DiGS : Divergence guided shape implicit neural representation for unoriented point clouds

Yizhak Ben-Shabat[1,2,3*] Chamin Hewa Koneputugodage[2,3*] Stephen Gould[2,3]

[1]Technion Israel Institute of Technology [2]The Australian National University
[3]Australian Centre for Robotic Vision

{yizhak.benshabat, chamin.hewa, stephen.gould}@anu.edu.au

Abstract

Shape implicit neural representations (INRs) have recently shown to be effective in shape analysis and reconstruction tasks. Existing INRs require point coordinates to learn the implicit level sets of the shape. When a normal vector is available for each point, a higher fidelity representation can be learned, however normal vectors are often not provided as raw data. Furthermore, the method's initialization has been shown to play a crucial role for surface reconstruction. In this paper, we propose a divergence guided shape representation learning approach that does not require normal vectors as input. We show that incorporating a soft constraint on the divergence of the distance function favours smooth solutions that reliably orients gradients to match the unknown normal at each point, in some cases even better than approaches that use ground truth normal vectors directly. Additionally, we introduce a novel geometric initialization method for sinusoidal INRs that further improves convergence to the desired solution. We evaluate the effectiveness of our approach on the task of surface reconstruction and shape space learning and show SOTA performance compared to other unoriented methods.
Code and model parameters available at our project page https://chumbyte.github.io/DiGS-Site/.

1. Introduction

Reconstructing surfaces from 3D point samples is a well studied problem in computer vision and computer graphics. Recently, neural networks have been used to learn an implicit neural representation (INR) that can be used to reconstruct the underlying surface [2, 3, 16, 17, 29, 34, 36, 37, 42], which we refer to as a shape INR. These methods are often supervised using estimated or known volumetric implicit representations in a regression setting [16, 29, 34] or using surface 3D points with or without extra 3D supervision (*e.g.* normal data) [2, 3, 17, 37] to regress the function directly. Due to the difficulty of optimizing a regression model to fit high fidelity surfaces, most methods use normal data. However, raw 3D point clouds from scans are typically unoriented, and thus do not have normal vectors, therefore a pre-processing estimation stage is required. While while some methods allow normal supervision to be absent [17, 37], we show that without it their performance drops significantly. While there have been significant advances in normal estimation algorithms [5, 6, 19, 25], all yield unoriented and noisy predictions. This poses a great challenge for existing normal-based shape INR learning approaches.

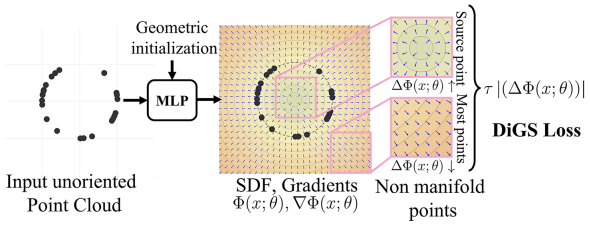

Figure 1. Toy 2D example. Given an unoriented input point cloud sampled on a shape, we train a shape implicit neural representation using a geometric initialization and a divergence penalty loss which arises from the observation that in most locations the divergence of the signed distance function should be low.

In this work, we introduce a divergence guided shape implicit neural representation learning approach (DiGS INR), which uses raw 3D point data for supervision without any pre-processing stages. Our approach is motivated by the observation, illustrated in Figure 1, that the gradient vector field of the signed distance function (produced by the network) has low divergence nearly everywhere. We incorporate this geometric prior as a soft constraint in the loss function and anneal it as training progresses. This requires a network architecture that has continuous second derivatives, such as SIRENs [37] (which we use), in contrast to

*Equal contribution

many previous works that use ReLU multi-layer perceptrons (MLPs). Additionally, we propose two novel geometrically driven initialization methods for this architecture to set the initial zero level set (i.e., the estimated shape's surface) to be approximately spherical, one of which explicitly maintains high frequencies in a controllable fashion.

