Learning Bijective Surface Parameterization for Inferring Signed Distance Functions from Sparse Point Clouds with Grid Deformation

Anonymous CVPR submission

Paper ID 11027

Abstract

Inferring signed distance functions (SDFs) from sparse point clouds remains a challenge in surface reconstruction. The key lies in the lack of detailed geometric information in sparse point clouds, which is essential for learning a continuous field. To resolve this issue, we present a novel approach that learns a dynamic deformation network to predict SDFs in an end-to-end manner. To parameterize a continuous surface from sparse points, we propose a bijective surface parameterization (BSP) that learns the global shape from local patches. Specifically, we construct a bijective mapping for sparse points from the parametric domain to 3D local patches, integrating patches into the global surface. Meanwhile, we introduce grid deformation optimization (GDO) into the surface approximation to optimize the deformation of grid points and further refine the parametric surfaces. Experimental results on synthetic and real scanned datasets demonstrate that our method significantly outperforms the current state-of-the-art methods.

1. Introduction

Surface reconstruction from 3D point clouds is an important task in 3D computer vision. Continuous surfaces are widely used in downstream applications, such as autonomous driving, VR, and robotics. With the development of deep learning [2, 16, 17, 38, 43], significant breakthroughs have been made in learning signed distance functions (SDFs) to represent continuous surfaces [12, 21]. The SDFs learned from dense point clouds are continuous and complete, which allow us to obtain robust isosurfaces of discrete scalar fields using the Marching-cubes algorithm [27]. However, when confronted with sparse point clouds, current approaches fail to accurately predict a signed distance field around the surface, impacting their performance on the real-world scenario where only sparse point clouds are available.

Previous works [13, 19, 34, 39] which infer the SDFs from raw point clouds typically require ground truth signed distances or dense point clouds as supervision. With sparse point clouds, current state-of-the-art methods [3] obtain shape priors from large scale supervision to handle the sparsity. Although prior-based methods can leverage data-driven information to infer SDFs with simple topological structures, they still struggle to deal with real-world diverse sparse inputs containing complex geometries. Some methods learn self-supervised up-sampling priors from sparse point clouds to maintain the shape integrity [8, 36]. However, these approaches still do not work well with sparse points, making it remain challenging to recover complete geometry.

To address this issue, we propose Bijective Surface Parameterization (BSP) for learning a continuous global surface. Unlike previous approaches, we construct a continuous bijective mapping between the canonical spherical parametric domain and the 3D space. For each 3D point, we transform it into a code on the sphere manifold in the parameter space, and then regard the code as a center to densify the patch it locates by sampling more codes on the sphere. With the learned BSP, we transform each densified patch in the parameter space back into 3D space, leading to a 3D shape with denser points. Compared to methods which directly upsample a global shape, we train local patches with a shared network to recover more details on patches.

Based on the densified points, we propose a Grid Deformation Optimization (GDO) strategy to estimate the SDFs. Our key insight is to utilize deformable tetrahedral grids to generate watertight shapes under the supervision of densified points. To this end, we sample uniformly distributed vertices from the generated shape to match the densified points. It allows us to progressively learn SDFs from coarse to fine.

Extensive experiments on widely-used benchmark datasets demonstrate that our method significantly outperforms the current state-of-the-art methods. Our contributions are summarized as follows:

- We propose a novel framework that learns the neural deformation network to infer signed distance fields from sparse points without additional surface priors.

- We demonstrate that learning bijective surface parameterization can parameterize the surface represented by sparse points, which introduces a novel way of sampling dense patches in the parameter space.
- We achieve state-of-the-art results in surface reconstruction on synthetic data and real scenes in widely used benchmarks, demonstrating the great potential in sparse reconstruction tasks.

2. Related Work

Surface reconstruction from 3D point clouds has made significant progress over the years[6, 25, 26, 32, 43]. Earlier optimization based methods infer continuous surfaces from the point cloud geometry. With the development of datasets[1, 15], deep learning methods achieved promising results. In particular, learning the neural implicit field (NIF) has been widely applied in various reconstruction tasks, including multiview reconstruction[22, 35], point cloud reconstruction[18, 29, 51], and occupancy estimation[33, 34, 37]. In the following section, we focus on implicit representation learning methods based on sparse point clouds.

Neural Implicit Surface Reconstruction. In recent years, a lot of advances have been made in 3D surface reconstruction tasks with NIF methods. NIF represents shapes in implicit functions using occupancy [10, 37, 43] or signed distance functions [30, 31, 53] and reconstructs surfaces with the Marching-cubes algorithm. Previous studies employ the global optimization based strategy [28, 42], embedding objects as latent codes to predict the NIF. Furthermore, to reconstruct finer local details, some methods use different training strategies to capture the local priors [11, 44]. In addition, some recent methods introduce new perspectives for learning NIF through differentiable Poisson solvers [38], iso-points [51], and grid interpolation[7]. However, these methods rely on dense point cloud inputs or real signed distance values and normals, which limits their ability to accurately predict NIF from sparse point clouds.

Learning Self-Priors from Sparse Points Clouds. Learning NIF from sparse point clouds without real supervision is a more intricate task. Onsurf [3] manages to understand sparse points by using pretrained priors. However, it is limited by the weak generalization ability for diverse inputs. Some studies focus on learning NIF from sparse point clouds. Ndrop[5] introduces a statistical strategy to learn the decision function for implicit occupancy fields with sample points. However, this method struggles to constrain all sample points accurately. Building on this, SparseOcc[36] utilizes a classifier to simplify the process of learning occupancy functions, which significantly improves the efficiency to learn occupancy field from sparse inputs. Despite these advancements, these methods only depend on sparse inputs as supervision, the learned decision functions tend to produce coarse approximations and fail to handle extremely sparse or complex inputs. Meanwhile, inferring smooth surfaces from occupancy fields still challenging. VIPSS[20] and SparseOcc attempt to address this issue through Hermite interpolation and entropy-based regularization. However, these approaches are sensitive to hyperparameters and lack general applicability.

Learning Parametric Surfaces from Sparse Point Clouds. Previous studies[45, 52] proved that learning surface parameterization can map across dimensions and naturally infer the geometry of shapes. Although TPS[8] explores its application to sparse point clouds, it is limited by sparse inputs and only learn parametric surface in global manner. TPS++[9] additionally introduces structure-aware distance constraints to enhance accuracy. However, it still struggles to learn the global geometry from local parametric surfaces. To address this issue, we propose bijective surface parameterization, which enables networks to learn multiple local parametric surfaces from the sparse point cloud and infer a finer global surface.

Dynamic Deformation Network. Neural deformation network [40, 41, 55] dynamically learns 3D shapes with arbitrary topological structures using differentiable mesh vertex grids. GET3D [14] explicitly learns 3D models through a differentiable decoder to obtain detailed 3D models. By contrast, DynoSurf [50] learns keyframe point clouds as templates and uses neural networks to predict movement steps to obtain time-series 3D models with arbitrary topological structures. However, these methods rely on dense point clouds or pre trained embeddings. Here, we explore the feasibility of learning a deformation network for sparse tasks from a new perspective. During training, we learn parametric surfaces from sparse point clouds as supervision and learn implicit fields through the deformation of tetrahedral vertices in a differential manner.

