

000
001
002054
055
056

Surface Reconstruction from Point Clouds by Learning Predictive Context Priors

003
004
005
006
007
008
009
010
011
012
013
014
015057
058
059
060
061
062
063
064
065
066
067
068
069
070
071
072
073
074

Anonymous CVPR submission

Paper ID 536

Abstract

Surface reconstruction from point clouds is vital for 3D computer vision. State-of-the-art methods leverage large datasets to first learn local context priors that are represented as neural network-based signed distance functions (SDFs) with some parameters encoding the local contexts. To reconstruct a surface at a specific query location at inference time, these methods then match the local reconstruction target by searching for the best match in the local prior space (by optimizing the parameters encoding the local context) at the given query location. However, this requires the local context prior to generalize to a wide variety of unseen target regions, which is hard to achieve. To resolve this issue, we introduce Predictive Context Priors by learning Predictive Queries for each specific point cloud at inference time. Specifically, we first train a local context prior using a large point cloud dataset similar to previous techniques. For surface reconstruction at inference time, however, we specialize the local context prior into our Predictive Context Prior by learning Predictive Queries, which predict adjusted spatial query locations as displacements of the original locations. This leads to a global SDF that fits the specific point cloud the best. Intuitively, the query prediction enables us to flexibly search the learned local context prior over the entire prior space, rather than being restricted to the fixed query locations, and this improves the generalizability. Our method does not require ground truth signed distances, normals, or any additional procedure of signed distance fusion across overlapping regions. Our experimental results in surface reconstruction for single shapes or complex scenes show significant improvements over the state-of-the-art under widely used benchmarks.

1. Introduction

Surface reconstruction from 3D point clouds estimates continuous surfaces from 3D point clouds that can be captured by various 3D sensors. This is still a challenge even with the help of state-of-the-art deep learning models. A standard strategy [1, 12, 17] is to first learn a Signed Distance

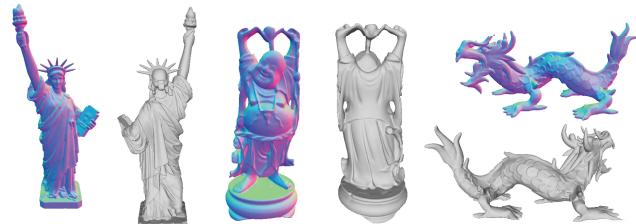


Figure 1. We reconstruct highly accurate surfaces from 300K points. Front views of each shape are shown with normal maps. The comparison with Poisson is in supplemental materials.

Function (SDF) from a point cloud [1, 17] or from ground truth signed distances [12] using a neural network, and then reconstruct a surface based on the learned SDF via marching cubes [40]. If the SDF is trained to capture a global shape prior from a global 3D shape, however, it is hard to capture local geometry details.

As a remedy, state-of-the-art methods learn local SDFs from local regions [6, 31, 62]. The global shape is usually split into overlapping [31, 62] or non-overlapping [6] parts, and the local region prior is learned as a local SDF that is represented by a neural network with some parameters encoding the geometry of local regions. The intuition behind this idea is that the local region prior will generalize to various unseen local reconstruction targets, and for surface reconstruction at inference time, its parameters can be optimized to match the reconstruction target at specific locations. However, the matching requires the learned local region prior to cover as many specific locations on target regions as possible, which dramatically limits the generalization ability of the learned local prior.

To resolve this issue, we propose to learn SDFs as a Predictive Context Prior for highly accurate surface reconstruction from point clouds, as shown in Fig. 1. Specifically, we first train a neural network to represent local SDFs of local regions across a large dataset of point clouds. This aims to capture a local context prior in a local coordinate system, similar as in previous work. Our main contribution is that during surface reconstruction at inference time, we specialize the pre-trained local context prior into a *Predictive Context Prior* for a specific point cloud by learning *Predic-*

CVPR
#536

108 *tive Queries.* More concretely, Predictive Queries learn to
109 predict query locations for the pre-trained local SDF from
110 queries given in the global coordinate system of the specific
111 point cloud. This is achieved by sampling a set of queries in
112 the global coordinate system, and learning to predict queries
113 for the local SDF to minimize surface reconstruction error.

114 Intuitively, learning to predict query locations for the lo-
115 cal SDF allows us to more flexibly match the learned local
116 context prior to best fit the given point cloud during surface
117 reconstruction. This further leads to a global SDF for the
118 specific point cloud. Our method does not require ground
119 truth signed distances, normals in the training, or any addi-
120 tional post processing. Our contributions are as follows:

- i) We introduce a neural network architecture using Predictive Context Priors to learn SDFs for surface reconstruction from point clouds. Predictive Context Priors are implemented by learning Predictive Queries.
 - ii) We demonstrate that Predictive Queries allow us to improve generalizability of a pre-trained local context prior, which improves reconstruction accuracy.
 - iii) We report state-of-the-art results in surface reconstruction for single shapes or complex scenes under widely used benchmarks.

2. Related Work

Surface reconstruction has been studied for decades. Classic methods [5, 33, 40] do not leverage any prior learned from large scale datasets. With the development of deep learning, data driven strategies [8, 16, 19, 21, 23, 32, 44, 46, 49, 62] can learn effective priors from datasets to improve the surface reconstruction accuracy. We will focus on reviewing the studies of deep learning based methods.

Deep Learning based Surface Reconstruction. The state-of-the-art methods mainly represent the reconstruction target as an implicit function [8, 11, 16, 21, 32, 44, 47, 49, 53, 56, 59, 62], due to advantages of SDFs or occupancy fields over other representations in representing high resolution shapes with arbitrary topology. To reveal more detailed geometry, one strategy is to leverage more latent codes [43, 58] to capture local shape priors as SDFs [6, 31, 39]. This requires to split the point cloud into different voxels, and then represent the points in each voxel as a latent code that is either extracted by a neural network [31, 39] or learned in an auto-decoding manner [6, 31]. These methods need normals for each point to produce signed distances as supervision in the optimization. Given the ground truth signed distances, Points2Surf [12] encodes points sampled in a local patch and on the whole point cloud as a shape prior, while DeepMLS [37] learns to produce oriented points to approximate SDFs. Similarly, PatchNet [62] learns local SDFs to

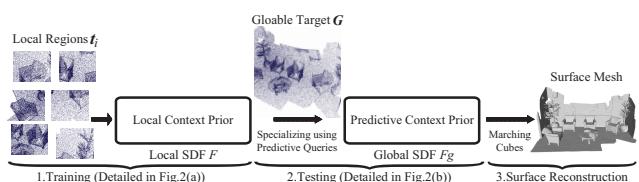


Figure 2. Overview of our method.

represent patches with explicit control over positions, orientations, and scales. Neural-pull [42] introduced a new way of learning SDFs by pulling nearby space onto the surface, which is achieved by predicting the SDFs and its gradient using the network. This removes the requirement of ground truth normals or signed distances. A similar idea is introduced to learn unsigned distances [9], but requires to move dense sampling with additional directions to form the surface. Moreover, other novel ways for surface reconstruction have been proposed, such as a differentiable formulation of Poisson solver [50] and part retrieval [55].

Other information is also leveraged to learn implicit functions [45, 57, 68]. Occupancy is used to capture a prior at a global level [30, 57] or a local level [45]. Iso-points [68] tried to impose geometry-aware sampling and regularization in the learning. Moreover, implicit functions can also be learned from point clouds with additional constraints, such as geometric regularization [17], sign agnostic learning with a specially designed loss function [1], sign agnostic learning with local surface self-similarities and post sign processing [60, 70], constraints on gradients [2] or a divergence penalty [4].

From a meshing perspective, surfaces can also be reconstructed by generating local connectivity with intrinsic-extrinsic metrics [36], Delaunay triangulation of point clouds [41] or inheriting connectivity from an initial mesh [25]. With local chart parameterizations in neural networks, a local point cloud is reconstructed via fitting using the Wasserstein distance as a measure of approximation [65].

Deep Shape Prior. Beside the priors reviewed above, shape priors can also be captured by parameters in neural networks in shape reconstruction [3, 14, 18, 20, 22, 23, 29, 54, 63, 67], segmentation [38, 51, 52], and completion [27, 28, 64, 66, 69]. Deep manifold prior [15] was introduced to reconstruct 3D shapes starting from random initializations.

3. Method

Overview. We provide an overview of our method in Fig. 2. We aim to reconstruct a surface mesh for a 3D point cloud G . Our method consists of the following three stages.

