization. In *Proc. of the IEEE International Conf. on Computer Vision (ICCV)*, 2019. 3

[61] Shunsuke Saito, Tomas Simon, Jason M. Saragih, and Hanbyul Joo. Pifuhd: Multi-level pixel-aligned implicit function for high-resolution 3d human digitization. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2020. 3

[62] Vincent Sitzmann, Julien N.P. Martel, Alexander W. Bergman, David B. Lindell, and Gordon Wetzstein. Implicit neural representations with periodic activation functions. In *Proc. NeurIPS*, 2020.

[63] Vincent Sitzmann, Michael Zollhöfer, and Gordon Wetzstein. Scene representation networks: Continuous 3d-structure-aware neural scene representations. In *Advances in Neural Information Processing Systems (NIPS)*, 2019. 3

[64] David Stutz and Andreas Geiger. Learning 3d shape completion from laser scan data with weak supervision. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2018. 2

[65] Towaki Takikawa, Joey Litalien, Kangxue Yin, Karsten Kreis, Charles T. Loop, Derek Nowrouzezahrai, Alec Jacobson, Morgan McGuire, and Sanja Fidler. Neural geometric level of detail: Real-time rendering with implicit 3d shapes. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2021. 3

[66] Jiapeng Tang, Jiabao Lei, Dan Xu, Feiying Ma, Kui Jia, and Lei Zhang. Sa-convonet: Sign-agnostic optimization of convolutional occupancy networks. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 6504–6513, 2021. 2, 3

[67] M. Tatarchenko, A. Dosovitskiy, and T. Brox. Octree generating networks: Efficient convolutional architectures for high-resolution 3d outputs. In *Proc. of the IEEE International Conf. on Computer Vision (ICCV)*, 2017. 2

[68] Maxim Tatarchenko, Stephan R Richter, René Ranftl, Zhuwen Li, Vladlen Koltun, and Thomas Brox. What do single-view 3d reconstruction networks learn? In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 3405–3414, 2019. 6

[69] Hugues Thomas, Charles R. Qi, Jean-Emmanuel Deschaud, Beatriz Marcotegui, François Goulette, and Leonidas J. Guibas. Kpconv: Flexible and deformable convolution for point clouds. In *Proc. of the IEEE International Conf. on Computer Vision (ICCV)*, 2019. 2

[70] Suhani Vora, Noha Radwan, Klaus Greff, Henning Meyer, Kyle Genova, Mehdi SM Sajjadi, Etienne Pot, Andrea Tagliasacchi, and Daniel Duckworth. Nesf: Neural semantic fields for generalizable semantic segmentation of 3d scenes. *arXiv preprint arXiv:2111.13260*, 2021. 2, 8

[71] Nanyang Wang, Yinda Zhang, Zhuwen Li, Yanwei Fu, Wei Liu, and Yu-Gang Jiang. Pixel2mesh: Generating 3d mesh models from single rgb images. In *Proc. of the European Conf. on Computer Vision (ECCV)*, 2018. 2

[72] Francis Williams, Zan Gojcic, Sameh Khamis, Denis Zorin, Joan Bruna, Sanja Fidler, and Or Litany. Neural fields as learnable kernels for 3d reconstruction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18500–18510, 2022. 3, 6

[73] Alex Wong, Safa Cicek, and Stefano Soatto. Learning topology from synthetic data for unsupervised depth completion. *IEEE Robotics and Automation Letters*, 6(2), 2021. 2

[74] Alex Wong, Xiaohan Fei, Stephanie Tsuei, and Stefano Soatto. Unsupervised depth completion from visual inertial odometry. *IEEE Robotics and Automation Letters*, 5(2), 2020. 2

[75] Alex Wong and Stefano Soatto. Unsupervised depth completion with calibrated backprojection layers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 12747–12756, 2021. 2

[76] Jiajun Wu, Ilker Yildirim, Joseph J. Lim, Bill Freeman, and Joshua B. Tenenbaum. Galileo: Perceiving physical object properties by integrating a physics engine with deep learning. In *Advances in Neural Information Processing Systems (NIPS)*, 2015. 2

[77] Jiajun Wu, Chengkai Zhang, Tianfan Xue, Bill Freeman, and Josh Tenenbaum. Learning a probabilistic latent space of object shapes via 3d generative-adversarial modeling. In *Advances in Neural Information Processing Systems (NIPS)*, 2016. 2

[78] Haozhe Xie, Hongxun Yao, Xiaoshuai Sun, Shangchen Zhou, and Shengping Zhang. Pix2vox: Context-aware 3d reconstruction from single and multi-view images. In *Proc. of the IEEE International Conf. on Computer Vision (ICCV)*, 2019. 2

[79] Qiangeng Xu, Weiyue Wang, Duygu Ceylan, Radomír Mech, and Ulrich Neumann. DISN: deep implicit surface network for high-quality single-view 3d reconstruction. In

Advances in Neural Information Processing Systems (NIPS), 2019. 3

[80] Guandao Yang, Xun Huang, Zekun Hao, Ming-Yu Liu, Serge J. Belongie, and Bharath Hariharan. Pointflow: 3d point cloud generation with continuous normalizing flows. In *Proc. of the IEEE International Conf. on Computer Vision (ICCV)*, 2019. 2

[81] Yanchao Yang, Alex Wong, and Stefano Soatto. Dense depth posterior (ddp) from single image and sparse range. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019. 2

[82] Yao Yao, Zixin Luo, Shiwei Li, Tian Fang, and Long Quan. Mvsnet: Depth inference for unstructured multi-view stereo. *Proc. of the European Conf. on Computer Vision (ECCV)*, 2018. 2

[83] Yao Yao, Zixin Luo, Shiwei Li, Tianwei Shen, Tian Fang, and Long Quan. Recurrent mvsnet for high-resolution multi-view stereo depth inference. *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2019. 2

[84] Zehao Yu and Shenghua Gao. Fast-mvsnet: Sparse-to-dense multi-view stereo with learned propagation and gauss-newton refinement. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2020. 2

[85] Biao Zhang, Matthias Nießner, and Peter Wonka. 3dilg: Irregular latent grids for 3d generative modeling. *arXiv preprint arXiv:2205.13914*, 2022. 8

[86] Hengshuang Zhao, Li Jiang, Jiaya Jia, Philip HS Torr, and Vladlen Koltun. Point transformer. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 16259–16268, 2021. 5