3. Method

3.1. Overview

Given a sparse point cloud $Q = \{q_n\}_{n=1}^N$, we aim to learn the signed distance field that represents a continuous surface from Q. An overview of the proposed method is shown in Fig. 1. We present the bijective surface parameterization (BSP) in Fig. 1 (a) to learn a continuous parametric surface representation. We first learn a canonical mapping Ψ to encode Q into the unit sphere parametric domain $\mathcal{U}$, where we can sample local patches $P = \{p_m\}_{m=1}^M$ around each point. Subsequently, we learn an inverse mapping Ψ to decode P back to 3D space and integrate patches into a global surface $S = \{s_i\}_{i=1}^I$. With S as supervision, we employ the grid deformation optimization (GDO) strategy to move the deformable grid points $V = \{v_j\}_{j=1}^J$ towards S to infer the SDFs shown in Fig. 1 (b). The target is to minimize the differences between S and Q. Moreover, We regulate

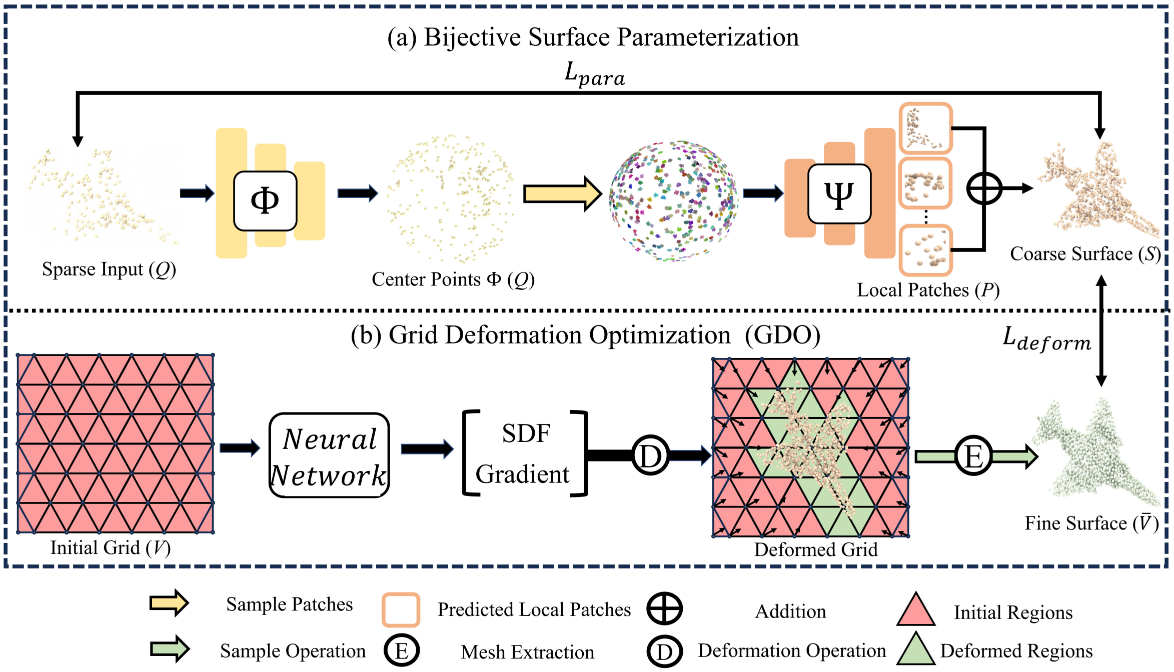

Figure 1. Overview of Our method. Given a sparse point cloud Q, we first learn a mapping function Φ to encode Q to a unit sphere parametric domain. We consider each point as center point and sample local patches on the parametric surface. Next, we learn the inverse mapping Ψ to predict the positions of these local patches in 3D space and integrate them to obtain S. We leverage S as the supervision for the grid deformation network g and predict the signed distance field through the GDO optimization strategy. We further extract dense point cloud $\bar{V}$ from the implicit field and optimize the parameterized surface S.

the deformation of V by constraining the distance between V and S to infer a continuous surface. To predict accurate SDFs, we also encourage V to be on the zero level set of the field. In this section, we will begin with introducing the bijective surface parameterisation (BSP). Subsequently, we will describe the grid deformation optimisation (GDO) strategy in the following.

3.2. Bijective Surface Parameterization

Previous methods [8] are constrained in representing continuous surfaces with multiple patches due to sparsity, which limits the completeness of the parametric surface. In contrast, we learn two mapping functions to achieve this: Φ maps each $q \in Q$ to the canonical parametric domain while Ψ conducts inverse mapping. We learn two mapping functions with an auto-encoder structure as follows.

Canonical Mapping Φ. For each point $q_n \in Q$, we first encode the point-wise feature $\Phi(q_n)$ based on the PointTransformer [46] layer Φ, which can be formulated as

$$\Phi(q_n) = \sum_{q_k=1}^{k} \rho\big(\gamma(\beta(q_n) - \eta(q_k) + \xi)\big) \odot (\alpha(q_k) + \xi), \quad (1)$$

where k is the set of k-nearest neighbors (KNN) of q_n, we set $k = 8$ by default. $\{\alpha, \beta, \eta\}$ are linear layers, $\{\gamma, \delta\}$ are MLP layers, ρ is the softmax function, position embeding $\xi = \delta(q_n - q_k)$, $\odot$ is point-wise product operation.

With the learned Φ, we extract features into the coordinates and project them in the canonical unit sphere $\mathcal{U}$, where $\mathcal{U}(q_n) \in \mathcal{U}(Q)$. Each $\mathcal{U}(q_n)$ is a center point where KNN is utilized to sample a local patch $\mathcal{U}(p_m)$ around it. Specifically, we construct a uniform sphere coplanar with $\mathcal{U}(Q)$ to provide samples for $\mathcal{U}(q_n)$.

