ARO-Net: Learning Implicit Fields from Anchored Radial Observations

Yizhi Wang[1,2]*, Zeyu Huang[1]*, Ariel Shamir[3], Hui Huang[1], Hao Zhang[2], Ruizhen Hu[1]†

[1]Shenzhen University [2]Simon Fraser University [3]Reichman University

Abstract

We introduce anchored radial observations *(ARO), a novel shape encoding for learning implicit field representation of 3D shapes that is* category-agnostic *and* generalizable *amid significant shape variations. The main idea behind our work is to reason about shapes through partial observations from a set of viewpoints, called anchors. We develop a general and unified shape representation by employing a fixed set of anchors, via Fibonacci sampling, and designing a coordinate-based deep neural network to predict the occupancy value of a query point in space. Differently from prior neural implicit models that use global shape feature, our shape encoder operates on* contextual, query-specific *features. To predict point occupancy, locally observed shape information from the perspective of the anchors surrounding the input query point are encoded and aggregated through an attention module, before implicit decoding is performed. We demonstrate the quality and generality of our network, coined ARO-Net, on surface reconstruction from sparse point clouds, with tests on novel and unseen object categories, "one-shape" training, and comparisons to state-of-the-art neural and classical methods for reconstruction and tessellation.*

1. Introduction

Despite the substantial progress made in deep learning in recent years, transferability and generalizability issues still persist due to domain shifts and the need to handle diverse, out-of-distribution test cases. For 3D shape representation learning, a reoccurring challenge has been the inability of trained neural models to generalize to unseen object categories and diverse shape structures, as these models often overfit to or "memorize" the training data.

In this paper, we introduce a novel shape encoding for learning an implicit field [34] representation of 3D shapes that is *category-agnostic* and *generalizable* amid significant shape variations. In Figure 1 (top), we show 3D reconstructions from sparse point clouds obtained by our approach that is trained on chairs but tested on airplanes, riffles, and
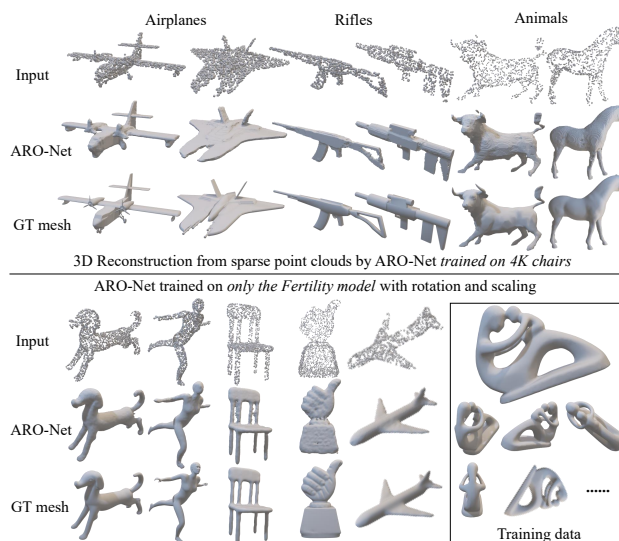

Figure 1. Neural 3D reconstruction using ARO-Net from sparse point clouds (1,024 or 2,048 points). Top: reconstruction results on airplanes, riffles, and animals when the network was trained only on chairs. Bottom: reconstruction of a variety of shapes when the network was train on numerous versions of one model - the Fertility. See comparison to other methods in Section 4.

animals. Our model can even reconstruct a variety of shapes when the training data consists of *only one* shape, augmented with rotation and scaling; see Figure 1 (bottom).

The main idea behind our work is to reason about shapes from *partial observations* at a set of viewpoints, called *anchors*, and apply this reasoning to learn implicit fields. We develop a general and unified shape representation by designating a *fixed* set of anchors and designing a coordinate-based neural network to predict the occupancy at a query point. In contrast to classical neural implicit models such as IM-Net [8], OccNet [23], and DeepSDF [24], which learn global shape features for occupancy/distance prediction, our novel encoding scheme operates on *local* shape features obtained by viewing the query point from the anchors.

Specifically, for a given query point x, we collect locally observable shape information surrounding x, as well as directional and distance information toward x, from the perspective of the set of anchors, and encode such information using

---

*Equal contribution

†Corresponding author. E-mail: ruizhen.hu@gmail.com

r_1		O_1			f_1	r_1	$\|r_1\|$		
	a_1				...	...	...		
r_1	x	O_1		PointNet	...	...	...	Attention Module	Implicit Decoder
x					...	...	...		
r_m	O_m				...	...	...		
	a_m	O_m			f_m	r_m	$\|r_m\|$		
r_m									

(a) Anchors and input shape   (b) ARO fixated on query   (c) ARO-Net
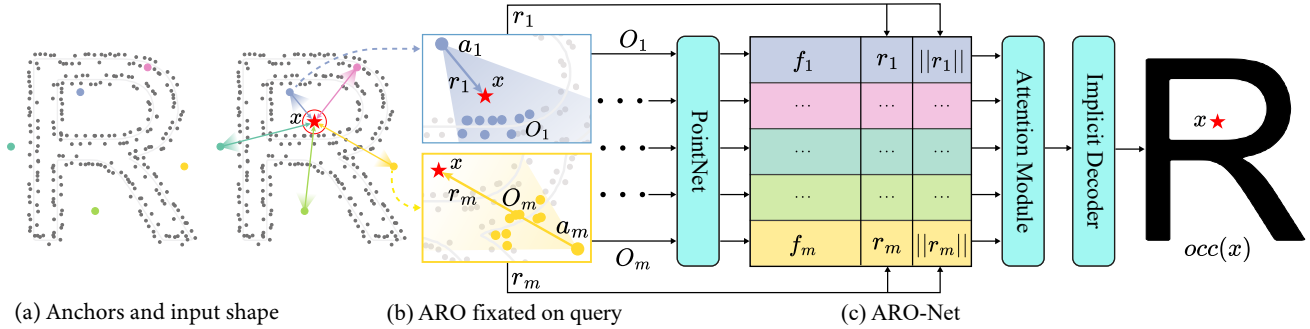
x ★ R $occ(x)$

Figure 2. 2D illustration of ARO and ARO-Net architecture: (a) Input point cloud (in grey) and a set of m fixed anchors (coloured dots). (b) Radial observation O_i from each anchor a_i toward the query point x consists of closed points inside the cone apexed at a_i, with axis $r_i = x - a_i$. (c) Each radial observation O_i is passed to a PointNet encoder to obtain an embedding feature f_i, which is concatenated with r_i and its norm to form the query-specific ARO encoding of x with respect to a_i. Finally, all the ARO features are decoded into the occupancy value $occ(x)$ though an attention module and several MLPs. For 3D reconstruction, the ARO features are computed for each query point, while the PointNet, attention module, and implicit decoder are fixed during inference – their weights were determined during training.