We test DiGS on the task of surface reconstruction and shape space learning, and use shapes from the Surface Reconstruction Benchmark (SRB) [7], DFAUST [10], and ShapeNet [15] datasets, which consist of shapes with challenging properties, *e.g.*, topological complexity, nonuniform sampling, registration misalignment, missing data and different feature sizes (levels of detail). We show that our method suffers very little, if at all, without normal supervision and performs on par and sometimes better than state-of-the-art methods that use normals. Importantly, it performs significantly better than other methods that work on unoriented point clouds.

The main contributions of this paper are:

- Introducing divergence guided shape INR learning which incorporates soft second order derivative constraints to guide the INR learning process.
- Deriving two novel geometric initialization methods for sinusoidal-based shape INR networks that lead to better representations.

2. Related-work

Surface reconstruction. Reconstructing surfaces from point clouds is a difficult problem that has been studied for many years. Its main challenges include (1) nonuniform point sampling, (2) noisy point positions and normals due to sampling inaccuracy, scan registration misalignment and normal estimation errors, and (3) missing data in parts of the surface due to occlusions. Reconstruction methods attempt to overcome these challenges and infer the unknown underlying surface. Classical combinatorical approaches for reconstruction include triangulation methods [13, 24], alpha shapes [9] and Voronoi diagrams [1]. Classical implicit function based approaches include piecewise polynomial functions [32, 33], summing radial basis functions [12], and solving a Poisson equation to find a global indicator function [22, 23]. For a more in depth survey of classical surface reconstruction methods we refer the reader to Berger *et al*. [8]. Recently, and more relevant to this work, learning based approaches with neural networks have been proposed. These include parameteric methods that learn mappings from a parametric space to a region of the surface [18, 44], methods that learn an implicit volumetric function on regular grids [21, 40], methods that learn implicit neural representations (INRs) for the shape and recently a differentiable point-to-mesh optimisation layer [35].

Shape implicit neural representations (INRs). Neural networks have shown to be very efficient in representing shapes as implicit functions [2, 3, 17, 26, 29, 30, 34, 37, 39, 45, 46], and scenes [4, 20, 36, 38]. These networks are trained to output either a signed distance function [2–4, 17, 20, 30, 34, 37, 38, 45, 46], an occupancy function [29, 36, 39], or recently a representation that unifies the two [26].

Our method uses a signed distance function (SDF) to represent the shape. SDFs were popularised for shape INRs by DeepSDF [34]. However, for off surface points they require the ground truth for whether they are inside/outside the shape (i.e. the sign of the SDF), which is unrealistic. SAL [2] improves upon this by using a sign agnostic training method that requires no extra 3D supervision, and SALD [3] shows that using ground truth normals for points on the surface greatly improves performance. IGR [17] introduce an important eikonal equation based loss term to regularise the learnt function towards being an SDF, which can be used with or without normals. NSP [45] frame the task as a kernel regression problem where the kernel chosen is equivalent to the limit of infinitely wide, shallow ReLU networks. FFN [39] and SIREN [37] demonstrate that INRs using conventional MLPs bias toward low-frequency solutions, and thus explicitly introduce high frequencies into their architecture. FFN (which learn an occupancy implicit function) use a ReLU MLP with a Fourier feature layer, while SIREN uses periodic (specific sine) activation functions. The benefit of the latter is that second derivatives (and in fact all orders) are defined and continuous, which allows for higher order supervision as we use in this paper. On the other hand, DeepSDF, SAL use ReLU for activation, so the same cannot be done with those methods.

Ground truth normal information is not usually available in real world, raw scan data, and must be noisily estimated. Note that all these methods, except SAL (and DeepSDF which uses other unrealistic ground truth information), report results with ground truth normal information. Technically IGR and SIREN can operate without such information by dropping the corresponding loss term in their overall loss, but our experimental results in Section 6 show that performance significantly drops in their absence. On the other hand, we show that without normal information our method does not reduce performance as significantly, and in fact does on par with methods that use normals.