Inverse Mapping Ψ. Similarly, we efficiently estimate $\Psi \approx \Phi^{-1}$ with an neural network. We utilize the standard transformer decoder block as Ψ, which receives point-wise features with several linear layers φ as the global condition, and local patch $\mathcal{U}(p_m)$ as queries to integrate a global shape S in 3D space, which can be formulated as

$$S = \sum_{m=1}^{M} \Psi(\varphi(\Phi(Q), \mathcal{U}(p_m))). \quad (2)$$

We measure the distance between the parameterized surface S and the sparse point cloud Q using Chamfer Distance (CD), denoted as L_{para}.

$$L_{para} = \frac{1}{I} \sum_{s \in S} \min_{q \in Q} \|s - q\|^2 + \frac{1}{N} \sum_{q \in Q} \min_{s \in S} \|q - s\|^2. \quad (3)$$

We visualize the BSP process in Fig. 2. For $q_n \in Q$, Φ maps its position in $\mathcal{U}$ and samples local patches $\mathcal{U}(p_m)$. With the inverse mapping Ψ, we generate a local surface on

S. We integrate all local patches to obtain the global shape as the coarse surface S.

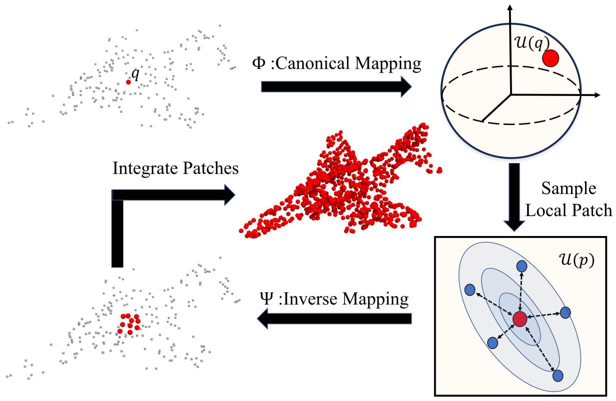

Figure 2. Illustration of BSP. The white points indicate the sparse input Q. For each point $q \in Q$, we learn mapping function Φ to map q to a local patch $\mathcal{U}(p)$ on the parametric surface. Subsequently, we employ an inverse mapping Ψ to assembles these patches into a global surface (red points).

3.3. Grid Deformation Optimization

With the learned BSP, we parameterize the coarse surface S. Naive implementations rely on S to infer SDFs and reconstruct surfaces, often producing holes due to the non-uniformity distribution. Unlike these methods, we design the grid deformation optimization strategy to learn continuous signed distance functions and further optimize parametric surface. Given tetrahedral grid points V, a straightforward strategy to update the deformed points V' is to learn an offset ε from neural network g, formulated as $V' = V + \varepsilon$. However, directly learning offsets from g fails to maintain consistency of deformation direction, resulting in difficulties in convergence. We maintain the consistency of the deformation by constraining on normals n_V with gradients $\nabla g(V)$. During training, we predict the SDFs $g(V;\theta)$ and the gradient $\nabla g(V)$ to guide the deformation process of V. We consider $g(V)$ and n_V to be the stride and direction, respectively. Therefore, the deformation process of V can be described as

$$V \rightarrow V' = \|g(V;\theta) \cdot n_V - V\|_2, \tag{4}$$

where θ is learnable parameter in deformation network g, $n_V = g(V;\theta)/\|\nabla g(V)\|_2$.

We further compare the movement directions and optimization results of GDO in Fig. 3(a) and the classical strategy [40] in Fig. 3(b). The red lines indicate the next deformation direction of the grid points. Compared to direct offset prediction, GDO achieves more consistent deformation directions, resulting in a more accurate shape learning. Meanwhile, we extract the surface using Deep Marching Tetrahedra (DMT), the operation denoted as $\mathrm{DMT}(\cdot)$. The deformation grid points $\bar{V}$ on the surface can be expressed as $\bar{V} = \mathrm{DMT}(V')$, where $\bar{V} = \{\bar{v}_t\}_{t=1}^{T}$. We use Chamfer Distance to regulate the deformation process of $\bar{V}$ and minimize the difference to S, denoted as L_{deform}, we have:

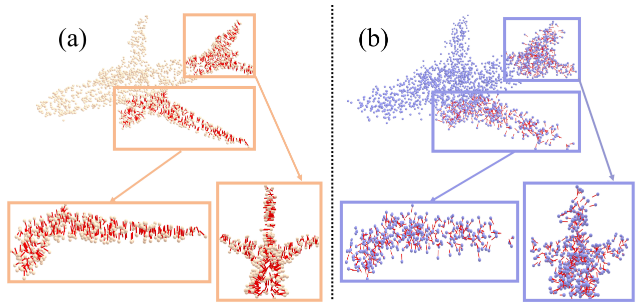

Figure 3. Visual comparison of GDO (a) and direct offset optimization (b), the red lines indicate the offset direction.

$$L_{deform} = \frac{1}{T}\sum_{\bar{v}\in\bar{V}} \min_{s\in S} \|\bar{v} - s\|^2 + \frac{1}{I}\sum_{s\in S} \min_{\bar{v}\in\bar{V}} \|s - \bar{v}\|^2. \tag{5}$$

To make the implicit field more accurate, we add the L_{surf} term to encourage the network to learn zero level set from $g(V)$. Formulated as:

$$L_{surf} = \mid g(V) \mid . \tag{6}$$

Therefore, the total loss L is given as:

$$L = \lambda_1 L_{para} + \lambda_2 L_{surf} + L_{deform}, \tag{7}$$

where λ_1 and λ_2 are weight parameters, which we set to 10 and 0.01 by default.

3.4. End-to-end Training

Existing self-supervised strategies [5, 36] struggle to accurately predict implicit fields from sparse point clouds. Here, we propose an effective framework to train our methods in an end-to-end manner. We first use BSP to map the sparse point cloud Q into a continuous parametric point cloud S, providing more precise supervision for GDO. Next, we leverage the neural network g to learn grid deformations to predict the implicit field. To further enhance the smoothness of the implicit field, GDO learns more consistent deformation directions from the gradients to improve overall details. Experimental results validate the effectiveness of our method.

4. Experiments

4.1. Experiment Setup

Datasets and Metrics. We adopt five synthetic and real scanned datasets to evaluate our method. We first compare the performance of our method on D-FAUST [4], SRB [48],