(a) Input   (b) OccNet   (c) ConvONet   (d) ARO-Net   (e) GT

Figure 3. A somewhat extreme toy example comparing ARO-Net to prior occupancy prediction networks on 3D reconstruction from a sparse point cloud of a cube (a), with training on a single sphere. The results from OccNet [23] and ConvONet) [25] show more signs of overfitting to the training sphere than ARO-Net (d).

PointNet [27]; see Figure 2(b). The PointNet features are then aggregated through an attention module, whose output is fed to an implicit decoder to produce the occupancy value for x. We call our *query-specific* shape encoding *Anchored Radial Observations*, or *ARO*. The prediction network is coined ARO-Net, as illustrated in Figure 2.

The advantages of ARO for learning implicit fields are three-fold. First, shape inference from partial and local observations is not bound by barriers set by object categories or structural variations. Indeed, local shape features are more prevalent, and hence more generalizable, across categories than global ones [8, 23, 24]. Second, ARO is query-specific and the directional and distance information it includes is intimately tied to the occupancy prediction at the query point, as explained in Section 3.1. Last but not least, by aggregating observations from all the anchors, the resulting encoding is not purely local, like the voxel-level encoding or latent codes designed to better capture geometric details across large-scale scenes [3, 15, 25]. ARO effectively captures more global and *contextual* query-specific shape features.

In Figure 3, we demonstrate using a toy example the difference between ARO-Net and representative neural implicit models based on global (OccNet [23]) and local grid shape encodings (convolutional occupancy network or ConvONet [25]), for the 3D reconstruction task. All three networks were trained on a single sphere shape, with a single anchor for ARO inside the sphere. When tested on reconstructing a cube, the results show that both OccNet and ConvONet exhibit more memorization of the training sphere either globally or locally, while the ARO-Net reconstruction is more faithful to the input without special fine-tuning.

In our current implementation of ARO-Net, we adopt Fibonacci sampling [19] to obtain the fixed set of anchors. We demonstrate the quality and generalizability of our method on surface reconstruction from sparse point clouds, with testing on novel and unseen object categories, as well as "one-shape" training (see bottom of Figure 1). We report extensive quantitative and qualitative comparison results to state-of-the-art methods including both neural models for reconstruction [8, 11, 23, 25, 26] and tessellation [7], as well as classical schemes such as screen Poisson reconstruction [18]. Finally, we conduct ablation studies to evaluate our choices for the number of anchors, the selection strategies, and the decoder architecture: MLP vs. attention modules.

2. Related work

Learning neural fields is an emerging and one of the most intensely studied subjects in the vision and machine learning communities in recent years, while surface reconstruction is one of the most classical problems in computer vision and graphics. In this section, we shall only focus on the most closely related prior works. For a more comprehensive coverage on the relevant topics, we refer the readers to the surveys on surface reconstruction from point clouds [1] and neural fields [34], as well as the the excellent summary on neural implicit representations by Vincent Sitzmann.

Surface reconstruction and local priors. Classical reconstruction schemes such as Screened Poisson (SPR) [18] often produce artifacts amid noise and under sampling. One remedy is to rely on *local shape priors*, which are reoccurring patch templates that can be matched to help reconstruction over sparse input. Gal et al. [12] develop such an example-based surface reconstruction method, relying on a pre-built set of enriched patches sampled from a database of 3D models that include surface normals and point feature classification (e.g., edge or corner) to facilitate reconstruction. These patches essentially serve as a training set.

Taking local templates to the extreme gives us classical tessellation methods such as Marching Cubes [22] and Dual Countouring [16], whose "neural" versions have been developed recently as neural Marching Cubes [9] and neural Dual Contouring (NDC) [7]. In particular, the unsigned version of NDC, or UNDC, can achieve state-of-the-art results on reconstruction from un-oriented point clouds with sufficient sampling. Indeed, what is common about all of these purely local reconstruction schemes is their ability to generalize across object categories, while disregarding any contextual shape information and global reconstruction priors.

Neural implicit models. Since OccNet [23], IM-Net [8], and DeepSDF [24] concurrently introduced the coordinate-based implicit neural representation in 2019, research and development on neural implicit fields [34] have flourished. In these early works, the shape encoder [8, 23] and latent code [24] employed for occupancy/distance predictions only model global shape features. The resulting learned representations were unable to reconstruct fine geometric details.

The next wave of developments has focused on conditioning implicit neural representations on local features stored in voxel [3, 10, 15, 25] or image grids [21, 28, 35] to more effectively recover geometric or topological details and to scale to scene reconstruction, or at the patch level [13, 32] to improve generalizability across object categories.

Some of these locally conditioned neural implicit models essentially learn local shape priors [12] from training shapes, where the learned priors are parameterized in different ways, e.g., voxel-level latent codes in Deep Local Shape [3], embeddings of ShapeNet [5] parts in local implicit grid representations [15], structured Gaussians in Local Deep Implicit Functions (LDIF) [13], and canonical patches in PatchNets [32]. Other works [10, 11, 25], like ARO-Net, also utilize query-specific shape features.