Concurrent work include IDF [46] and PHASE [26]. IDFs extend SIRENs to explicitly decompose learning the high and low frequencies of a shape, and combine by having the local high frequency as a displacement in the low frequency SIREN's normal direction. PHASE introduce a loss that learns a density function whose limit is an occupancy, and whose log transform is a SDF. They also introduce a version without ground truth normal supervision, PHASE+FF, which uses the Fourier features of FFN [39].

Initializing shape INRs. As finding good shape implicit representations is hard due to the ill-posed nature of

the task [8], many methods introduce initialization or regularization to bias toward favourable solutions [2, 3, 17]. SAL [2] introduce a geometric initialization for ReLU MLPs, which carefully chooses the initial weights (or the distribution of the initial weights) in particular layers to make the initial function be approximately the signed distance function to a r-radius sphere. This initialization has shown to have favourable properties for reconstruction, such as in-object and out-object sign consistency.

For SIRENs, to allow for training deeper networks for general INRs, Sitzmann *et al*. [37] proposed an initialization that preserves the distribution of activations through its layers. However for shape INRs, this initialization lacks a geometric grounding and sometimes causes unwanted ghost geometries. In this paper, we propose two novel initializations for SIRENs that extend the notion of geometric initialization to periodic activations and show that initializing with high frequencies is important for capturing fine detail.

3. Outline of Divergence Guided Shape INRs

We use a smooth-to-sharp approach that keeps the gradient vector field stay highly consistent during training (see Figure 2). In particular, it allows us to learn a good implicit representation without normal information. The proposed training procedure consists of the following four steps, which will be detailed in subsequent sections:

- **Geometric initialization**. Initialize to a sphere, biasing the function to start with an SDF that is positive away from the object and negative in the centre of the object's bounding box, while keeping the model's ability to have high frequencies (in a controllable manner).
- **High divergence phase.** Guide the model towards a smooth reconstruction of the coarse shape. Importantly, this prevents the model from prematurely fitting to fine details.
- **Annealing divergence phase.** Slowly allow fine details to emerge while still learning a function that has smoothly changing normals.
- **Low divergence phase.** Allow very fine details such as sharp corners to emerge, and for the function to interpolate the original data (point cloud samples) as much as possible (subject to also minimising the Eikonal term).

A high weight on the divergence loss produces very smooth SDF functions, leading to oversmoothed reconstructions. However learning such smooth representations can be done quickly and robustly. In the supplemental material, we provide a video showing the effects of this procedure on the reconstructions at different iteration steps. We divide the total number of iterations in 50%, 25% and 25% for the high, annealing and low divergence phases, respectively.

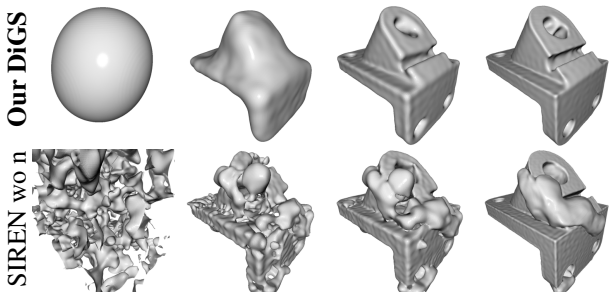

Figure 2. Results at four different training iterations (progressing from left to right) for DiGS (top) and SIREN wo n (bottom).

4. Geometric initialization for SIRENs

We now detail our two geometrically motivated initializations for SIRENs, shown in Figure 3 for 2D.

Initialization to a sphere. A key component of our method is a geometrically meaningful initialization for the parameters of SIRENs such that the initial signed distance function is approximately an r-radius sphere.

Let us consider a SIREN specified by

$$\begin{aligned} \Phi(x;\theta) &= \mathbf{w}_n^T \left(\phi_{n-1} \circ \phi_{n-2} \circ \ldots \circ \phi_0\right)(x) + b_n, \\ \phi_i(x_i) &= \sin\left(\mathbf{W}_i x_i + \mathbf{b}_i\right) \end{aligned} \tag{1}$$