- During training, we start by learning a local context prior as a local SDF F by training a neural implicit network under a local region set $\mathbf{T} = \{\mathbf{t}_i, i \in [1, I]\}$. As shown in Fig. 3 (a), the neural implicit network learns F as a mapping from a query point \mathbf{q}_l with its corresponding condition \mathbf{f}_l to a

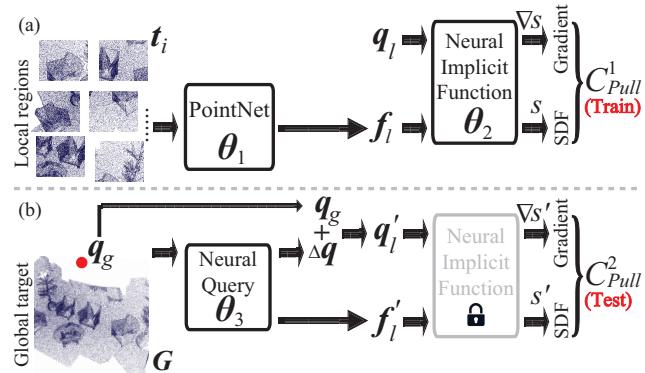


Figure 3. We learn local context prior from each local region t_i as a local SDF F in (a). We learn a predictive context prior to reconstruct G by predicting queries q'_l associated with conditions f'_l for F in (b).

signed distance s in a local coordinate system.

2. For surface reconstruction at test time, we specialize the local context prior into a predictive context prior for a specific point cloud G by learning predictive queries. The predictive context prior leads to a global SDF F_g to fit the point cloud G . Specifically, the learned local context prior is represented by the fixed parameters in the neural implicit network, as shown in Fig. 3 (b). We train an additional neural query network to predict queries q'_l and their conditions f'_l to explore the learned local context prior, which significantly improves the generalization ability of the learned local prior by enabling us to flexibly search the whole learned prior space.

3. Finally, we leverage the global SDF F_g to reconstruct the surface of G using the marching cubes algorithm [40].

Local Context Prior. During training shown in Fig. 3 (a), we learn a local context prior from local point clouds $t_i \in \mathbf{T}$ as a local SDF F in a local coordinate system. We learn the context around each point of t_i using a neural implicit network. For a local point cloud t_i , we normalize all points of t_i by centering them to the origin, and then linearly scaling them to fit the longest edge of the bounding box of t_i into a range of $[-0.5, 0.5]$. This normalization makes the local context on different t_i comparable to each other.

We leverage PointNet [51] to extract the feature f_l of each local region t_i . f_l is regarded as a condition of any query point q_l sampled near t_i when training the neural implicit network as a local SDF F , such that the signed distance s at the location of q_l is,

$$s = F(q_l, f_l). \quad (1)$$

To remove the requirement of ground truth signed distance values or normals in training, we minimize a pulling cost introduced in Neural-pull [42] to train the local SDF F . We simultaneously optimize the parameters of θ_1 in PointNet and θ_2 in the neural implicit network. The intuition of the pulling cost is to pull a query q_l using the predicted

signed distance s to its nearest neighbor $nn(q_l)$ on region t_i along the direction of the gradient $\nabla s = \partial F / \partial q_l$ at q_l . Our objective function during training is to minimize the pulling cost C_{Pull}^1 ,

$$\min_{\theta_1, \theta_2, nn(q_l) \in t_i} \|nn(q_l) - (q_l - s \times \nabla s / \|\nabla s\|_2)\|_2. \quad (2)$$

Predictive Context Prior. For surface reconstruction at test time, we first specialize the learned local context prior into a predictive context prior for a specific point cloud G . Point cloud G is located in a global coordinate system without normalization. We train an additional neural query network with parameters of θ_3 specially for G , where we keep the neural network parameters θ_2 representing the learned local context prior fixed. This leads to a global SDF F_g that captures the predictive context prior which we use to reconstruct the surface of G .

The neural query network learns to generate predictive queries, that is, to transform a query point q_g around G in a global coordinate system into a point q'_l in the local coordinate system that the learned local context prior covers. In addition, the neural query network also predicts the condition f'_l of predictive query q'_l . Here, we are inspired by the idea of ResNet [26], and predict the shift Δq from q_g to q'_l ,

$$q'_l = q_g + \Delta q. \quad (3)$$

The intuition behind the neural query network is to train a network specific to point cloud G that is able to manipulate the queries for the learned local context prior. This prediction is equivalent to flexibly searching for correct information from the learned local SDF F , and then combining them together to fit the point cloud G . This leads to a global SDF F_g that predicts the signed distance s' at a query location of q_g with a condition of G ,

$$s' = F_g(q_g, G) = F(q'_l, f'_l). \quad (4)$$

Similar to Eq. (2) in training, we further optimize the parameters θ_3 of the neural query network to pull the query q_g in the global coordinate system to its nearest neighbor $nn(q_g)$ on point cloud G . We leverage the learned local SDF to produce the gradient, $\nabla s' = \partial F / \partial q'_l$. So, our objective function during testing is to minimize a pulling cost C_{Pull}^2 below,

$$\min_{\theta_3, nn(q_g) \in G} \|nn(q_g) - (q_g - s' \times \nabla s' / \|\nabla s'\|_2)\|_2. \quad (5)$$

Reconstruction. After we learn parameters θ_3 in the neural query network using Eq. (5), we keep θ_3 and the parameters θ_2 in the neural implicit network fixed to produce the global SDF F_g for point cloud G , which is then used to reconstruct the surface using marching cubes [40].

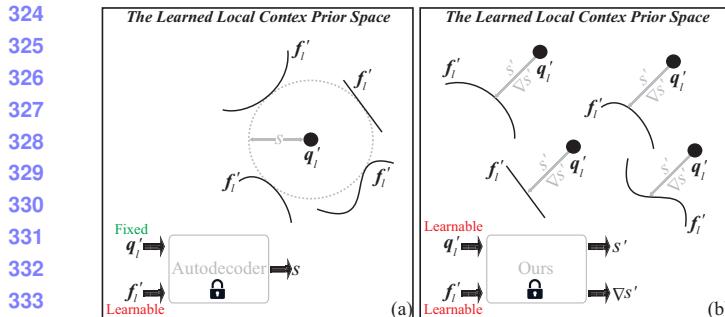


Figure 4. Current methods can merely search similar regions at a specific location q'_l in the learned prior space by optimizing condition f'_l . While we can search over the whole prior space by simultaneously optimizing q'_l and f'_l .

Optimization. We conduct the optimization of Eq. (2)(training) and Eq. (5)(inference) using a similar procedure. For each point p on t_i or G , we randomly sample 40 queries q_l around p . Due to the difference numbers of points on each local region $t_i \in T$, we randomly select 2000 q_l around t_i , and regard their nearest neighbors $\{nn(q_l)\}$ on t_i as the input to PointNet in each training epoch, where the randomness makes the local context prior more robust to noise. We perform this optimization in an overfitting manner, either on each single G or multiple point clouds with one-hot vectors as the condition of each q_g .

Intuition and Advantages. The intuition of our predictive context prior is to leverage information at different locations queried from the learned prior in local SDF F to form a global SDF F_g , which significantly improves the generalization ability of the learned local context prior.

A widely used strategy to explore the local context prior is to use an autodecoder [22, 31, 49], as shown in Fig. 4 (a). The autoencoder aims to optimize a learnable condition f'_l for a fixed location q'_l to minimize the signed distance error compared to the ground truth s . This optimization is equivalent to finding a local region whose feature is f'_l with a nearest distance to the location q'_l as s in the space covered by the learned local context prior. Since the specific location q'_l is fixed, the performance is only guaranteed when a qualified local region represented by f'_l has been seen during the learning of the local context prior, which is hard to generalize for various unseen regions during test.

Differently, without requiring the ground truth signed distances, our method aims to find a similar way of pulling a location q'_l to a local region represented by f'_l , where the pulling is implemented by a signed distance prediction s' and its gradient $\nabla s'$. As shown in Fig. 4 (b), rather than searching (optimizing f'_l) at a fixed location q'_l like an autodecoder, our neural query network can adjust query locations, which makes it possible to search a similar pulling way across the whole space covered by the learned local context prior.

Obviously, our advantage is the ability of transforming

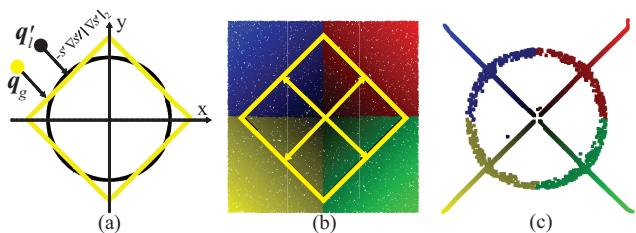


Figure 5. Demonstration of predicted queries q'_l (c) from queries q_g (b). Optimization is visualized in our video supplementary.

the searching at a specific location into anywhere across the learned context prior. This advantage not only significantly improves the generalization ability of the learned prior, but also dramatically reduces the requirement of the local regions used to learn the local context prior, since it is easy to observe various ways of pulling points to the surface around arbitrary local regions during training.