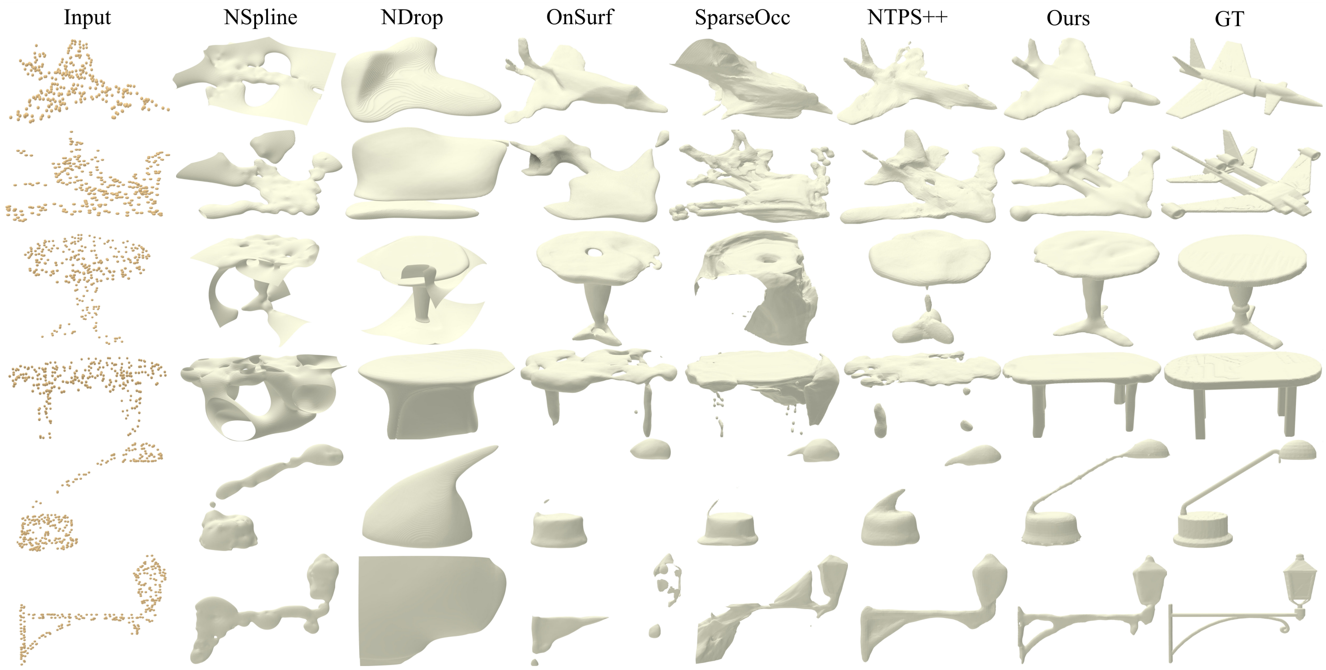

Figure 4. Visual comparison on ShapeNet. The input contains 300 points.

Class	Nspline	NP	NDrop	Onsurf	SparseOcc	NTPS	NTPS++	Ours
Plane	0.119	0.141	0.499	0.153	0.219	0.095	0.088	**0.072**
Chair	0.306	0.196	0.395	0.316	0.183	0.197	0.195	**0.142**
Cabinet	0.181	0.163	0.229	0.244	0.220	0.138	0.137	**0.105**
Display	0.193	0.145	0.287	0.204	0.091	0.127	0.122	**0.099**
Vessel	0.134	0.116	0.488	0.128	0.158	0.104	0.101	**0.080**
Table	0.318	0.400	0.426	0.288	0.261	0.225	0.215	**0.108**
Lamp	0.213	0.162	0.554	0.229	0.192	0.120	0.112	**0.077**
Sofa	0.168	0.139	0.259	0.147	0.178	0.125	0.129	**0.116**
Mean	0.206	0.183	0.392	0.214	0.187	0.141	0.137	**0.099**

Table 1. Reconstruction accuracy under ShapeNet in terms of $\text{CD}_{L1} \times 10$.

Class	Nspline	NP	NDrop	Onsurf	SparseOcc	NTPS	NTPS++	Ours
Plane	0.127	0.036	0.755	0.112	0.165	0.030	0.026	**0.022**
Chair	0.247	0.174	0.532	0.448	0.162	0.149	0.140	**0.115**
Cabinet	0.064	0.086	0.245	0.171	0.178	0.050	0.050	**0.046**
Display	0.095	0.099	0.401	0.153	0.081	0.083	**0.078**	**0.078**
Vessel	0.066	0.074	0.844	0.066	0.073	0.051	0.046	**0.042**
Table	0.312	0.892	0.701	0.419	0.415	0.272	0.264	**0.188**
Lamp	0.183	0.144	1.071	0.351	0.466	0.051	0.047	**0.043**
Sofa	0.053	0.072	0.463	0.066	0.010	0.056	0.062	**0.052**
Mean	0.143	0.197	0.627	0.223	0.193	0.093	0.089	**0.073**

Table 2. Reconstruction accuracy under ShapeNet in terms of $\text{CD}_{L2} \times 100$.

Class	Nspline	NP	NDrop	Onsurf	SparseOcc	NTPS	NTPS++	Ours
Plane	0.895	0.897	0.819	0.864	0.853	0.899	0.912	**0.913**
Chair	0.759	0.861	0.777	0.813	0.844	0.863	0.873	**0.896**
Cabinet	0.840	0.888	0.843	0.787	0.813	0.898	0.897	**0.904**
Display	0.830	0.909	0.873	0.855	0.872	0.924	**0.936**	0.927
Vessel	0.842	0.880	0.838	0.879	0.841	0.908	**0.913**	0.911
Table	0.771	0.835	0.795	0.827	0.856	0.877	0.888	**0.890**
Lamp	0.814	0.887	0.828	0.858	0.883	0.902	0.910	**0.914**
Sofa	0.828	0.905	0.808	0.881	0.870	0.919	0.915	**0.923**
Mean	0.822	0.883	0.823	0.845	0.854	0.899	0.905	**0.909**

Table 3. Reconstruction accuracy under ShapeNet in terms of NC.

and ShapeNet [1], following the Ndrop and NTPS. To verify the applicability of the method under extremely sparse conditions, we follow NTPS to randomly sample 300 points for each shape as the input for ShapeNet and D-FAUST. For fair comparison, we trained SparseOcc according to the default settings with open source code. For SRB dataset, we follow SparseOcc [36] to sample 1024 points for comparison. To further validate the effectiveness of in real large-scale scenarios, we validate our method on the 3DScene [54] and KITTI [15]. For the 3DScene dataset, we follow previous methods to randomly sample 100 points $/m^2$. For the KITTI dataset, we use point clouds in single frames to conduct a comparison.

We leverage L1 and L2 Chamfer Distance (CD_{L1}, CD_{L2}), Normal Consistency (NC) and Hausdorff Distance (HD) as evaluation metrics. For the shape and scene surface reconstruction, we sample $100k$ and $1000k$ points from the reconstructed and ground truth surfaces to calculate the errors.

4.2. Surface Reconstruction On Shapes

ShapeNet. We compare our method with Nspline[49], NP[28], NDrop, Onsurf, SparseOcc, NTPS++ and NTPS. The comparison results for different metrics are reported in Tab. 1, Tab. 2 and Tab. 3, where our method achieves the best results across all classes. We further present the visual comparison in Fig. 4. Ndrop and Nspline fail to generate accurate shape surfaces from sparse input, while NTPS++ and Onsurf generate correct shapes but with larger errors.

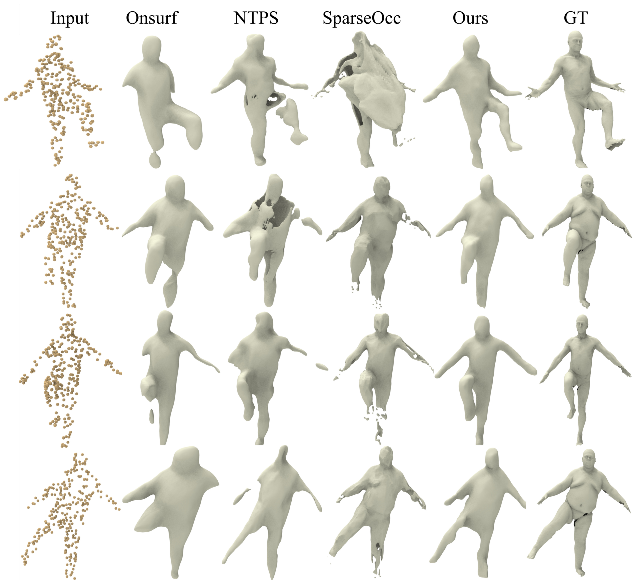

Figure 5. Visual comparison on D-FAUST. The input contains 300 points.