Implicit field from query-specific features. For occupancy prediction at a query point p, ConvONet [25] extracts convolutional features at p by interpolating over a feature field defined for an input shape. IF-Net [10] also extracts a feature at p but its associated feature grid encodes multi-scale features representing both global and local shape information. Given an input point cloud, Points2Surf [11] samples a

local point patch close to the query p, as well as a global point sub-sampling emanating from p, and encodes them into two features vectors for implicit decoding. More recently, AIR-Net [14], POCO [2], 3DLIG [36], and SA-ConvONet [31], improve over ConvONet with irregular grids, attention mechanisms, and sign-agnostic optimizations.

Compared to all these query-specific encodings, ARO captures shape information around the query p in a more contextual and structure-aware manner. In particular, while Points2Surf provides a "uni-directional" observation from p, the local observations by ARO are radially distributed around p, providing a more global context. In addition, the inclusion of distance and directional information in ARO reveals a direction connection to occupancy prediction, as explained in Section 3.1. Owing to these distinctive properties of ARO, ARO-Net consistently produces higher-quality and more faithful reconstructions, as demonstrated in Section 4.

Basis point set. Our view-based ARO encoding is related to basis point set (BPS) [26] for point cloud encoding. BPS is a fixed-length feature vector computed as the minimal distances from a fixed set of anchors to a given point cloud. BPS can serve as a global shape feature, e.g., in place of ARO, for neural implicit reconstruction from point clouds.

3. Method

In this section, we describe and analyze our novel query-specific shape encoding, ARO, in detail. We first show in Section 3.1 that when a given set of anchors can provide a full coverage of a shape, i.e., it observe the entire shape surface, then point occupancies around the shape surface can be fully determined from the corresponding ARO representation. More generally however, the full coverage condition may not be satisfied or in the case of 3D reconstruction from a noisy and sparse point cloud, the partial observations at the anchors may be inaccurate. In such cases, we train a neural network to predict the point occupancies. This network learns a mapping from the anchored observations to the occupancy value at any spatial query point. We present this network, ARO-Net, in Section 3.2.

3.1. Anchor visibility and point occupancy

Given an anchor point a, we can cast rays from a in all directions to obtain its visible region relative to a given watertight surface S. A ray that does not intersect S can be clipped by the bounding box of the S. In previous works, such visible regions have been adopted for points inside the shape to assist in shape partition, i.e., the Shape Diameter Function [30], as well as for points outside the shape for reconstruction [29], or to guide relationship optimization between two shapes [37]. These works have shown that the visible regions can not only characterize local shape regions, but also provide a global and contextual view of the shape

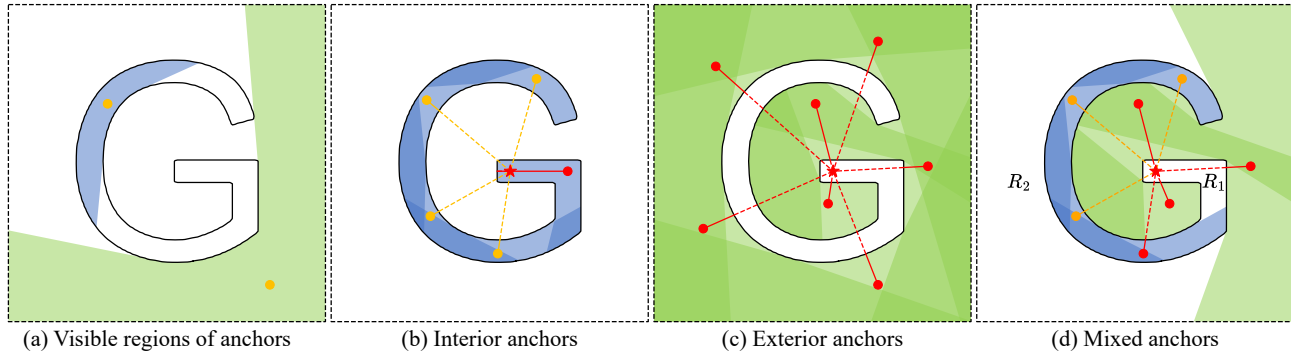| (a) Visible regions of anchors | (b) Interior anchors | (c) Exterior anchors | (d) Mixed anchors |

Figure 4. Anchor visibility vs. point occupancy. The anchors are shown in yellow dots, and the ones that are used to determine the occupancy of the query point (red star) are turn into red. (a) The visible regions of different anchors relative to the given shape 'G', colored in blue for interior anchors and green for exterior anchors. (b) A set of interior anchors that fully cover the surface, the inside region of the shape is exactly the union of the visible regions of all the anchors. (c) For a set of exterior anchors that fully cover the surface, the region outside the shape is exactly the union of the visible regions of all the anchors. (d) For a set of mixed anchors that together cover the surface, apart from the visible regions, the occupancy of uncovered regions, i.e., R_1 and R_2, can also be determined. R_1 bounded by the surface and the visibility boundary is totally inside the shape, while R_2 with dashed virtual boundary is totally outside the shape.

structure. Figure 4 (a) shows the visible regions of one anchor inside the shape (blue) and the other anchor outside the shape (green).

Our key observation to guide the design of our ARO representation is that if we have a set of well-distributed anchors, then combining their visible regions together can fully characterize the entire shape. We consider a set of anchors to be well-distributed if any surface point of the shape can be viewed by at least one anchor. In the following, we discuss different cases of well-distributed anchors and how they define the shape; see illustrations in Figure 4(b-d).

When the set of well-distributed anchors is all inside the shape as shown in Figure 4(b), then the union of all the visible regions of the anchors is identical to the interior region of the shape. Thus, a query point, the red star in Figure 4(b), is inside the shape if and only if it is *covered* by at least one radial observation, which we call a *covering anchor*, i.e., the red point in Figure 4(b). This is true if the distance between the query point and the covering anchor is smaller than the radial distance of this anchor along the ray direction pointing to the query point.

When the set of well-distributed anchors is all outside the shape as shown in Figure 4(c), the conclusion is similar for the points outside the shape. Thus, the union of all the visible regions of the anchors is identical to the region exterior to the shape. Therefore, a query point, i.e., the red star in Figure 4(c), is inside the shape if and only if it is *not covered by any* of the visible regions. That is, its distance to each anchor is always larger than the radial distance of this anchor along the ray direction pointing to the query point. Thus, all the anchors together determine the occupancy of the query point.