where $\phi_i : \mathbb{R}^{M_i} \to \mathbb{R}^{N_i}$ is the i^{th} layer of the network, with input $x_i \in \mathbb{R}^{M_i}$ (so $x_0 = x$) and parameters $\theta = \left\{\mathbf{w}_n, b_n, \mathbf{W}_{n-1}, \mathbf{b}_{n-1}, \ldots \mathbf{W}_1, \mathbf{b}_1\right\}$ where $\mathbf{W}_i \in \mathbb{R}^{N_i \times M_i}$, $\mathbf{b}_i \in \mathbb{R}^{N_i}$, $\mathbf{w}_n \in \mathbb{R}^{M_n}$, $b_n \in \mathbb{R}$. Rather than approximating a norm using smooth SIRENs, we instead approximate the more tractable signed squared norm [43] and apply the following function to the output of the SIREN:

$$\nu(d) = \textbf{sign}\,(d)\,\sqrt{|d| + \varepsilon}. \tag{2}$$

Thus, we develop an initialization of the network's parameters, $\theta = \theta_0$, such that $\nu(\Phi(x;\theta_0)) \approx \|x\|_2$ for x within the unit ball. Note translating and scaling to the unit ball is standard for point clouds, so we are only interested in the INR within this region. We then manually minus r from this to initialize to a an r-radius sphere.

The following proposition shows this for a single hidden layer SIREN, where we approximate $z \mapsto z^2$ using a translated sine wave (see supplemental for proof).

Proposition 4.1. Let Φ be a single hidden layer SIREN ($n = 1$ in Equation 1) of dimension M_n and let x be a point within the unit ball. Set $\mathbf{W}_{n-1} = \frac{\pi}{2} I$, $\mathbf{b}_{n-1} = \frac{\pi}{2}\mathbf{1}$, $\mathbf{w}_n = -\mathbf{1}$ and $b_n = M_n$. Then, $\nu(\Phi(x)) \approx \|x\|_2$.

To extend this to networks with an arbitrary number of layers, we design layers ϕ_i that preserve the norm on expectation w.r.t. the weights of the layer: $\mathbb{E}[\|\phi_i(x_i)\|_2] = \|x_i\|_2$, so $\|x_{i+1}\|_2 \approx \|x_i\|_2$. We do this by sampling entries for the

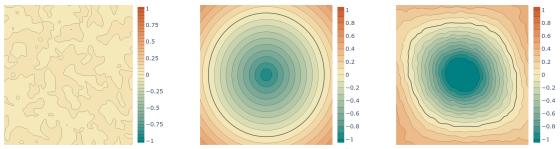

Figure 3. Visualization of the SDF of the proposed geometric initialization and multi-frequency geometric initialization (MFGI) for SIRENs in 2D compared to Sitzmann *et al*. [37].

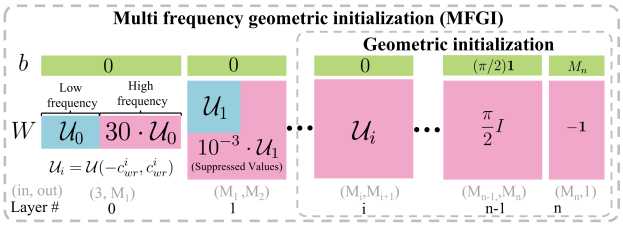

Figure 4. Illustration of our geometric initialization and MFGI.

weight matrix uniformly in the range $[-c^i_{wr}, c^i_{wr}]$ (defined below), which makes the rows approximately orthonormal. This also preserves the distribution type of the activations through all layers (see Sitzmann *et al*. [37]). Applying Proposition 4.1 to the last two layers yields the following proposition (see Figure 4 for a illustration of this).

Proposition 4.2. Let Φ be a n-hidden layer SIREN (Equation 1) that maps from $\mathbb{R}^{M_0} \to \mathbb{R}$ and $\|x\|_2 \leq 1$. Set $\mathbf{W}_i \sim \mathcal{U}\left(-c^i_{wr}, c^i_{wr}\right)$, $c^i_{wr} = \sqrt{\frac{3}{M_{i+1}}}$, $\mathbf{b}_i = \mathbf{0}$ for $0 \leq i \leq n-2$ and $\mathbf{W}_{n-1} = \frac{\pi}{2} I$, $\mathbf{b}_{n-1} = \frac{\pi}{2}\mathbf{1}$, $\mathbf{w}_n = -\mathbf{1}$ and $b_n = M_n$. Then $\nu(\Phi(x)) \approx \|x\|_2$.