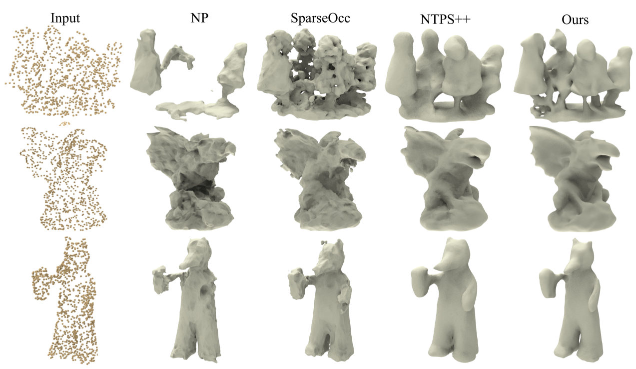

Figure 6. Visual comparison on SRB. The input contains 1024 points.

Methods	$CD_{L2} \times 100$			NC
	5%	50%	95%	
VIPSS	0.518	4.327	9.383	0.890
NDrop	0.126	1.000	7.404	0.792
NP	0.018	0.032	0.283	0.877
Nspline	0.037	0.080	0.368	0.808
SAP	0.014	0.024	0.071	0.852
SparseOcc	0.012	0.019	0.034	0.870
OnSurf	0.015	0.037	0.123	0.908
NTPS	0.012	0.160	0.022	0.909
Ours	**0.007**	**0.133**	**0.019**	**0.914**

Table 4. Reconstruction accuracy under DFAUST in terms of $CD_{L2} \times 100$ and NC.

Methods	$CD_{L1} \times 100$	HD
PSR	2.27	21.1
NTPS	0.73	7.78
NP	0.58	8.90
NTPS++	0.66	7.30
SparseOcc	0.49	6.04
Ours	**0.41**	**5.66**

Table 5. Reconstruction accuracy under SRB in terms of $CD_{L1} \times 100$ and HD.

		PSR	NP	Ndrop	NTPS	SparseOcc	Ours
Burghers	CD_{L1}	0.178	0.064	0.200	0.055	0.022	**0.015**
	CD_{L2}	0.205	0.008	0.114	0.005	**0.001**	**0.001**
	NC	0.874	0.898	0.825	0.909	0.871	**0.890**
Copyroom	CD_{L1}	0.225	0.049	0.168	0.045	0.041	**0.037**
	CD_{L2}	0.286	0.005	0.063	0.003	0.012	**0.003**
	NC	0.861	0.828	0.696	0.892	0.812	**0.897**
Lounge	CD_{L1}	0.280	0.133	0.156	0.129	0.021	**0.012**
	CD_{L2}	0.365	0.038	0.050	0.022	**0.001**	**0.001**
	NC	0.869	0.847	0.663	0.872	0.870	**0.903**
Stonewall	CD_{L1}	0.300	0.060	0.150	0.054	0.028	**0.021**
	CD_{L2}	0.480	0.005	0.081	0.004	0.003	**0.002**
	NC	0.866	0.910	0.815	0.939	0.931	**0.937**
Totempole	CD_{L1}	0.588	0.178	0.203	0.103	0.026	**0.022**
	CD_{L2}	1.673	0.024	0.139	0.017	**0.001**	**0.001**
	NC	0.879	0.908	0.844	0.935	**0.936**	0.931

Table 6. CD_{L1}, CD_{L2} and NC comparison under 3DScene.

SparseOcc cannot rely on decision boundaries to accurately predict occupancy fields under extremely sparse input conditions, making it challenging to reconstruct complex geometries. In contrast, our method produces more complete and smoother surfaces.

DFAUST. As shown in Tab. 4, we follow Ndrop to report the 5%, 50% and 95% of CD_{L1}, CD_{L2}, and NC results on the DFAUST dataset, achieving the best performance across all metrics. Additionally, we present a visual comparison with Onsurf, SparseOcc and NTPS in Fig. 5. Our method generates more complete human body with different poses.

SRB. We report the results on the real scanned dataset SRB in Tab. 5 and present a visual comparison in Fig. 6. All baseline methods reconstruct coarse surfaces with input of 1024 points. In contrast, our method not only reconstructs the complete shape but also recovers more local details.

4.3. Surface Reconstruction On Scenes

3DScene. We compare our method with the current state-of-the-art methods, including PSR[23], NP, Ndrop, SparseOcc, NTPS on the 3DScene dataset. The extensive results presented in Tab. 6 demonstrate that our method performs well in real-world scenarios. As shown in Fig. 7, our method reconstructs smoother surfaces and captures more internal details than NP and SparseOcc.

KITTI. We make a visual comparision the performance of our method with IMLS[47], Ndrop, PSR, SAP, NTPS and SparseOcc on real scanned large-scale street and local pedestrians on the KITTI dataset. Due to the lack of ground truth models, we provide the visual comparison in Fig. 9 and Fig. 8. Decision boundary based methods like SparseOcc only capture the shape's edges. In contrast, parameterized methods excel at reconstructing continuous sur-

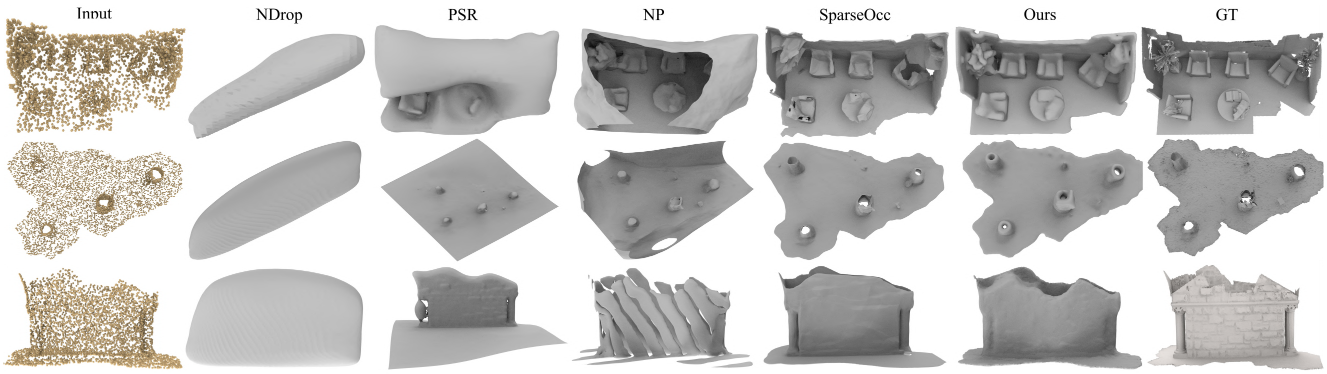

Figure 7. Visual comparison on 3DScene. The input contains 100 points $/m^2$.