We further demonstrate our advantages in a 2D case. We aim to estimate an SDF F_g of a yellow square in Fig. 5 (a) from an SDF F learned from a black circle in Fig. 5 (a) during training. During testing, we sample query points q_g in the space occupied by the yellow square, as shown by the dense points in color in Fig. 5 (b). After the optimization using Eq. (5), we visualize the predicted queries q'_l obtained from Eq. (3) in Fig. 5 (c), where the color of each q'_l is the same as the corresponding q_g . The color correspondence demonstrates that the q'_l appearing on the radius with an angle of 45° to the x-axis can provide the expected s' and $\nabla s'$ to get the corresponding q_g pulled on the square, i.e., the appropriate signed distances s' and gradients $\nabla s'$ with 45° to the x-axis (shown by black arrows in Fig. 5 (a) and yellow arrows in Fig. 5 (b)), since q'_l gets pulled to the circle using the same s' and $\nabla s'$, while the q'_l appearing on the circle correspond to q_g on the square, both of which have 0 signed distances. This example shows that our method can flexibly search over the whole learned local context prior and easily find the correct prior information. This significantly increases our generalization ability.

4. Experiments, Analysis, and Discussion

4.1. Setup

Implementation details. To predict a signed distance value s , we use an OccNet [44] without activation functions in the last layer. The condition f'_l or f_l is 512 dimensional. The neural query network is a feed-forward network with 8 layers, where each one of the first 7 layers has 512 nodes with ReLU activation functions while the last layer has 515 nodes with linear activation functions to predict the 512 dimensional condition f'_l and 3 dimensional query q'_l .

We separate each 3D shape or scene in the training set under each benchmark into a 6^3 grid according to its bounding box, where the points located in each grid form a local region t_i in T . In addition, we use the same method as

324	378
325	379
326	380
327	381
328	382
329	383
330	384
331	385
332	386
333	387
334	388
335	389
336	390
337	391
338	392
339	393
340	394
341	395
342	396
343	397
344	398
345	399
346	400
347	401
348	402
349	403
350	404
351	405
352	406
353	407
354	408
355	409
356	410
357	411
358	412
359	413
360	414
361	415
362	416
363	417
364	418
365	419
366	420
367	421
368	422
369	423
370	424
371	425
372	426
373	427
374	428
375	429
376	430
377	431

432 Neural-pull [42] to sample 40 queries q_l or q_g around each
 433 point p on t_i or G , respectively, where a Gaussian function
 434 $\mathcal{N}(p, \sigma^2)$ is used to calculate the sampling probability, and
 435 σ^2 is the 50-th nearest neighbor of p on t_i or G .

436 **Dataset.** In surface reconstruction for 3D shapes, we evaluate
 437 our method under three datasets including ABC [34],
 438 FAMOUS [12], and a subset of ShapeNet [7]. In surface
 439 reconstruction for scenes, we report our results under two
 440 datasets including 3D Scene [71] and SceneNet [24]. Under
 441 ShapeNet and ABC, we leverage marching cubes on a
 442 128^3 grid to reconstruct meshes, while using a 512^3 grid
 443 under FAMOUS, 3D Scene and SceneNet.

444 **Metrics.** Under the ABC and FAMOUS datasets, we randomly
 445 sample 1×10^4 points on the reconstructed mesh to compare
 446 with the input point clouds using L2-CD which keeps the same as the setting in Points2Surf [12]. We also follow MeshingPoint [36] to report our results under ShapeNet [7] in terms of L1-CD, Normal Consistency (NC) [44], and F-score [61] to evaluate the reconstruction performance, where we evaluate the distance between the 1×10^5 points sampled on the reconstructed shape and the 1×10^5 ground truth points released by OccNet [44]. The threshold μ in F-score calculation is 0.002 which is the same as MeshingPoint [36] and Neural-pull [42].

447 Under the 3D Scene [71] dataset, we follow DeepLS [6] to report the error between reconstructed meshes and the
 448 ground truth mesh. The error with a unit of mm is the average distance from each reconstructed vertex to its nearest triangle on the ground truth mesh. We also produce L1-CD,
 449 L2-CD, and normal consistency to compare with others.

450 Under the SceneNet [24] dataset, we follow LIG [31] to report L1-CD, Normal Consistency (NC) [44], and F-score [61] under different sampling densities on the reconstructed meshes, such as 20, 100, 500 and 1000 points per m^2 , where the threshold μ_s in F-score calculation is 0.025 m which is the same as LIG [31].

4.2. Surface Reconstruction for Single Shapes

471 **Evaluation under ShapeNet.** We first report our numerical
 472 comparison under the ShapeNet subset by comparing with the non data-driven and the latest data-driven methods in terms of L2-CD in Tab. 1, normal consistency in Tab. 2, and F-score with a threshold of μ in Tab. 3, and 2μ in Tab. 4. The compared methods include Poisson Surface Reconstruction (PSR) [33], Ball-Pivoting algorithm (BPA) [5], AtlasNet (ATLAS) [18], Deep Geometric Prior (DGP) [65], Deep Marching Cube (DMC) [35], DeepSDF (DSDF) [49], MeshP [36], Neural Unsigned Distance (NUD) [9], SALD [2], Local Implicit Grid (LIG) [31], IMNET [8], and Neural-Pull(NP) [42].

473 The reconstruction accuracy in Tab. 1 demonstrates that our method reveals the most accurate surface from point
 474 clouds even under some challenging classes, such as Lamp,

Class	PSR	DMC	BPA	ATLAS	DMC	DSDF	DGP	MeshP	NUD	SALD	NP	Ours
Display	0.273	0.269	0.993	1.094	0.662	0.317	0.293	0.069	0.077	-	0.039	0.0087
Lamp	0.227	0.244	0.060	1.988	3.377	0.958	0.167	0.053	0.075	0.071	0.080	0.0380
Airplane	0.217	0.171	0.059	1.011	2.205	1.043	0.200	0.049	0.076	0.054	0.008	0.0065
Cabinet	0.36	0.373	0.292	1.661	0.766	0.921	0.237	0.112	0.041	-	0.026	0.0153
Vessel	0.254	0.228	0.078	0.997	2.487	1.254	0.199	0.061	0.079	-	0.022	0.0079
Table	0.383	0.375	0.120	1.311	1.128	0.660	0.333	0.076	0.067	0.066	0.060	0.0131
Chair	0.293	0.283	0.099	1.575	1.047	0.483	0.219	0.071	0.063	0.061	0.054	0.0110
Sofa	0.276	0.266	0.124	1.307	0.763	0.496	0.174	0.080	0.071	0.058	0.012	0.0086
Mean	0.286	0.276	0.116	1.368	1.554	0.766	0.228	0.071	0.069	0.062	0.038	0.0136

Table 1. L2-CD ($\times 100$) comparison under ShapeNet.

Class	PSR	DMC	BPA	ATLAS	DMC	DSDF	MeshP	LIG	IMNET	NP	Ours
Display	0.889	0.842	0.952	0.828	0.882	0.932	0.974	0.926	0.574	0.964	0.9775
Lamp	0.876	0.872	0.951	0.593	0.725	0.864	0.963	0.882	0.592	0.930	0.9450
Airplane	0.848	0.835	0.926	0.737	0.716	0.872	0.955	0.817	0.550	0.947	0.9490
Cabinet	0.880	0.827	0.836	0.682	0.845	0.872	0.957	0.948	0.700	0.930	0.9600
Vessel	0.861	0.831	0.917	0.671	0.706	0.841	0.953	0.847	0.574	0.941	0.9546
Table	0.833	0.809	0.919	0.783	0.831	0.901	0.962	0.936	0.702	0.908	0.9595
Chair	0.850	0.818	0.938	0.638	0.794	0.886	0.962	0.920	0.820	0.937	0.9580
Sofa	0.892	0.851	0.940	0.633	0.850	0.906	0.971	0.944	0.818	0.951	0.9680
Mean	0.866	0.836	0.923	0.695	0.794	0.884	0.962	0.903	0.666	0.939	0.9590

Table 2. Normal consistency comparison under ShapeNet.