When the anchor locations are mixed, with some inside and some outside the shape as shown in Figure 4(d), the

uncovered spatial regions, R_1 and R_2, could be either inside or outside the shape. If the region is bounded by the shape boundary and the visible region boundaries for the anchors, such as R_1, then the entire region is inside the shape. Locally, this is similar to cases shown in (c) with those red anchors fully covering its boundary. As for R_2, the region is partially bounded by the virtual bounding box, i.e., for the image/volume capture, then it is totally outside the shape.

To summarize, if we have a set of anchors that together observe the entire surface of the shape, then their visible regions together can fully determine the inside and outside regions of the shape. Moreover, one important property is that it converts the inside/outside judgment of a spatial point relative to the shape, to the point's relationship to the radial observations of the anchors, which leads to our definition of query-specific ARO. This property allows to design a network than predicts the occupancy of a spatial point based on *local features* captured by ARO when the anchors are not well-distributed and the shape is only partially observed.

3.2. ARO-Net for occupancy prediction

Given a point cloud P, our goal is to reconstruct the corresponding watertight surface S based on our ARO representation for point clouds. We choose to use an implicit representation of the surface, which is essentially a function determining whether a given query point x in space is inside or outside the surface S. Unlike previous methods that encode the entire point cloud as a single global latent code [8, 23], we utilize the set of anchors to identify local surface regions that are essential to classify occupancy. This allows us to implement a *category-agnostic* method to reconstruct any shape. Figure 2 shows how query-specific ARO is extracted and utilized by ARO-Net, which outputs the
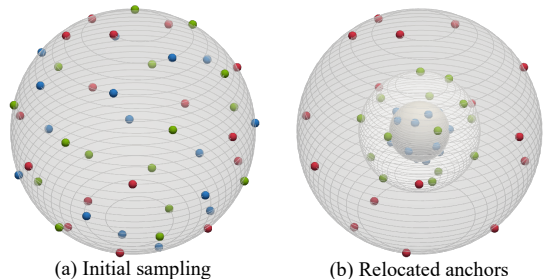
(a) Initial sampling   (b) Relocated anchors

Figure 5. Anchor placement.

occupancy for any given query point in space.

The input of our method consists of a point cloud $P = \{p_1, \ldots, p_n\}$ and an anchor set $A = \{a_1, \ldots, a_m\}$ as shown in Figure 2 (a), where the point cloud are drawn in gray and anchors are drawn with different colors. Given a query point x shown as the red dot in Figure 2 (b), for each anchor a_i, we can compute the relative position of x to a_i as $r_i = x - a_i$. Next, we create a cone with apex at a_i and r_i as the axis to get the nearest k points $O_i = \{p_i^j\}_{j=1,\ldots,k}$ from P, which is defined as the radial observation of a_i relative to the query point x and referred as ARO. Each ARO O_i is then passed to the PointNet encoder to get the corresponding embedding feature f_i. The relative location r_i and its norm $\|r_i\|$ are further concatenated to get a local feature representative of query x relative to anchor a_i. All the local features relative to all the anchors are then passed to an attention module and finally decoded into an occupancy value.

Anchors placement. A good set of anchor points $A = \{a_1, a_2, \ldots, a_m\}$ must be located in different positions in the space bounding the shape, so that they will observe more of the shape's surface with high probability. Hence, to build our set, we sample a set of anchors on three spheres with different radii around the shape. In more detail, assuming that the input point cloud is normalized inside a unit sphere, we first uniformly sample m points on the surface of the unit sphere with radius $r = 1/2$ using Fibonacci Sampling [19], then for the points with index $i\%3 = 1$ we move the point towards the center to position them on the sphere with radius $r = 1/4$, and for the points with index $i\%3 = 2$ we further move the point to the sphere with radius $r = 1/8$, as shown in Figure 5. This creates a set of anchors distributed in space that can well observe any shape within the sphere.

4. Results and evaluation

4.1. Experiment setting

Implementation details. We employ a binary cross entropy loss for predicted occupancies and an Adam optimizer whose initial learning rate is 3e-4, decaying by 0.5 every 100 epochs. The conical angle is set to be $24°$ and $k = 16$

Method	LFD↓	HD↓	CD↓	EMD↓	IOU↑
IM-Net [8]	3.27	11.96	57.00	11.40	3.63
OccNet [23]	2.49	11.95	54.68	11.60	3.51
BPS [26]	2.64	5.03	11.72	2.66	6.56
ConvONet [25]	2.69	3.87	8.43	1.71	6.78
Points2Surf [11]	1.64	<u>2.75</u>	5.69	1.25	<u>8.36</u>
UNDC [7]	**1.25**	2.78	**4.90**	<u>1.17</u>	8.21
ARO-Net	<u>1.35</u>	**2.25**	<u>5.46</u>	**1.12**	**8.79**

Table 1. Quantitative comparisons to state-of-the-art methods on the ABC dataset. Numbers in bold represent first ranking and those underlined in second ranking, in each column. Note that the original version of IM-Net does not take point clouds as input, thus we replace its encoder with OccNet's PointNet encoder.

in all our experiments. The attention module is a Transformer [33] encoder consisting of 6 encoding layers and 8 attention heads. The implicit decoder is a fully connected layer which outputs the occupancy value. We use the sample PointNet architecture as Occ-Net [23]. ARO-Net receives 2,048 points as input when training, and can handle arbitrary point numbers when testing. We tested 3 sets of point numbers $n = \{512, 1024, 2048\}$.

Dataset. Our experiments are conducted on the "Big CAD dataset", ABC [20], and ShapeNet V1 [4]. For ABC, we use the same train/test split as NDC [7], with 4,011 shapes for training and 982 shapes for testing. For ShapeNet V1, we train ARO-Net using 4K chairs and test 100 shapes per category. Note that ABC is an especially challenging dataset for 3D representation learning since there is no clear notion of object category (no category labels at all), while the shapes exhibit significant geometry and topology variations.