We perturb all constant parameters in Proposition 4.2 with small Gaussian noise to facilitate learning.

However, as this initialization keeps all activations within the first period of sin, in practice SIRENs initialized this way will not generate high frequency output. Sitzmann *et al*. [37] also notes this problem, and specifically scales the weight matrix of the first layer by $\omega_0 = 30$ to hit up to 30 periods, thus giving multiple frequencies through the network. To overcome this problem, we proposed the multi frequency geometric initialization (MFGI).

Multi frequency geometric initialization (MFGI). We initialize using the geometric initialization of Proposition 4.2 and introduce high frequencies into the first layer in a controlled manner (see Figure 4 for an illustration of the method). Specifically, we keep the initialization for the first k_r rows of the weight matrix $\mathbf{W}_0 = \left[w^0_{ij}\right] \in \mathbb{R}^{N_0 \times M_0}$ as per Proposition 4.2, and for the last $N_0 - k_r$ rows we scale the initialized values by $n_p = 30$. This means that the output after the first layer, $x_1 \in \mathbb{R}^{N_0}$, has its first k_r elements hitting only one period and its remaining $N_0 - k_r$ elements hitting up to n_p periods. Then so that our geometric initialization still works, we scale down any part of the next weight matrix, $\mathbf{W}_1 = \left[w^1_{ij}\right] \in \mathbb{R}^{N_1 \times M_1}$, that would multiply into the multi-period part of the vector by a factor $s = 10^{-3}$. This can be summarised as:

$$w^0_{ij} \sim \begin{cases} \mathcal{U}\left(-c^0_{wr}, c^0_{wr}\right) & 0 \leq j \leq k_r \\ \mathcal{U}\left(-n_p\, c^0_{wr}, n_p\, c^0_{wr}\right) & \text{otherwise} \end{cases}$$
$$w^1_{ij} \sim \begin{cases} \mathcal{U}\left(-c^1_{wr}, c^1_{wr}\right) & 0 \leq j \leq k_r, 0 \leq i \leq k_r \\ \mathcal{U}\left(-s\, c^1_{wr}, s\, c^1_{wr}\right) & \text{otherwise} \end{cases} . \quad (3)$$

In our experiments we found that $k_r \approx \frac{1}{4} N_0$ is sufficient.

Visualisation of the initalizations. Figure 3 shows the two initializations introduced compared to the initialization proposed in SIREN [37] for a 4-layer SIREN with 128 nodes in each layer. Notice that SIREN's initialization has both SDF and gradient vector norm of almost zero everywhere. On the other hand, our geometric initialization has sphere-like level sets, which provide smooth and desirable eikonal and divergence terms (see supplemental for visualization of these components). Our MFGI initialization is a noisy version of our geometric initialization, where the amount of noise can be tuned by n_p, k_r and s.

5. DiGS Loss

5.1. Existing loss function components and setup

Prominent neural implicit representation methods [2, 3, 17,34,37] train a neural network with parameters θ to output a SDF $\Phi(x;\theta)$ to the surface of an underlying (unknown) shape for every given point $x \in \mathbb{R}^3$. They require several loss functions when training to constrain the learned function in certain ways: (A) *manifold constraint:* points on the surface manifold should be on the function's zero level set [2, 3, 17, 34, 37], (B) *normal constraint:* the gradients of points on the surface manifold should match the ground truth normal if such supervision is available [3, 17, 37] (C) either (C1) *non-manifold constraint:* points off the surface manifold should match their ground truth SDF or SDF magnitude [2,3,34] or (C2) *Eikonal constraint:* all points should have a unit gradient [17, 37], (D) *non-manifold penalisation constraint:* off surface points should not have zero SDF [37]. Note that (A) and (C) are necessary, and (B) and (D) are used for improving results.