Class	PSR	DMC	BPA	ATLAS	DMC	DSDF	DGP	MeshP	NUD	LIG	IMNET	NP	Ours
Display	0.468	0.495	0.834	0.071	0.108	0.632	0.417	0.903	0.903	0.551	0.601	0.989	0.9939
Lamp	0.455	0.518	0.826	0.029	0.047	0.268	0.405	0.855	0.888	0.624	0.836	0.891	0.9382
Airplane	0.415	0.442	0.788	0.070	0.050	0.350	0.249	0.844	0.872	0.564	0.698	0.994	0.9942
Cabinet	0.392	0.392	0.553	0.077	0.154	0.573	0.513	0.860	0.950	0.733	0.343	0.980	0.9888
Vessel	0.415	0.466	0.789	0.058	0.055	0.323	0.387	0.862	0.883	0.467	0.147	0.985	0.9935
Table	0.233	0.287	0.772	0.080	0.089	0.577	0.307	0.880	0.908	0.844	0.425	0.922	0.9969
Chair	0.382	0.433	0.802	0.050	0.088	0.447	0.481	0.875	0.913	0.710	0.181	0.954	0.9970
Sofa	0.499	0.535	0.786	0.058	0.129	0.577	0.638	0.895	0.945	0.822	0.199	0.968	0.9943
Mean	0.407	0.446	0.769	0.062	0.091	0.468	0.425	0.872	0.908	0.664	0.429	0.961	0.9871

Table 3. F-score(μ) comparison under ShapeNet.

Class	PSR	DMC	BPA	ATLAS	DMC	DSDF	DGP	MeshP	NUD	LIG	IMNET	NP	Ours
Display	0.666	0.669	0.929	0.179	0.246	0.787	0.607	0.975	0.944	0.991	0.991	0.9958	
Lamp	0.648	0.681	0.934	0.077	0.113	0.478	0.662	0.951	0.945	0.924	0.9402	0.9505	
Airplane	0.619	0.639	0.914	0.179	0.289	0.566	0.515	0.946	0.944	0.997	0.9972	0.9937	
Cabinet	0.598	0.591	0.706	0.195	0.128	0.694	0.738	0.946	0.980	0.989	0.993	0.9939	
Vessel	0.633	0.647	0.906	0.153	0.120	0.508	0.648	0.956	0.945	0.990	0.9958	0.9988	
Table	0.442	0.462	0.886	0.195	0.221	0.743	0.494	0.963	0.962	0.973	0.973	0.9985	
Chair	0.617	0.615	0.913	0.134	0.345	0.665	0.693	0.964	0.954	0.969	0.991	0.9991	
Sofa	0.725	0.708	0.895	0.153	0.208	0.734	0.834	0.972	0.968	0.974	0.974	0.9987	
Mean	0.618	0.626	0.885	0.158	0.209	0.647	0.649	0.959	0.950	0.976	0.976	0.9989	

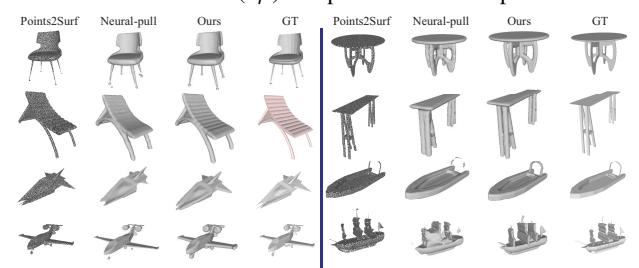
Table 4. F-score(2μ) comparison under ShapeNet.

Figure 6. Visual comparison under ShapeNet dataset.

475 **Chair, and Table.** Although we achieve comparable normal
 476 consistency to MeshP in Tab. 2, we do not require dense
 477 and clean point clouds as MeshP. In addition, our method
 478 outperforms all implicit function based methods including
 479 DSDF [49], NUD [9], SALD [2], LIG [31], IMNET [8],
 480 NP [42] in Tab. 1 and 2, which justifies our ability of lever-
 481 aging prior information in a more effective way.

482 Moreover, we also compare with other methods which
 483 reported a reconstruction accuracy over the subset under
 484 ShapeNet in terms of L1-CD in Tab. 5. These methods in-
 485 clude 3DR2 [10], PSGN [13], DMC [35], Occupancy Net-
 486 work (OccNet) [44], SSRNet [45], DDT [41], and NP [42].
 487 We also report our accuracy under each involved class at the
 488 bottom of the Tab. 5. This comparison further demonstrates
 489 our ability to reconstruct surfaces at high accuracy.

490 We visually compare with P2S and NP in Fig. 6. We
 491 present more accurate geometry on the complete surface
 492 while P2S fails to reconstruct the complete surface.

493 **Evaluation under ABC and FAMOUS.** Tab. 6 reports
 494 the comparison under ABC and FAMOUS dataset with

	3DR2	PSGN	DMC	OccNet	SSRNet	DDT	NP	Ours
540	0.169	0.202	0.117	0.079	0.024	0.020	0.011	0.0077
541	Display	Lamp	Airplane	Cabinet	Vessel	Table	Chair	Sofa
542	0.0073	0.0082	0.0057	0.0081	0.0071	0.0083	0.0080	0.0088

Table 5. Reconstruction accuracy in terms of L1-CD.

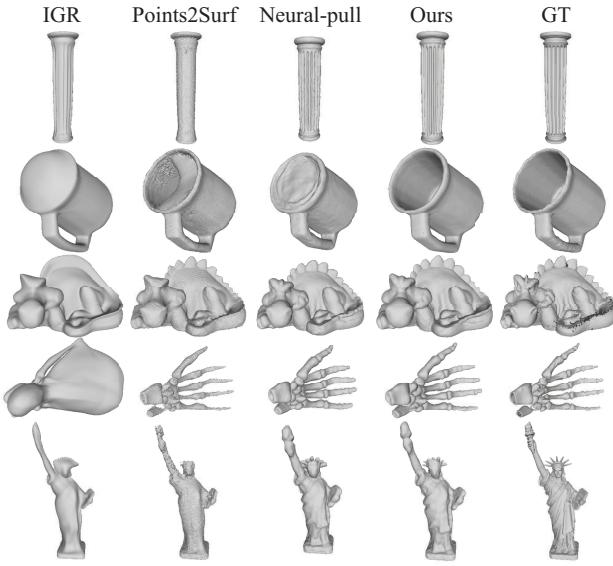


Figure 7. Visual comparison under FAMOUS dataset.

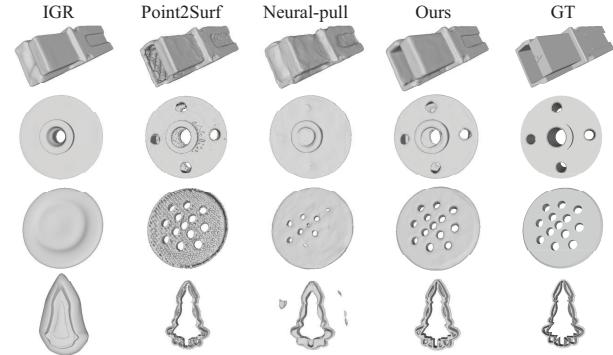


Figure 8. Visual comparison under ABC dataset.

DeepSDF (DSDF) [49], AtlasNet (ATLAS) [18], PSR [33], Points2Surf (P2S) [12], IGR [17], Neural-Pull (NP) [42] and IMLS [37]. The numerical comparison shows that our method significantly outperforms the other methods. We also highlight our advantage by visually comparing with I-GR, P2S, and NP under FAMOUS in Fig. 7 and under ABC in Fig. 8, where our reconstruction presents more geometry details with arbitrary topology.

We also evaluate our method under some variants of ABC and FAMOUS by adding different noise levels or changing the point density, which is released by P2S [12]. The L2-CD comparison in Tab. 7 demonstrates that our method is also good at resisting dramatic noise and density changes, and still achieves the best performance compared to the others.

Dataset	DSDF	ATLAS	PSR	P2S	IGR	NP	Ours
ABC	8.41	4.69	2.49	1.80	0.51	0.48	0.200
FAMOUS	10.08	4.69	1.67	1.41	1.65	0.22	0.044
Mean	9.25	4.69	2.08	1.61	1.08	0.35	0.122

Table 6. Reconstruction accuracy in terms of L2-CD ($\times 100$).

Dataset	DSDF	ATLAS	PSR	P2S	NP	IMLS	Ours
ABC var-noise	12.51	4.04	3.29	2.14	0.72	0.566	0.488
ABC max-noise	11.34	4.47	3.89	2.76	1.24	0.675	0.571
F-med-noise	9.89	4.54	1.80	1.51	0.28	0.798	0.071
F-max-noise	13.17	4.14	3.41	2.52	0.31	0.387	0.298
F-Sparse	10.41	4.91	2.17	1.93	0.84	-	0.083
F-Dense	9.49	4.35	1.60	1.33	0.22	-	0.087
Mean	11.73	4.30	3.10	2.23	0.60	-	0.266

Table 7. Noise and density in terms of L2-CD ($\times 100$).