Evaluation metrics. We adopt Chamfer distance (CD), Hausdoff distance (HD), earth mover's distance (EMD), occupancy intersection over union (IOU) proposed in [23], and light filed distance (LFD) [6] as the evaluation metrics for reconstructed meshes. Note that since the IOU computation method provided by [23] requires watertight shapes, we report IOU only on the ABC dataset.

4.2. Comparisons to state-of-the-art methods

Reconstruction on ABC. Quantitative comparisons to state-of-the-art neural 3D reconstruction models on ABC are shown in Table. 1. We can see that those methods using only global shape features, i.e., Occ-Net [23], IM-Net [8], and BPS [26], are consistently outperformed by UNDC, which learns local tessellation, and ConvONet, Points2Surf, and ARO-Net, which all encode local and query-specific features. Among the last four methods, ARO-Net exhibits a clear advantage in Hausdoff distance and OCC-IOU, and produces the overall best performance by ranking the first in three out of five metrics (IOU, EMD and HD) and rank-
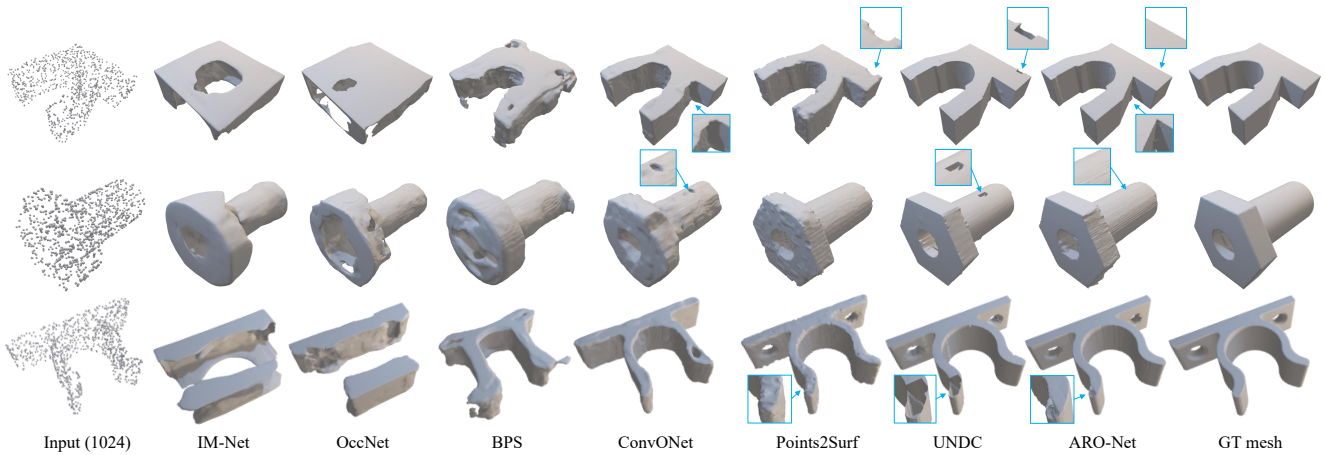
Figure 6. Visual comparisons to state-of-the-art methods on ABC. Zoom-ins highlight reconstruction artifacts to contrast with ARO-Net.

ing the second in one metric (LFD). For the remaining CD metric, ARO-Net is still a close runner-up.

Some visual comparisons are shown in Figure 6. As shape variations in ABC is large, Occ-Net [23] and IM-Net [8] cannot faithfully reconstruct global shape structures. With the anchored global features, BPS [26] better captures the overall shapes but fails to reconstruct local details. ConvONet [25] yields smoother local geometry but is unable to reconstruct sharp features. Points2Surf [11] tends to produce random noises near reconstructed surfaces. While showing high performance in quantitative evaluations, visual results from UNDC [7] often contain holes with sparse point cloud inputs. Compared to all those methods, ARO-Net is able to reconstruct both global structure and local detail well, including sharp features as UNDC, and its results exhibit the least amount of artifacts in terms of holes and noise.

ARO-Net also exhibits superior robustness to sparsity of input point clouds. Once trained with 2,048-point input, ARO-Net can produce quality results with sparser inputs *without re-training*. As shown in Figure 7, when the number of points decreases from 2,048 to 512, the reconstruction quality of both UNDC [7] and Points2Surf [11] degrades dramatically, as also reflected in the CD and IOU metrics. Specifically, Points2Surf [11] tends to generate more holes and UNDC [7] produces rougher, oscillatory surfaces. ConvONet [25] is relatively stable but holes also appear with 512-point inputs. By contrast, ARO-Net yields the highest level of robustness both visually and quantitatively.

Reconstruction on ShapeNet V1. For our experiments on the ShapeNet V1 dataset, we train ARO-Net on 4K chairs and test it on other object categories: airplanes and rifle in ShapeNet V1 and animals in PSB [17]. Quantitative comparisons are shown in Table 2, with some visual results in Figure 8. It is clear that implicit models employing global features do not generalize well. Compared to UNDC [7],
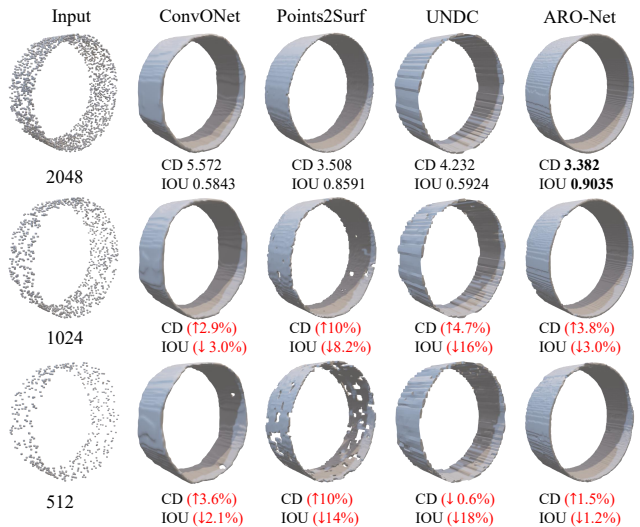

Figure 7. ARO-Net exhibits superior robustness to sparsity of input point clouds compared to its close competitors.