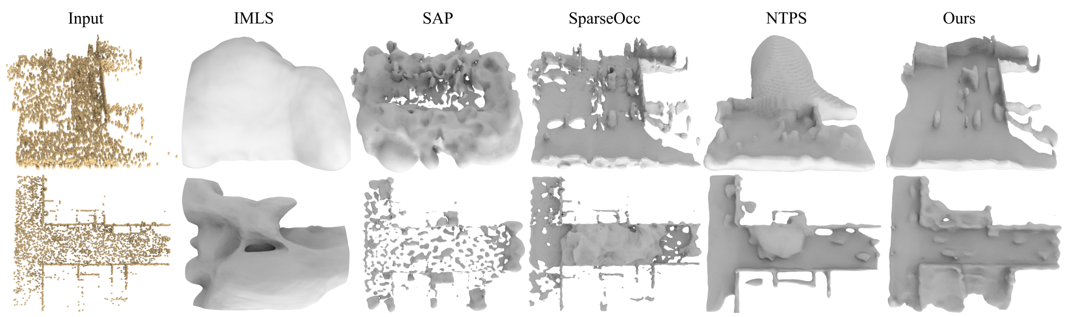

Figure 8. Visual comparison on KITTI-street.

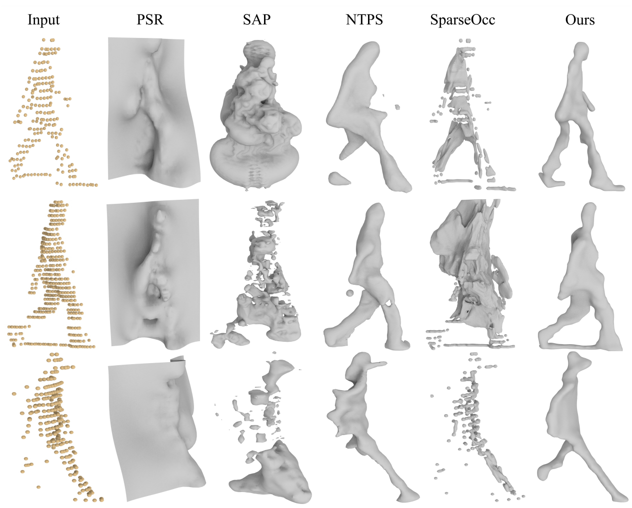

Figure 9. Visual comparison under KITTI-pedestrians.

faces. Our method can reconstruct more complete and detailed surfaces, such as diverse human poses and complex street scenes.

5. Ablation Studies

To validate the effectiveness of each module, we conduct ablation experiments on the lamp class of ShapeNet dataset. We present the quantitative results and visualization under different experimental settings.

Effect of BSP. To evaluate the effectiveness of the BSP, we firstly remove the BSP and only rely on sparse input to infer signed distance functions (denoted as Sparse), which lead a significant increase in CD error. It indicates that the parameterized supervision generated by BSP has a substantial impact on reconstruction accuracy. Next, we replace the BSP with the parameterization strategies proposed by TPS [8] and Atlas [52], which denoted as Single and Multiple, respectively. As shown in Tab. 7, both Single and Multiple lead to an increase in CD error. We additionally compared the CD error maps of point clouds predicted by different parameterization methods in Fig. 10 . Notably, single based parameterization (such as TPS) only generate a coarse global surface. Meanwhile, the multi-part parameterization strategy based on AtlasNet exhibites truncation and overfitting in local regions. In contrast, BSP efficiently integrates local parameterized surfaces to construct a continuous global surface, achieving the best performance. To further illustrate the applicability of BSP to sparse input, we replace BSP with the state-of-art upsampling method LID [24] noted as Upsample. As shown in Tab. 7, LID also struggles to predict accurate result due to the highly sparse distribution. We provide detailed visualization comparison in supplementary.

Level of Input Size. We evaluate the robustness of our

	$CD_{L1} \times 10$	$CD_{L2} \times 100$	NC
Sparse	0.873	4.315	0.814
Upsample	0.427	0.986	0.830
Single	0.083	0.049	0.906
Multiple	0.087	0.051	0.901
Ours	**0.077**	**0.043**	**0.914**

Table 7. Effect of BSP.

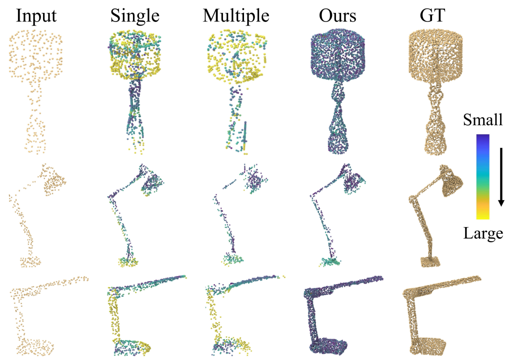

Figure 10. Effect of BSP. The color indicate the point distance error to ground truth surface.

method with different point size levels. Our visualization results are reported in Fig. 11. As the number of point clouds increases, we are able to generate more uniform parameterized surfaces and accurate geometries.

Effect of GDO. We demonstrate that GDO can learn a consistent deformation direction from gradients in Fig. 3. Here, we further justify the effectiveness of GDO in inferring the implicit function f. We first remove the gradient consistency constrain, and only learn the implicit functions from predicting the grid point offsets, denoted as f_{offset}. Then, we remove GDO and apply TPS optimization strategy as baseline denoted as f_{TPS}. Both of them cause increases the CD error at different levels. As shown in Tab. 8, grid deformation-based strategies (f_{offset} and Ours) achieve higher accuracy, and f_{GDO} provides the most precise geometric surface prediction. We provide visualizations of reconstruction results under different optimization strategies in supplementary.

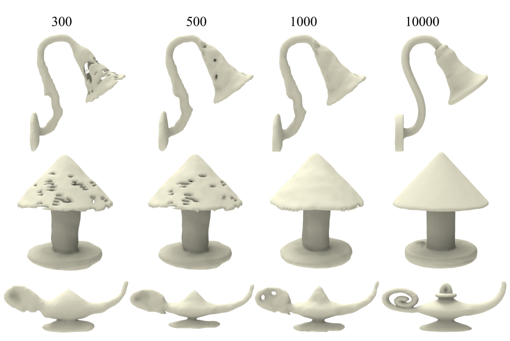

Figure 11. Visual comparison under levels of input size.

Effect of Loss Functions. To validate the effectiveness of each optimization term, we present the results of removing different loss terms in Tab. 9 to assess their importance in our method. We first remove L_{para} and rely solely on the sparse point cloud for reconstruction, which leads to a significant increase in CD error. To remove L_{deform}, we pretrain BSP to obtain a parameterized surface as supervision without further optimization, which also results in decreased accuracy. Finally, we remove L_{surf} results in slightly worse results. Overall, L_{para} and L_{deform} have a greater impact on the metrics, indicating that dense and further optimizable parameterized surface are important for learning accurate implicit functions.