4.3. Surface Reconstruction for Scenes

Evaluation under 3D Scene. We first evaluate our method by comparing with MPU [48], Convolutional OccNet (ConvOcc) [57], Local Implicit Grid (LIG) [31], Deep Local Shape (DeepLS) [6], and Neural-Pull (NP) [42] under five scenes in the 3D Scene dataset in Tab. 8. We use the official code of MPU and NP to produce their results, while using the trained ConvOcc and LIG from the author and normals of point clouds to report their results, where we do not use the post processing in LIG for fair comparison. Tab. 8 shows that our method can achieve much higher accuracy than these state-of-the-art methods in terms of different metrics, where we also do not require normals as LIG and DeepLS. The improvement over the-state-of-the-art is further demonstrated by the visual comparison in Fig. 9.

Evaluation under SceneNet. We compare our method with ConvOcc [57], LIG [31], and NP [42] under 5 classes in SceneNet. We produce the results of NP by training it using its code, while using the trained model of LIG to produce their results, where we do not leverage the post processing in LIG for fair comparison. The results in each of five classes in Tab. 9 show that our method achieves the best performances under different input point densities. Our visual comparison in Fig. 10 further shows that our method can reconstruct more detailed surfaces in complex scenes.

Reconstructions for Real Scan. We also show surface reconstruction comparison for a real scanned scene in our video and text supplementary.

4.4. Analysis and Discussion

We justify the effectiveness of each element in our network and explore the effect of some important parameters on the performance under the ABC dataset in terms of L2-CD and normal consistency.

Ablation Studies. We report ablation studies in Tab. 10. We first highlight the effectiveness of the predicted shift Δq by removing Δq from the network. We first try to directly use the query q_g from the global coordinate system as q'_g . We found that the performance degenerates dramatically, as shown by “No Δq ”. Then, we push the neural query network to predict q'_g directly. But the performance still goes

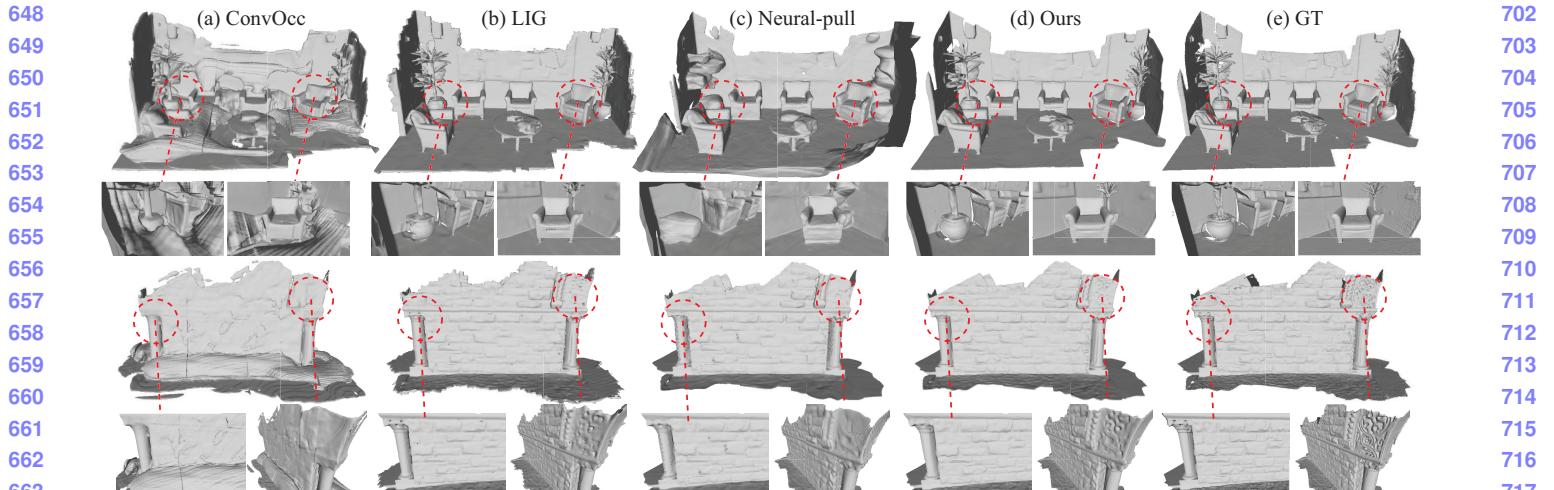


Figure 9. Visual comparison with the state-of-the-art under 3D Scene dataset.

	Burgers				Lounge				Copyroom				Stonewall				Totempole			
	L2CD	L1CD	Norm	Error	L2CD	L1CD	Norm	Error	L2CD	L1CD	Norm	Error	L2CD	L1CD	Norm	Error	L2CD	L1CD	Norm	Error
MPU	775.04	0.456	0.720	894.95	203.87	0.206	0.817	203.87	29.18	0.062	0.832	110.36	680.86	0.428	0.800	486.70	1652.25	0.671	0.763	1328.80
ConvOcc	26.69	0.077	0.865	354.60	8.68	0.042	0.857	73.20	10.99	0.045	0.848	79.68	19.12	0.066	0.866	120.61	1.16	0.016	0.925	21.56
LIG	0.839	0.018	0.904	28.70	0.789	0.017	0.910	28.20	0.906	0.018	0.910	30.50	1.08	0.020	0.928	33.65	1.37	0.023	0.917	38.90
DeepLS	-	-	-	5.74	-	-	-	7.38	-	-	-	10.09	-	-	-	6.45	-	-	-	8.97
NP	1.76	0.010	0.883	11.23	39.71	0.059	0.857	98.03	0.51	0.011	0.884	8.76	0.063	0.007	0.868	6.84	0.19	0.010	0.765	10.21
Ours	0.267	0.008	0.914	9.28	0.061	0.006	0.928	6.76	0.076	0.007	0.918	7.76	0.061	0.0065	0.888	6.33	0.10	0.008	0.784	8.36

Table 8. Surface reconstruction under 3D Scene. L2-CD×1000. Norm is short for normal consistency. The unit of error is mm.

down, as shown by “Direct q'_l ”. These two results demonstrate the importance of Δq in the learning.

Moreover, we highlight the effectiveness of the predicted condition f'_l by removing it from the output of the neural query network. We first leverage autodecoding similar as DeepSDF [49] to learn f'_l . The result of “No f'_l ” shows that this does not work well with the learnable Δq . Then, we try to use f_l from the trained PointNet to replace f'_l , but the result of “No f'_l+f_l ” gets worse neither. These experiments show that the learnable condition f'_l is only effective when it is optimized together with its corresponding query q'_l .

	No Δq	Direct q'_l	No f'_l	No f'_l+f_l	Ours
L2-CD	3.13	2.78	4.03	4.21	2.090
Normal	0.924	0.935	0.908	0.901	0.945

Table 10. Ablation studies under ABC. L2-CD×1000.

Specializing Context Prior.

We found that specializing the local context prior into the predictive context prior also plays an important role to reconstruct surfaces in high accuracy. We first highlight the importance of the local context prior by removing the training procedure shown in Fig. 3 (a). We overfit the global point cloud G by si-

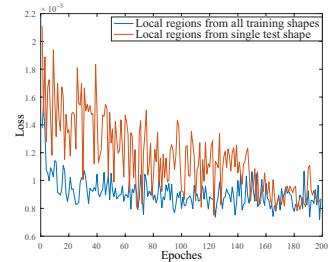


Figure 11. Loss with the prior learned from different regions.

multaneously optimizing the parameters θ_3 in the neural query network and θ_2 in the neural implicit network shown in Fig. 3 (b). The result of “No prior” in Tab. 11 shows that the performance significantly drops compared to “Our specializing”. Moreover, even with the learned local context prior as an initialization, if we tune θ_2 and θ_3 simultaneously in Fig. 3 (b), the result of “Tune $\theta_2+\theta_3$ ” is still not satisfactory. These experiments demonstrate the importance of the specializing in leveraging the learned prior.

	No Prior	Tune $\theta_2+\theta_3$	Our specializing
L2-CD	4.04	3.60	2.09
Normal	0.9200	0.9250	0.9446

Table 11. Effect of specializing under ABC. L2-CD×1000.

Normalizing Local Regions. We found the normalization of local regions t_i in T slightly affects the learning of the local context prior. As we mentioned before, we normalize t_i by centering and scaling it in the local coordinate system. We report the effect of centering and scaling on the performance in Tab. 12, which shows that both centering and scaling contribute to the increase of performance.