ConvONet [25], and Points2Surf [11], ARO-Net performs the best overall, quantitatively, especially with the test on airplanes, showing its strong generalizability. Visually, ARO-Net produces the most complete, artifact-free results with faithful reconstruction of fine details. In the airplane results in second row, it is worth observing that ARO-Net managed to reconstruct the left-side turbine from extremely sparse points, while other methods all completely missed it.

One-shape training. As a stress test, we train the networks using only one 3D shape, Fertility (see Figure 1), with rotation and scaling. From the comparisons in Figure 9, we see that SPR suffers from noise and under-sampling (e.g., chair and human hand), as expected. OccNet clearly over-fit to the training shape. ConvONet generalizes better but produces overly smoothed results. In contract, ARO-Net re-
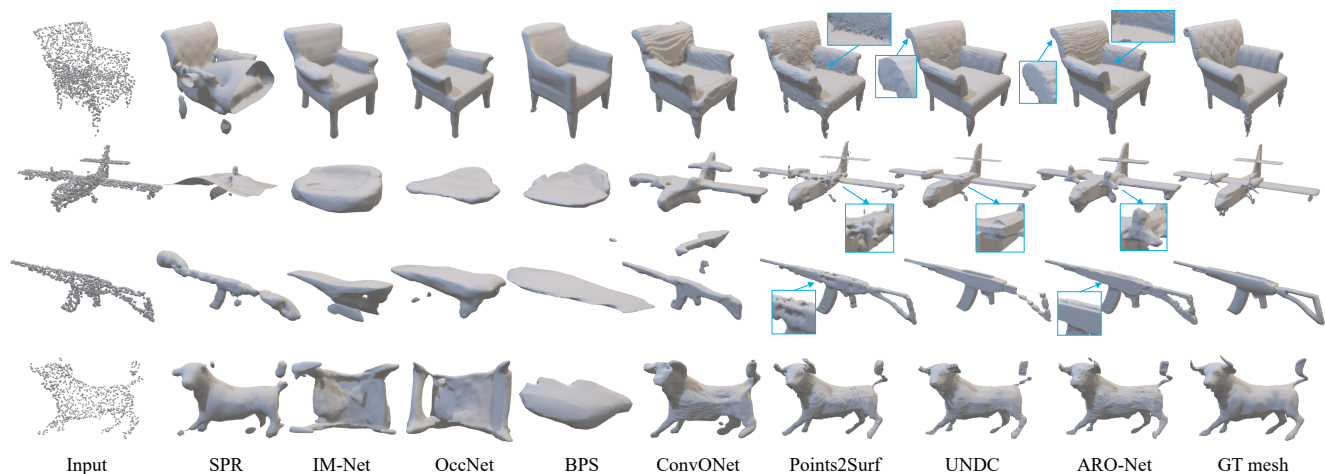
Figure 8. Visual comparisons on ShapeNet V1 and PSB datasets to show reconstruction quality and generalizability of different methods. All methods (except SPR) were trained on chairs in ShapeNet and tested on chairs, airplanes, and rifles in ShapeNet V1 and animals in PSB. Zoom-ins highlight contrasts in details and artifacts in the reconstructions. More results can be found in supplementary material.

Trained on chairs, tested on chairs			
Method	LFD↓	HD↓	CD↓	EMD↓
IM-Net [8]	3.18	9.04	13.84	15.40
OccNet [23]	2.59	7.77	12.36	13.47
BPS [26]	4.31	12.28	20.51	23.30
ConvONet [25]	2.12	6.22	11.20	12.30
Points2Surf [11]	2.51	6.46	8.60	7.34
UNDC [7]	2.19	5.60	**6.06**	**6.80**
ARO-Net	**1.92**	**5.33**	7.14	7.70

Trained on chairs, tested on airplanes			
Method	LFD↓	HD↓	CD↓	EMD↓
IM-Net [8]	14.30	19.50	44.84	41.38
OccNet [23]	12.31	18.86	39.75	38.57
BPS [26]	13.73	19.48	38.10	36.61
ConvONet [25]	5.69	15.98	13.50	10.75
Points2Surf [11]	5.48	4.70	4.86	5.76
UNDC [7]	3.62	4.40	**3.98**	**3.98**
ARO-Net	**3.56**	**4.32**	4.61	4.73

Table 2. Quantitative comparisons to state-of-the-art methods on the ShapeNet V1 dataset. Number in bold means ranking the first and the one with underline means ranking the second in each column.

constructs both global structures and fine details better than the alternatives, with consistence over the shape variety.

4.3. Ablation study

Anchor visibility vs. point occupancy. To verify whether ARO-Net can learn the close correlation between anchor visibility and point occupancy with the help of a visualization, we implemented a 2D vision of ARO-Net with implementation details provided in the supplementary material. We train ARO-Net using the letter "G" with anchors distributed
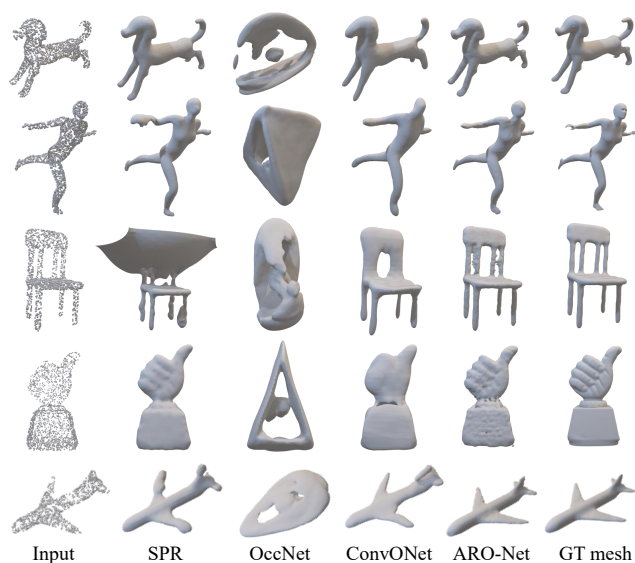