	$CD_{L1} \times 10$	$CD_{L2} \times 100$	NC
f_{TPS}	0.094	0.058	0.861
f_{offset}	0.089	0.055	0.883
Ours	**0.077**	**0.043**	**0.914**

Table 8. Effect of GDO.

Number of Samples. We explore the effectiveness of different sample numbers of single local patch in Tab. 10. With the increasing of samples, the network can predict the parametric surface more precisely. However, when the hyper-parameter set to 15, the improvements in accuracy become marginal. To consider the balance between performance and efficiency, we set this hyper-parameter to 10 by default.

	$CD_{L1} \times 10$	$CD_{L2} \times 100$	NC
w/o L_{para}	0.873	4.315	0.814
w/o L_{deform}	0.085	0.053	0.898
w/o L_{surf}	0.081	0.044	0.908
Ours	**0.077**	**0.043**	**0.914**

Table 9. Effect of loss functions.

Sample Size	3	5	10	15
$CD_{L1} \times 10$	0.086	0.081	0.077	**0.075**
$CD_{L2} \times 100$	0.049	0.047	0.043	**0.041**
NC	0.896	0.905	**0.914**	**0.914**

Table 10. Number of Samples.

6. Conclusion

We propose an innovative training framework that learns smooth implicit fields from sparse point cloud inputs and reconstructs complete and continuous surfaces. Unlike previous methods, we parametrize local surfaces by learning bijective functions and integrate them into a global surface to ensure shape continuity. Experimental results demonstrate that the BSP strategy can generate more accurate parametrized surfaces. Additionally, we introduce a novel approach to apply deformation networks to sparse reconstruction tasks and propose GDO to further improve the accuracy of implicit field predictions. We validate the effectiveness of our method across extensive datasets and ablation studies. The results demonstrate its robustness for under varied conditions and settings.

References

[1] ShapeNet. https://www.shapenet.org/. 2, 5

[2] Matan Atzmon and Yaron Lipman. SALD: Sign agnostic learning with derivatives. In *International Conference on Learning Representations*, 2020. 1

[3] Ma Baorui, Liu Yu-Shen, and Han Zhizhong. Reconstructing surfaces for sparse point clouds with on-surface priors. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022. 1, 2

[4] Federica Bogo, Javier Romero, Gerard Pons-Moll, and Michael J Black. Dynamic faust: Registering human bodies in motion. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 6233–6242, 2017. 4

[5] Alexandre Boulch, Pierre-Alain Langlois, Gilles Puy, and Renaud Marlet. Needrop: Self-supervised shape representation from sparse point clouds using needle dropping. In *2021 International Conference on 3D Vision (3DV)*, pages 940–950. IEEE, 2021. 2, 4

[6] 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 *European Conference on Computer Vision*, pages 608–625. Springer, 2020. 2

[7] Chao Chen, Yu-Shen Liu, and Zhizhong Han. GridPull: Towards scalability in learning implicit representations from 3D point clouds. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 18322–18334, 2023. 2

[8] Chao Chen, Yu-Shen Liu, and Zhizhong Han. Unsupervised inference of signed distance functions from single sparse point clouds without learning priors. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 17712–17723, 2023. 1, 2, 3, 7

[9] Chao Chen, Yu-Shen Liu, and Zhizhong Han. Neuraltps: Learning signed distance functions without priors from single sparse point clouds. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2024. 2

[10] Zhiqin Chen and Hao Zhang. Learning implicit fields for generative shape modeling. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5939–5948, 2019. 2

[11] Julian Chibane, Thiemo Alldieck, and Gerard Pons-Moll. Implicit functions in feature space for 3D shape reconstruction and completion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6970–6981, 2020. 2

[12] Yueqi Duan, Haidong Zhu, He Wang, Li Yi, Ram Nevatia, and Leonidas J Guibas. Curriculum deepsdf. In *European Conference on Computer Vision*, pages 51–67. Springer, 2020. 1

[13] Philipp Erler, Paul Guerrero, Stefan Ohrhallinger, Niloy J Mitra, and Michael Wimmer. Points2surf learning implicit surfaces from point clouds. In *European Conference on Computer Vision*, pages 108–124. Springer, 2020. 1

[14] Jun Gao, Tianchang Shen, Zian Wang, Wenzheng Chen, Kangxue Yin, Daiqing Li, Or Litany, Zan Gojcic, and Sanja Fidler. Get3d: A generative model of high quality 3d textured shapes learned from images. *Advances In Neural Information Processing Systems*, 35:31841–31854, 2022. 2

[15] Andreas Geiger, Philip Lenz, and Raquel Urtasun. Are we ready for autonomous driving? the KITTI vision benchmark suite. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 3354–3361. IEEE, 2012. 2, 5

[16] Kyle Genova, Forrester Cole, Daniel Vlasic, Aaron Sarna, William T Freeman, and Thomas Funkhouser. Learning shape templates with structured implicit functions. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 7154–7164, 2019. 1

[17] Amos Gropp, Lior Yariv, Niv Haim, Matan Atzmon, and Yaron Lipman. Implicit geometric regularization for learning shapes. In *International Conference on Machine Learning*, pages 3789–3799, 2020. 1

[18] Paul Guerrero, Yanir Kleiman, Maks Ovsjanikov, and Niloy J Mitra. PCPNet: learning local shape properties from raw point clouds. In *Computer Graphics Forum*, pages 75–85. Wiley Online Library, 2018. 2

[19] Jiahui Huang, Hao-Xiang Chen, and Shi-Min Hu. A neural galerkin solver for accurate surface reconstruction. *ACM Transactions on Graphics (TOG)*, 41(6):1–16, 2022. 1

[20] Zhiyang Huang, Nathan Carr, and Tao Ju. Variational implicit point set surfaces. *ACM Transactions on Graphics (TOG)*, 38(4):1–13, 2019. 2

[21] Chiyu Jiang, Avneesh Sud, Ameesh Makadia, Jingwei Huang, Matthias Nießner, Thomas Funkhouser, et al. Local implicit grid representations for 3D scenes. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6001–6010, 2020. 1

[22] Mohammad Mahdi Johari, Yann Lepoittevin, and François Fleuret. Geonerf: Generalizing nerf with geometry priors. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18365–18375, 2022. 2

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

[24] Shujuan Li, Junsheng Zhou, Baorui Ma, Yu-Shen Liu, and Zhizhong Han. Learning continuous implicit field with local distance indicator for arbitrary-scale point cloud upsampling. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 3181–3189, 2024. 7

[25] Shi-Lin Liu, Hao-Xiang Guo, Hao Pan, Peng-Shuai Wang, Xin Tong, and Yang Liu. Deep implicit moving least-squares functions for 3D reconstruction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1788–1797, 2021. 2

[26] Sandro Lombardi, Martin R Oswald, and Marc Pollefeys. Scalable point cloud-based reconstruction with local implicit functions. In *2020 International Conference on 3D Vision (3DV)*, pages 997–1007. IEEE, 2020. 2