	No normalization	Only centering	Only scale	Ours
L2-CD	2.83	2.13	2.67	2.09
Normal	0.9360	0.9410	0.9380	0.9446

Table 12. Effect of normalization under ABC. L2-CD×1000.

Obtaining Local Regions t_i . We also explore how the size of local regions t_i affects the learning of the local context prior. We try to split each point cloud in the training set in-

	Livingroom			Bathroom			Bedroom			Kitchen			Office			Mean			
	L1CD	Norm	FScore																
20m ²	LIG	0.032	0.719	0.790	0.030	0.737	0.807	0.029	0.735	0.818	0.029	0.727	0.817	0.033	0.737	0.805	0.030	0.730	0.808
	NP	0.068	0.827	0.718	0.072	0.716	0.658	0.044	0.782	0.740	0.069	0.720	0.689	0.066	0.834	0.663	0.037	0.776	0.693
Ours	0.027	0.835	0.856	0.032	0.749	0.801	0.028	0.800	0.842	0.033	0.737	0.826	0.029	0.861	0.829	0.029	0.796	0.831	
100m ²	LIG	0.019	0.922	0.919	0.018	0.930	0.915	0.017	0.918	0.920	0.016	0.920	0.936	0.020	0.910	0.936	0.018	0.920	0.925
	NP	0.069	0.883	0.799	0.028	0.907	0.893	0.032	0.890	0.878	0.042	0.896	0.838	0.066	0.866	0.733	0.047	0.888	0.828
Ours	0.018	0.902	0.953	0.016	0.872	0.950	0.014	0.893	0.957	0.015	0.884	0.945	0.024	0.916	0.907	0.017	0.893	0.943	
500m ²	LIG	0.019	0.910	0.919	0.017	0.924	0.914	0.017	0.915	0.926	0.017	0.916	0.937	0.020	0.907	0.937	0.018	0.914	0.925
	NP	0.050	0.905	0.838	0.041	0.916	0.856	0.033	0.915	0.877	0.047	0.893	0.844	0.064	0.879	0.750	0.047	0.902	0.833
Ours	0.017	0.938	0.969	0.016	0.943	0.979	0.016	0.946	0.976	0.016	0.943	0.968	0.024	0.927	0.918	0.017	0.939	0.962	
1000m ²	LIG	0.020	0.910	0.920	0.017	0.927	0.910	0.017	0.919	0.924	0.017	0.920	0.936	0.020	0.910	0.936	0.018	0.917	0.925
	NP	0.088	0.881	0.801	0.036	0.912	0.860	0.034	0.905	0.876	0.049	0.900	0.825	0.062	0.879	0.729	0.054	0.895	0.818
Ours	0.017	0.933	0.966	0.016	0.945	0.977	0.014	0.948	0.980	0.015	0.945	0.976	0.024	0.919	0.925	0.017	0.938	0.965	

Table 9. Surface reconstruction under SceneNet.

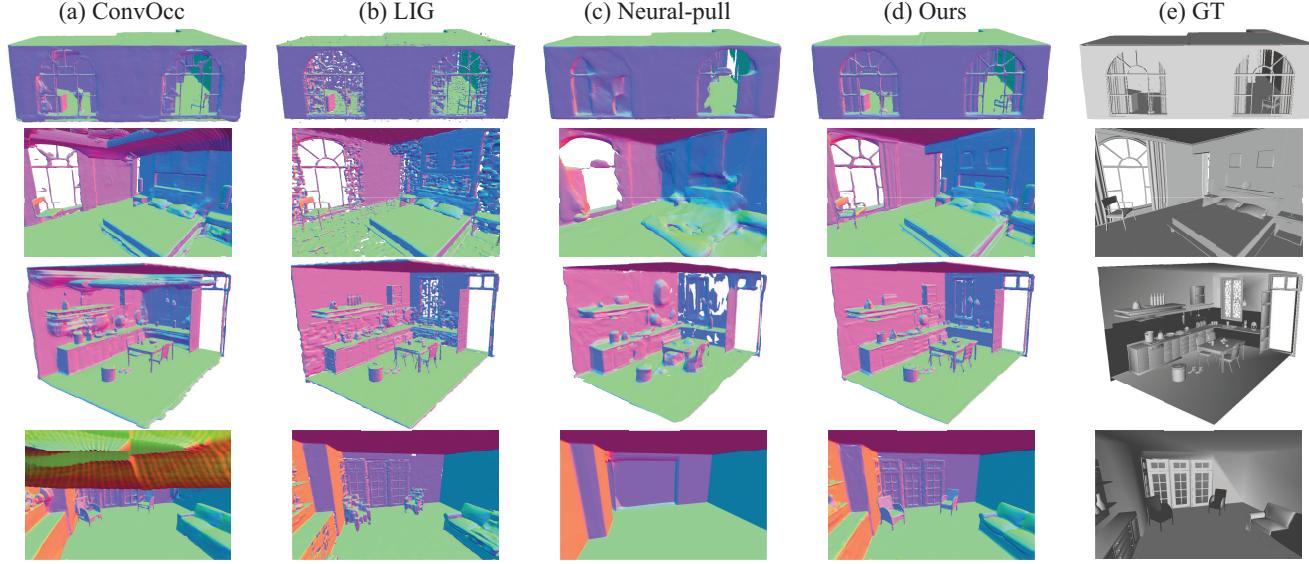


Figure 10. Visual comparison with the state-of-the-art under SceneNet dataset, where normal map is also shown.

to different numbers of parts, such as $\{0^3, 4^3, 6^3, 8^3\}$, and then, use each set of parts to learn the local context prior in Fig. 3 (a). The comparison shown in Tab. 13 demonstrates that it is hard to learn the prior well if the size of t_i is too large, such as “ 0^3 ” and “ 4^3 ”, since t_i is too complex to learn. While it is also hard to learn some meaningful prior if the size of t_i is too small, such as “ 8^3 ”. In addition, we found that the overlap between neighboring local regions does not contribute to the performance increasing under 6^3 , such as “Lap” and “self”. We also explore whether we can learn a more meaningful prior by using patch-wise t_i in training. We form each t_i using 1000, 2000, or 4000 neighbors in terms of geodesic distance. The results of “G1”, “G2” and “G4” show that patch-wise regions t_i do not work better than the part-wise t_i that we are using.

We highlight our advantage by learning the local context prior using local regions t_i merely from the reconstruction target G . Although “Self” is obtained with much fewer training regions, it achieves almost the same result as “ 6^3 ” which is obtained by learning the local context prior from all local regions across different training shapes. This advantage comes from our ability of flexibly searching over the whole prior space, which alleviates the necessity of learning a high quality local context prior. However, the

optimization can converge faster if more local regions t_i are used in learning as shown by the loss curve comparison in Fig. 11.

	0^3	4^3	6^3	8^3	Lap	Self	G1	G2	G4
L2-CD	12.06	4.43	2.09	2.58	2.09	2.09	2.9	2.3	2.5
Normal	0.904	0.922	0.945	0.940	0.944	0.942	0.939	0.941	0.940

Table 13. Effect of region size under ABC. L2-CD×1000.

Limitation. Although we achieve high reconstruction accuracy, we require further optimization during testing. This takes more time than methods [12] leveraging pretrained models for inference.

5. Conclusion

We propose to reconstruct surfaces from point clouds by learning implicit functions as a predictive context prior. Our method successfully specializes the learned local context prior into predictive context prior for a specific point cloud, which effectively searches the reconstruction prior across the whole prior space without focusing on some specific locations. This advantage significantly increases our ability of leveraging the learned prior, which makes the learned prior generalize to as many unseen target regions as possible. Our idea is justified by our experimental results which outperform the state-of-the-art in terms of various metrics under widely used benchmarks.