Figure 9. Visual comparison on the results obtained by training only on the "fertility" model with rotation and scaling.

as shown in Figure 4(c). Figure 10 visualizes the activation of each anchor by masking out the other anchors' radial observations. The output with all seven anchors considered are shown in the top-left corner. The probability of each point being inside the shape is drawn using the blue to yellow colormap, and the anchors are shown in white dots. We can see that the decision boundary of each anchor closely follows its visible region, without having that information specifically passed to ARO-Net. Another noteworthy observation is that for regions invisible to the anchor, points closer to the visibility boundary are assigned higher probability of
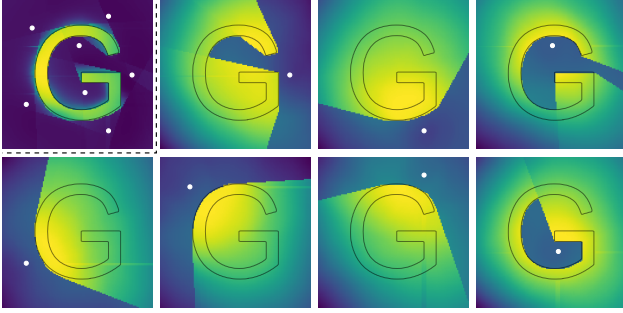
Figure 10. Visualization of ARO-Net activation, trained on a single letter "G" with seven anchors (top-left). The activation for each anchor is closely-related to its visible region.

Setting	LFD↓	HD↓	CD↓	EMD↓	IOU↑
$m = 24$	1.48	2.33	5.94	1.28	8.66
$m = 96$	1.30	2.16	5.31	1.03	8.92
MLP	1.44	2.30	5.90	1.14	8.75
Default	1.35	2.25	5.46	1.12	8.79

Table 3. Ablation studies on ABC to evaluate our choices for the anchor count (m) and decoder architecture. Default: $m = 48$ with attention modules in implicit decoder.

being inside the shape. This offers evidence that ARO-Net does implicitly learn to predict the point occupancy based on anchor visibility.

Anchor placement strategy. Other than our layered Fibonacci sampling, we also tried two different sampling strategies: uniform and grid-based sampling. In uniform sampling, we randomly sample m points in a unit sphere as anchors. In grid-based sampling, we randomly sample m grid points of a unit cube as anchors. We found that our layered Fibonacci sampling achieves the best performance. More detailed comparison can be found in the supplementary material.

Anchor count. There is a trade-off between reconstruction performance and computational cost when choosing, m, the number of anchors. We tested $m = \{24, 48, 96\}$ and report results in the top half of Table 3. We found that the performance gain by $m = 96$ is marginal, while the computation cost doubles. With $m = 24$, ARO's performance drops significantly due to lack of sufficient shape coverage.

Decoder architecture. The last two rows of Table 3 compare the current ARO-Net with a version by replacing our attention module and implicit decoder with the much simpler MLPs by IM-Net [8]. The similar performance numbers indicate that the superior results by ARO-Net are predominantly owing to ARO rather than the decoder architecture.

Fixed vs. adaptive anchors. When computing the radial observations, we perform top-k selection (pooling) on the cosine and Euclidean distances between the anchors and input points. Therefore, our network loss function is differentiable with respect to anchor coordinates: the gradients are passed through the indices that form the top k values, and then passed to anchor coordinates. While this does offer the possibility of fine-tuning the anchor positions during training and testing, we have found that optimizing anchor positions along with the network parameters can make the training process unstable and lead to worse results. Overall,

we believe that with the anchors fixed, it is easier for the network to learn the mapping from ARO to point occupancy.

During inference, it is possible to adjust the anchor locations to obtain a larger coverage of the shape surface to improve reconstruction. However, we found the improvements to be marginal; see supplementary material.

Distance and direction information in ARO. At last, we have confirmation that the r and $\|r\|$ play an important role as input to ARO-net. The reconstruction performance by the network drops significantly when we remove such information, with only the PointNet features. Again, more details can be found in the supplementary material.

5. Conclusion, limitation, and future work

We introduce a novel neural implicit representation for 3D shapes that is conditioned on Anchored Radial Observations (ARO). We use ARO to train our reconstruction network, ARO-Net, that outputs an occupancy at any query point in space. ARO is query-specific, defined by a set of local descriptors from the perspective of a set of fixed anchors to provide a global context. ARO-Net is category-agnostic and provides superior reconstruction results to alternative methods. It also has strong generalization capabilities and can even be used to learn from a single shape.

Some current limitations and future works are as follows. The quality and efficiency of the resulting representation are dependant on the number and placement of the anchors. We tested a number of configurations and settled on a fixed set of 48 anchors via Fibonacci sampling. However, further research into anchor selection, even learning, is needed. In addition, the ARO-Net representation still requires storing the original input point set and therefore is less memory efficient than pure network-based representations. Lastly, both training and inference of ARO-Net are slower than prior implicit models based on simpler and less contextual shape encodings (e.g. ConvONet). This can be alleviated by adaptive sampling of the anchors and using more efficient spatial search data structures to find closest points.

References

[1] Matthew Berger, Andrea Tagliasacchi, Lee M. Seversky, Pierre Alliez, Gaël Guennebaud, Joshua A. Levine, Andrei Sharf, and Claudio T. Silva. A survey of surface reconstruction from point clouds. *Computer Graphics Forum*, 36(1), 2017.

[2] 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.

[3] Rohan Chabra, Jan E. Lenssen, Eddy Ilg, Tanner Schmidt, Julian Straub, Steven Lovegrove, and Richard Newcombe. Deep local shapes: Learning local sdf priors for detailed 3D reconstruction. In *Eur. Conf. Comput. Vis.*, 2020.