[27] William E Lorensen and Harvey E Cline. Marching cubes: A high resolution 3D surface construction algorithm. *ACM Siggraph Computer Graphics*, 21(4):163–169, 1987. 1

[28] Baorui Ma, Zhizhong Han, Yu-Shen Liu, and Matthias Zwicker. Neural-Pull: Learning signed distance function

from point clouds by learning to pull space onto surface. In *International Conference on Machine Learning*, pages 7246–7257. PMLR, 2021. 2, 5

[29] Baorui Ma, Yu-Shen Liu, Matthias Zwicker, and Zhizhong Han. Surface reconstruction from point clouds by learning predictive context priors. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022. 2

[30] Baorui Ma, Yu-Shen Liu, and Zhizhong Han. Learning signed distance functions from noisy 3d point clouds via noise to noise mapping. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2023. 2

[31] Baorui Ma, Junsheng Zhou, Yu-Shen Liu, and Zhizhong Han. Towards better gradient consistency for neural signed distance functions via level set alignment. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 17724–17734, 2023. 2

[32] Julien NP Martel, David B Lindell, Connor Z Lin, Eric R Chan, Marco Monteiro, and Gordon Wetzstein. ACORN: adaptive coordinate networks for neural scene representation. *ACM Transactions on Graphics (TOG)*, 40(4):1–13, 2021. 2

[33] Lars Mescheder, Michael Oechsle, Michael Niemeyer, Sebastian Nowozin, and Andreas Geiger. Occupancy networks: Learning 3D reconstruction in function space. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4460–4470, 2019. 2

[34] Zhenxing Mi, Yiming Luo, and Wenbing Tao. SSRNet: Scalable 3D surface reconstruction network. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 970–979, 2020. 1, 2

[35] B Mildenhall, PP Srinivasan, M Tancik, JT Barron, R Ramamoorthi, and R Ng. NeRF: Representing scenes as neural radiance fields for view synthesis. In *European Conference on Computer Vision*, 2020. 2

[36] Amine Ouasfi and Adnane Boukhayma. Unsupervised occupancy learning from sparse point cloud. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 21729–21739, 2024. 1, 2, 4, 5

[37] Songyou Peng, Michael Niemeyer, Lars Mescheder, Marc Pollefeys, and Andreas Geiger. Convolutional occupancy networks. In *European Conference on Computer Vision*, pages 523–540. Springer, 2020. 2

[38] Songyou Peng, Chiyu Jiang, Yiyi Liao, Michael Niemeyer, Marc Pollefeys, and Andreas Geiger. Shape as points: A differentiable poisson solver. *Advances in Neural Information Processing Systems*, 34, 2021. 1, 2

[39] Albert Pumarola, Artsiom Sanakoyeu, Lior Yariv, Ali Thabet, and Yaron Lipman. Visco grids: Surface reconstruction with viscosity and coarea grids. In *Advances in Neural Information Processing Systems*, 2022. 1

[40] Tianchang Shen, Jun Gao, Kangxue Yin, Ming-Yu Liu, and Sanja Fidler. Deep marching tetrahedra: a hybrid representation for high-resolution 3d shape synthesis. *Advances in Neural Information Processing Systems*, 34:6087–6101, 2021. 2, 4

[41] Tianchang Shen, Jacob Munkberg, Jon Hasselgren, Kangxue Yin, Zian Wang, Wenzheng Chen, Zan Gojcic, Sanja Fidler, Nicholas Sharp, and Jun Gao. Flexible isosurface extraction for gradient-based mesh optimization. *ACM Trans. Graph.*, 42(4):37–1, 2023. 2

[42] Vincent Sitzmann, Julien Martel, Alexander Bergman, David Lindell, and Gordon Wetzstein. Implicit neural representations with periodic activation functions. *Advances in neural information processing systems*, 33:7462–7473, 2020. 2

[43] 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. 1, 2

[44] Edgar Tretschk, Ayush Tewari, Vladislav Golyanik, Michael Zollhöfer, Carsten Stoll, and Christian Theobalt. PatchNets: Patch-based generalizable deep implicit 3D shape representations. In *European Conference on Computer Vision*, pages 293–309. Springer, 2020. 2

[45] Maria Vakalopoulou, Guillaume Chassagnon, Norbert Bus, Rafael Marini, Evangelia I Zacharaki, M-P Revel, and Nikos Paragios. Atlasnet: Multi-atlas non-linear deep networks for medical image segmentation. In *Medical Image Computing and Computer Assisted Intervention–MICCAI 2018: 21st International Conference, Granada, Spain, September 16-20, 2018, Proceedings, Part IV 11*, pages 658–666. Springer, 2018. 2

[46] A Vaswani. Attention is all you need. *Advances in Neural Information Processing Systems*, 2017. 3

[47] Zixiong Wang, Pengfei Wang, Qiujie Dong, Junjie Gao, Shuangmin Chen, Shiqing Xin, and Changhe Tu. Neural-imls: learning implicit moving least-squares for surface reconstruction from unoriented point clouds. *arXiv preprint arXiv:2109.04398*, 1(2):3, 2021. 6

[48] Francis Williams, Teseo Schneider, Claudio Silva, Denis Zorin, Joan Bruna, and Daniele Panozzo. Deep geometric prior for surface reconstruction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10130–10139, 2019. 4

[49] Francis Williams, Matthew Trager, Joan Bruna, and Denis Zorin. Neural splines: Fitting 3d surfaces with infinitely-wide neural networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9949–9958, 2021. 5

[50] Yuxin Yao, Siyu Ren, Junhui Hou, Zhi Deng, Juyong Zhang, and Wenping Wang. Dynosurf: Neural deformation-based temporally consistent dynamic surface reconstruction. In *European Conference on Computer Vision*, pages 271–288. Springer, 2025. 2

[51] Wang Yifan, Shihao Wu, Cengiz Oztireli, and Olga Sorkine-Hornung. Iso-points: Optimizing neural implicit surfaces with hybrid representations. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 374–383, 2021. 2

[52] Qian Yu, Chengzhuan Yang, and Hui Wei. Part-wise atlasnet for 3d point cloud reconstruction from a single image. *Knowledge-Based Systems*, 242:108395, 2022. 2, 7

[53] Junsheng Zhou, Baorui Ma, Shujuan Li, Yu-Shen Liu, and Zhizhong Han. Learning a more continuous zero level set

in unsigned distance fields through level set projection. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2023. 2

[54] Qian-Yi Zhou and Vladlen Koltun. Dense scene reconstruction with points of interest. *ACM Transactions on Graphics (ToG)*, 32(4):1–8, 2013. 5

[55] Zi-Xin Zou, Zhipeng Yu, Yuan-Chen Guo, Yangguang Li, Ding Liang, Yan-Pei Cao, and Song-Hai Zhang. Triplane meets gaussian splatting: Fast and generalizable single-view 3d reconstruction with transformers. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10324–10335, 2024. 2