864

865

866

867

868

869

870

871

872

873

874

875

876

877

878

879

880

881

882

883

884

885

886

887

888

889

890

891

892

893

894

895

896

897

898

899

900

901

902

903

904

905

906

907

908

909

910

911

912

913

914

915

916

917

References

- [1] Matan Atzmon and Yaron Lipman. Sal: Sign agnostic learning of shapes from raw data. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2020. 1, 2
- [2] Matan Atzmon and yaron Lipman. SALD: sign agnostic learning with derivatives. In *International Conference on Learning Representations*, 2021. 2, 5
- [3] Jan Bednarik, Shaifali Parashar, Erhan Gundogdu, and Pascal Salzmann, Mathieu andFua. Shape reconstruction by learning differentiable surface representations. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2020. 2
- [4] Yizhak Ben-Shabat, Chamin Hewa Koneputugodage, and Stephen Gould. Digs : Divergence guided shape implicit neural representation for unoriented point clouds. *CoRR*, abs/2106.10811, 2021. 2
- [5] F. Bernardini, J. Mittleman, H. Rushmeier, C. Silva, and G. Taubin. The ball-pivoting algorithm for surface reconstruction. *IEEE Transactions on Visualization and Computer Graphics*, 5(4):349–359, 1999. 2, 5
- [6] Rohan Chabra, Jan Eric Lenssen, Eddy Ilg, Tanner Schmidt, Julian Straub, Steven Lovegrove, and Richard A. Newcombe. Deep local shapes: Learning local SDF priors for detailed 3D reconstruction. In *European Conference on Computer Vision*, volume 12374, pages 608–625, 2020. 1, 2, 5, 6
- [7] 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. Technical Report arXiv:1512.03012 [cs.GR], Stanford University — Princeton University — Toyota Technological Institute at Chicago, 2015. 5
- [8] Zhiqin Chen and Hao Zhang. Learning implicit fields for generative shape modeling. *IEEE Conference on Computer Vision and Pattern Recognition*, 2019. 2, 5
- [9] Julian Chibane, Aymen Mir, and Gerard Pons-Moll. Neural unsigned distance fields for implicit function learning. *arXiv*, 2010.13938, 2020. 2, 5
- [10] 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 *European Conference on Computer Vision*, pages 628–644, 2016. 5
- [11] Thomas Davies, Derek Nowrouzezahrai, and Alec Jacobson. On the effectiveness of weight-encoded neural implicit 3d shapes. *CoRR*, abs/2009.09808, 2021. 2
- [12] 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*, 2020. 1, 2, 5, 6, 8
- [13] Haoqiang Fan, Hao Su, and Leonidas J. Guibas. A point set generation network for 3D object reconstruction from a single image. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 2463–2471, 2017. 5
- [14] Matheus Gadelha, Rui Wang, and Subhransu Maji. Shape reconstruction using differentiable projections and deep priors. In *International Conference on Computer Vision*, 2019. 2
- [15] Matheus Gadelha, Rui Wang, and Subhransu Maji. Deep manifold prior. *CoRR*, abs/2004.04242, 2020. 2
- [16] Kyle Genova, Forrester Cole, Avneesh Sud, Aaron Sarna, and Thomas Funkhouser. Local deep implicit functions for 3d shape. In *IEEE Conference on Computer Vision and Pattern Recognition*, June 2020. 2
- [17] Amos Gropp, Lior Yariv, Niv Haim, Matan Atzmon, and Yaron Lipman. Implicit geometric regularization for learning shapes. *arXiv*, 2002.10099, 2020. 1, 2, 6
- [18] Thibault Groueix, Matthew Fisher, Vladimir G. Kim, Bryan C. Russell, and Mathieu Aubry. A papier-mch approach to learning 3D surface generation. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2018. 2, 5, 6
- [19] Zhizhong Han, Chao Chen, Yu-Shen Liu, and Matthias Zwicker. DRWR: A differentiable renderer without rendering for unsupervised 3D structure learning from silhouette images. In *International Conference on Machine Learning*, 2020. 2
- [20] Zhizhong Han, Chao Chen, Yu-Shen Liu, and Matthias Zwicker. Drwr: A differentiable renderer without rendering for unsupervised 3D structure learning from silhouette images. In *International Conference on Machine Learning*, 2020. 2
- [21] Zhizhong Han, Guanhui Qiao, Yu-Shen Liu, and Matthias Zwicker. SeqXY2SeqZ: Structure learning for 3D shapes by sequentially predicting 1D occupancy segments from 2D coordinates. In *European Conference on Computer Vision*, 2020. 2
- [22] Zhizhong Han, Mingyang Shang, Yu-Shen Liu, and Matthias Zwicker. View Inter-Prediction GAN: Unsupervised representation learning for 3D shapes by learning global shape memories to support local view predictions. In *AAAI*, pages 8376–8384, 2019. 2, 4
- [23] Zhizhong Han, Xiyang Wang, Yu-Shen Liu, and Matthias Zwicker. Multi-angle point cloud-vae:unsupervised feature learning for 3D point clouds from multiple angles by joint self-reconstruction and half-to-half prediction. In *IEEE International Conference on Computer Vision*, 2019. 2
- [24] A. Handa, V. Patraucean, V. Badrinarayanan, S. Stent, and R. Cipolla. Understanding realworld indoor scenes with synthetic data. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 4077–4085, 2016. 5
- [25] Rana Hanocka, Gal Metzger, Raja Giryes, and Daniel Cohen-Or. Point2mesh: A self-prior for deformable meshes. *ACM Trans. Graph.*, 39(4), 2020. 2
- [26] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 770–778, 2016. 3
- [27] Tao Hu, Zhizhong Han, Abhinav Shrivastava, and Matthias Zwicker. Render4Completion: Synthesizing multi-view depth maps for 3D shape completion. *ArXiv*, abs/1904.08366, 2019. 2