[4] 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] Angel X. Chang, Thomas Funkhouser, Leonidas Guibas, Pat Hanrahan, Qixing Huang, Zimo Li, Silvio Savarese, Manolis Savva, Shuran Song, Hao Su, Jianxiong Xiao, Li Yi, and Fisher Yu. ShapeNet: An information-rich 3D model repository. *arXiv preprint arXiv:1512.03012*, 2015.

[6] Ding-Yun Chen, Xiao-Pei Tian, Yu-Te Shen, and Ming Ouhyoung. On visual similarity based 3d model retrieval. *Computer Graphics Forum*, 22(3):223–232, 2003.

[7] Zhiqin Chen, Andrea Tagliasacchi, Thomas Funkhouser, and Hao Zhang. Neural dual contouring. *ACM Trans. Graph.*, 41(4), 2022.

[8] Zhiqin Chen and Hao Zhang. Learning implicit fields for generative shape modeling. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2019.

[9] Zhiqin Chen and Hao Zhang. Neural marching cubes. *ACM Trans. Graph.*, 40(6), 2021.

[10] Julian Chibane, Thiemo Alldieck, and Gerard Pons-Moll. Implicit functions in feature space for 3D shape reconstruction and completion. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2020.

[11] Philipp Erler, Paul Guerrero, Stefan Ohrhallinger, Niloy J. Mitra, and Michael Wimmer. Points2Surf: Learning implicit surfaces from point clouds. In *Eur. Conf. Comput. Vis.*, pages 108–124, 2020.

[12] Ran Gal, Ariel Shamir, Tal Hassner, Mark Pauly, and Daniel Cohen-Or. Surface reconstruction using local shape priors. In *Symp. on Geometry Processing*, 2007.

[13] Kyle Genova, Forrester Cole, Avneesh Sud, Aaron Sarna, and Thomas Funkhouser. Local deep implicit functions for 3D shape. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2020.

[14] 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.

[15] Chiyu Max Jiang, Avneesh Sud, Ameesh Makadia, Jingwei Huang, Matthias Nießner, and Thomas Funkhouser. Local implicit grid representations for 3D scenes. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2020.

[16] Tao Ju, Frank Losasso, Scott Schaefer, and Joe Warren. Dual contouring of Hermite data. *ACM Trans. Graph.*, 21(3):339–346, 2002.

[17] Evangelos Kalogerakis, Aaron Hertzmann, and Karan Singh. Learning 3D mesh segmentation and labeling. *ACM Trans. Graph.*, 29(4):1–12, 2010.

[18] Michael Kazhdan and Hugues Hoppe. Screened Poisson surface reconstruction. *ACM Trans. Graph.*, 32(3), 2013.

[19] Benjamin Keinert, Matthias Innmann, Michael Sänger, and Marc Stamminger. Spherical fibonacci mapping. *ACM Trans. Graph.*, 34(6):1–7, 2015.

[20] Sebastian Koch, Albert Matveev, Zhongshi Jiang, Francis Williams, Alexey Artemov, Evgeny Burnaev, Marc Alexa, Denis Zorin, and Daniele Panozzo. Abc: A big cad model dataset for geometric deep learning. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 9601–9611, 2019.

[21] Manyi Li and Hao Zhang. D^2IM-Net: Learning detail disentangled implicit fields from single images. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2021.

[22] William E. Lorensen and Harvey E. Cline. Marching cubes: A high resolution 3D surface construction algorithm. In *SIGGRAPH*, page 163–169, 1987.

[23] Lars Mescheder, Michael Oechsle, Michael Niemeyer, Sebastian Nowozin, and Andreas Geiger. Occupancy networks: Learning 3d reconstruction in function space. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 4460–4470, 2019.

[24] Jeong Joon Park, Peter Florence, Julian Straub, Richard Newcombe, and Steven Lovegrove. DeepSDF: Learning continuous signed distance functions for shape representation. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 165–174, 2019.

[25] Songyou Peng, Michael Niemeyer, Lars Mescheder, Marc Pollefeys, and Andreas Geiger. Convolutional occupancy networks. In *Eur. Conf. Comput. Vis.*, 2020.

[26] Sergey Prokudin, Christoph Lassner, and Javier Romero. Efficient learning on point clouds with basis point sets. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 4332–4341, 2019.

[27] Charles R. Qi, Hao Su, Kaichun Mo, and Leonidas J Guibas. PointNet: Deep learning on point sets for 3D classification and segmentation. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2017.

[28] 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 *Int. Conf. Comput. Vis.*, 2019.

[29] Shy Shalom, Ariel Shamir, Hao Zhang, and Daniel Cohen-Or. Cone carving for surface reconstruction. *ACM Transactions on Graphics, Proceedings Siggraph Asia*, 29(5), 2010.

[30] Lior Shapira, Ariel Shamir, and Daniel Cohen-Or. Consistent mesh partitioning and skeletonisation using the shape diameter function. *The Visual Computer*, 24(4):249–259, 2008.

[31] 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.

[32] Edgar Tretschk, Ayush Tewari, Vladislav Golyanik, Michael Zollhoefer, Carsten Stoll, and Christian Theobalt. PatchNets: Patch-based generalizable deep implicit 3D shape representations. In *Eur. Conf. Comput. Vis.*, 2020.

[33] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

[34] Yiheng Xie, Towaki Takikawa, Shunsuke Saito, Or Litany, Shiqin Yan, Numair Khan, Federico Tombari, James Tompkin, Vincent Sitzmann, and Srinath Sridhar. Neural fields in visual computing and beyond. *Computer Graphics Forum*, 2022.

[35] Qiangeng Xu, Weiyue Wang, Duygu Ceylan, Radomir Mech, and Ulrich Neumann. Disn: Deep implicit surface network for high-quality single-view 3d reconstruction. In *Adv. Neural Inform. Process. Syst.*, pages 492–502, 2019.

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

[37] Xi Zhao, Ruizhen Hu, Paul Guerrero, Niloy Mitra, and Taku Komura. Relationship templates for creating scene variations. *ACM Trans. Graph.*, 35(6):1–13, 2016.