- 972 [28] Tao Hu, Zhizhong Han, and Matthias Zwicker. 3D shape
973 completion with multi-view consistent inference. In *AAAI*,
974 2020. 2
- 975 [29] Eldar Insafutdinov and Alexey Dosovitskiy. Unsupervised
976 learning of shape and pose with differentiable point clouds.
977 In *Advances in Neural Information Processing Systems*,
978 pages 2807–2817, 2018. 2
- 979 [30] Meng Jia and Matthew Kyan. Learning occupancy function
980 from point clouds for surface reconstruction. *arXiv*,
981 2010.11378, 2020. 2
- 982 [31] Chiyu Jiang, Avneesh Sud, Ameesh Makadia, Jingwei
983 Huang, Matthias Nießner, and Thomas Funkhouser. Local
984 implicit grid representations for 3D scenes. In *IEEE Conference
985 on Computer Vision and Pattern Recognition*, 2020. 1, 2,
986 4, 5, 6
- 987 [32] Yue Jiang, Dantong Ji, Zhizhong Han, and Matthias Zwicker.
988 SDFDiff: Differentiable rendering of signed distance fields
989 for 3D shape optimization. In *IEEE Conference on Computer
990 Vision and Pattern Recognition*, 2020. 2
- 991 [33] Michael M. Kazhdan and Hugues Hoppe. Screened poisson
992 surface reconstruction. *ACM Transactions Graphics*,
993 32(3):29:1–29:13, 2013. 2, 5, 6
- 994 [34] Sebastian Koch, Albert Matveev, Zhongshi Jiang, Francis
995 Williams, Alexey Artemov, Evgeny Burnaev, Marc Alexa,
996 Denis Zorin, and Daniele Panozzo. ABC: A big cad model
997 dataset for geometric deep learning. In *IEEE Conference on
998 Computer Vision and Pattern Recognition*, June 2019. 5
- 999 [35] Yiyi Liao, Simon Donné, and Andreas Geiger. Deep marching
1000 cubes: Learning explicit surface representations. In *Conference
1001 on Computer Vision and Pattern Recognition*, 2018. 5
- 1002 [36] Minghua Liu, Xiaoshuai Zhang, and Hao Su. Meshing point
1003 clouds with predicted intrinsic-extrinsic ratio guidance. In
1004 *European Conference on Computer vision*. 2, 5
- 1005 [37] Shi-Lin Liu, Hao-Xiang Guo, Hao Pan, Pengshuai Wang, X-
1006 in Tong, and Yang Liu. Deep implicit moving least-squares
1007 functions for 3D reconstruction. In *IEEE Conference on
1008 Computer Vision and Pattern Recognition*, 2021. 2, 6
- 1009 [38] Xinhai Liu, Zhizhong Han, Yu-Shen Liu, and Matthias
1010 Zwicker. Point2Sequence: Learning the shape representation
1011 of 3D point clouds with an attention-based sequence to
1012 sequence network. In *AAAI*, pages 8778–8785, 2019. 2
- 1013 [39] S. Lombardi, M. R. Oswald, and M. Pollefeys. Scalable point
1014 cloud-based reconstruction with local implicit functions. In
1015 *2020 International Conference on 3D Vision (3DV)*, pages
997–1007, 2020. 2
- 1016 [40] William E. Lorensen and Harvey E. Cline. Marching cubes:
1017 A high resolution 3D surface construction algorithm. *Computer
1018 Graphics*, 21(4):163–169, 1987. 1, 2, 3
- 1019 [41] Yiming Luo, Zhenxing Mi, and Wenbing Tao. Deepdt:
1020 Learning geometry from delaunay triangulation for surface
1021 reconstruction. *CoRR*, abs/2101.10353, 2020. 2, 5
- 1022 [42] Baorui Ma, Zhizhong Han, Yu-Shen Liu, and Matthias
1023 Zwicker. Neural-pull: Learning signed distance functions
1024 from point clouds by learning to pull space onto surfaces. In
1025 *International Conference on Machine Learning*, 2021. 2, 3,
5, 6
- 1026 [43] Julien N. P. Martel, David B. Lindell, Connor Z. Lin, Eric R.
1027 Chan, Marco Monteiro, and Gordon Wetzstein. ACORN:
1028 adaptive coordinate networks for neural scene representa-
1029 tion. *CoRR*, abs/2105.02788, 2021. 2
- 1030 [44] Lars Mescheder, Michael Oechsle, Michael Niemeyer,
1031 Sebastian Nowozin, and Andreas Geiger. Occupancy networks:
1032 Learning 3D reconstruction in function space. In *IEEE Con-
1033 ference on Computer Vision and Pattern Recognition*, 2019.
1034 2, 4, 5
- 1035 [45] Zhenxing Mi, Yiming Luo, and Wenbing Tao. Ssrnet: Scal-
1036 able 3D surface reconstruction network. In *IEEE Conference
1037 on Computer Vision and Pattern Recognition*, June 2020. 2,
5
- 1038 [46] Ben Mildenhall, Pratul P. Srinivasan, Matthew Tancik,
1039 Jonathan T. Barron, Ravi Ramamoorthi, and Ren Ng. Nerf:
1040 Representing scenes as neural radiance fields for view syn-
1041 thesis. In *European Conference on Computer Vision*, 2020.
1042 2
- 1043 [47] Michael Oechsle, Songyou Peng, and Andreas Geiger. Uni-
1044 nisurf: Unifying neural implicit surfaces and radiance fields
1045 for multi-view reconstruction. In *International Conference
1046 on Computer Vision*, 2021. 2
- 1047 [48] Yutaka Ohtake, Alexander G. Belyaev, Marc Alexa, Greg
1048 Turk, and Hans-Peter Seidel. Multi-level partition of unity
1049 implicits. *ACM Transactions on Graphics*, 22(3):463–470,
1050 2003. 6
- 1051 [49] Jeong Joon Park, Peter Florence, Julian Straub, Richard
1052 Newcombe, and Steven Lovegrove. DeepSDF: Learning
1053 continuous signed distance functions for shape represen-
1054 tation. In *IEEE Conference on Computer Vision and Pattern
1055 Recognition*, 2019. 2, 4, 5, 6, 7
- 1056 [50] Songyou Peng, Chiyu "Max" Jiang, Yiyi Liao, Michael
1057 Niemeyer, Marc Pollefeys, and Andreas Geiger. Shape as
1058 points: A differentiable poisson solver. In *Advances in Neu-
1059 ral Information Processing Systems*, 2021. 2
- 1060 [51] Charles R. Qi, Hao Su, Kaichun Mo, and Leonidas J. Guibas.
1061 PointNet: Deep learning on point sets for 3D classification
1062 and segmentation. In *IEEE Conference on Computer Vision
1063 and Pattern Recognition*, 2017. 2, 3
- 1064 [52] Charles Ruizhongtai Qi, Li Yi, Hao Su, and Leonidas J.
1065 Guibas. PointNet++: Deep hierarchical feature learning on
1066 point sets in a metric space. In *Advances in Neural Infor-
1067 mation Processing Systems*, pages 5105–5114, 2017. 2
- 1068 [53] Konstantinos Rematas, Ricardo Martin-Brualla, and Vittorio
1069 Ferrari. Sharf: Shape-conditioned radiance fields from a sin-
1070 gle view. In *International Conference on Machine Learning*,
1071 2021. 2
- 1072 [54] Jonathan Sauder and Bjarne Sievers. Self-supervised deep
1073 learning on point clouds by reconstructing space. In *Ad-
1074 vances in Neural Information Processing Systems*, pages
1075 12962–12972, 2019. 2
- 1076 [55] Yawar Siddiqui, Justus Thies, Fangchang Ma, Qi Shan,
1077 Matthias Nießner, and Angela Dai. Retrievalfuse: Neural 3d
1078 scene reconstruction with a database. In *Proceedings of the
1079 International Conference on Computer Vision (ICCV)*, 2021.
2

- 1080 neural representations with periodic activation functions. In
1081 *Advances in Neural Information Processing Systems*, 2020.
1082 2
- 1083 [57] Lars Mescheder Marc Pollefeys Andreas Geiger Songyu
1084 Peng, Michael Niemeyer. Convolutional occupancy net-
1085 works. In *European Conference on Computer Vision*, 2020.
1086 2, 6
- 1087 [58] Towaki Takikawa, Joey Litalien, Kangxue Yin, Karsten
1088 Kreis, Charles Loop, Derek Nowrouzezahrai, Alec Jacobson,
1089 Morgan McGuire, and Sanja Fidler. Neural geometric level
1090 of detail: Real-time rendering with implicit 3D shapes. In
1091 *IEEE Conference on Computer Vision and Pattern Recog-
1092 nition*, 2021. 2
- 1093 [59] Matthew Tancik, Pratul P. Srinivasan, Ben Mildenhall, Sara
1094 Fridovich-Keil, Nithin Raghavan, Utkarsh Singhal, Ravi Ra-
1095 mamoorthi, Jonathan T. Barron, and Ren Ng. Fourier fea-
1096 tures let networks learn high frequency functions in low di-
1097 mensional domains. *NeurIPS*, 2020. 2
- 1098 [60] Jiapeng Tang, Jiabao Lei, Dan Xu, Feiying Ma, Kui Jia,
1099 and Lei Zhang. Sa-convolnet: Sign-agnostic optimization of
1100 convolutional occupancy networks. In *Proceedings of the
1101 IEEE/CVF International Conference on Computer Vision*,
2021. 2
- 1102 [61] Maxim Tatarchenko, Stephan R. Richter, Rene Ranftl,
1103 Zhuwen Li, Vladlen Koltun, and Thomas Brox. What do
1104 single-view 3D reconstruction networks learn? In *The IEEE
1105 Conference on Computer Vision and Pattern Recognition*,
2019. 5
- 1106 [62] Edgar Treitschke, Ayush Tewari, Vladislav Golyanik, Michael
1107 Zollhöfer, Carsten Stoll, and Christian Theobalt. PatchNets:
1108 Patch-Based Generalizable Deep Implicit 3D Shape Re-
1109 presentations. *European Conference on Computer Vision*, 2020.
1110 1, 2
- 1111 [63] Bram Wallace and Bharath Hariharan. Few-shot generaliza-
1112 tion for single-image 3d reconstruction via priors. In *IEEE
1113 International Conference on Computer Vision*, pages 3817–
1114 3826. 2
- 1115 [64] Xin Wen, Tianyang Li, Zhizhong Han, and Yu-Shen Liu.
1116 Point cloud completion by skip-attention network with hi-
1117 erarchical folding. In *IEEE Conference on Computer Vision
1118 and Pattern Recognition*, 2020. 2
- 1119 [65] Francis Williams, Teseo Schneider, Claudio Silva, Denis
1120 Zorin, Joan Bruna, and Daniele Panozzo. Deep geomet-
1121 ric prior for surface reconstruction. In *IEEE Conference on
1122 Computer Vision and Pattern Recognition*, 2019. 2, 5
- 1123 [66] Jiajun Wu, Chengkai Zhang, Xiuming Zhang, Zhoutong
1124 Zhang, William T. Freeman, and Joshua B. Tenenbaum.
1125 Learning shape priors for single-view 3d completion and re-
1126 construction. In *European Conference on Computer Vision*,
1127 pages 673–691, 2018. 2
- 1128 [67] Mingyue Yang, Yuxin Wen, Weikai Chen, Yongwei Chen,
1129 and Kui Jia. Deep optimized priors for 3d shape modeling
1130 and reconstruction, 2020. 2
- 1131 [68] Wang Yifan, Shihao Wu, Cengiz Oztireli, and Olga Sorkine-
1132 Hornung. Iso-points: Optimizing neural implicit surfaces
1133 with hybrid representations. *CoRR*, abs/2012.06434, 2020.
2
- 1134
- 1135
- 1136
- 1137
- 1138
- 1139
- 1140
- 1141
- 1142
- 1143
- 1144
- 1145
- 1146
- 1147
- 1148
- 1149
- 1150
- 1151
- 1152
- 1153
- 1154
- 1155
- 1156
- 1157
- 1158
- 1159
- 1160
- 1161
- 1162
- 1163
- 1164
- 1165
- 1166
- 1167
- 1168
- 1169
- 1170
- 1171
- 1172
- 1173
- 1174
- 1175
- 1176
- 1177
- 1178
- 1179
- 1180
- 1181
- 1182
- 1183
- 1184
- 1185
- 1186
- 1187