Our method uses the same network architecture and loss functions of SIREN [37], which we provide for completeness. Given domain Ω and surface manifold Ω_0, they define the above loss functions as

$$L_A = \int_{\Omega_0} \|\Phi(x;\theta)\|_2 \, dx \quad (4)$$
$$L_B = \int_{\Omega_0} (1 - \langle \nabla_x \Phi(x;\theta), n_{\text{GT}}(x) \rangle) \, dx \quad (5)$$

$$L_{C2} = \int_{\Omega} \left| \|\nabla_x \Phi(x;\theta)\|_2 - 1 \right| dx \tag{6}$$

$$L_D = \int_{\Omega \setminus \Omega_0} \exp\left(-\alpha \left|\Phi(x;\theta)\right|\right), \quad \alpha \gg 1. \tag{7}$$

In summary, the final loss for SIREN is given by a weighted sum of all of the above terms:

$$L_{SIREN} = \lambda_A L_A + \lambda_B L_B + \lambda_{C2} L_{C2} + \lambda_D L_D \tag{8}$$

with $(\lambda_A, \lambda_B, \lambda_{C2}, \lambda_D) = (3000, 100, 50, 100)$ as given by Sitzmann et al. [37].

Supervision for (B) and (C1) are either unlikely to have in practice or costly to approximate well. Gropp *et al.* [17] showed however that (C1) can be replaced by (C2), a popular constraint in PDE theory called the Eikonal equation, which does not require extra data. In fact, given continuous (and sufficiently well behaved) boundary constraints (A), solving for a viscous solution to the PDE defined by the Eikonal equation will be unique (i.e., only (A) and (C2) are necessary). The difficulty in practice, however, is that constraint (A) is only defined at discrete points, resulting in infinitely many solutions, hence the requirement of other constraints to guide to favourable solutions (especially (B)).

We observe and deal with another problem (that we highlight in Section 5.3): our loss functions (and thus constraints) can only be evaluated at finite, discrete points within the domain. As a result the quality of our solution depends highly on the sampling density of our method, and how effectively our losses constrain the surrounding region. While we cannot increase the number of discrete points we get as supervision for (A), (B) greatly constrains the function space by adding higher order information (but requires extra supervision). We now do the same for (C) without extra supervision, and show that it can even remove the need for (B). Note that to benchmark against no normal supervision (B), we can also define

$$L_{SIREN\ wo\ n} = \lambda_A L_A + \lambda_{C2} L_{C2} + \lambda_D L_D. \tag{9}$$

5.2. Second order unsupervised constraints

We turn to second order information to further constrain the SDF around each sample on $\Omega \setminus \Omega_0$. Given the gradient vector field $\nabla\Phi$, we can compute its curl, $\nabla \times \nabla\Phi$, and divergence, $\Delta\Phi := \nabla \cdot \nabla\Phi$ (note that this is also known as the Laplacian of the underlying scalar field Φ) [28]. The curl of any gradient vector field is zero everywhere so it does not give us any information. However, we can observe that for a ground truth SDF, the magnitude of the divergence is very low at most areas of the domain (see supplemental material for discussion and intuition). We can thus impose a penalty on the magnitude of the divergence, *i.e.*,

$$L_{div} = \int_{\Omega \setminus \Omega_0} \left|\Delta\Phi(x;\theta)\right| dx = \int_{\Omega \setminus \Omega_0} \left|\nabla_x \cdot \nabla_x \Phi(x;\theta)\right| dx. \tag{10}$$

This leads to our proposed DiGS loss, given by:

$$L_{DiGS} = L_{SIREN\ wo\ n} + \tau \lambda_{div} L_{div} \tag{11}$$

where we use $\lambda_{div} = 100$ and τ is an annealing factor we discuss in Section 3.

Note that our approach can only be used with architectures that have activation functions with nonzero second derivative such as SIRENs. ReLU based networks will not be affected by the divergence constraint.

5.3. Minimising divergence as regularisation

We now provide some justification of the loss in Equation 10 by showing that it is equivalent to regularising the learnt function. We can quantify the complexity of our learnt function by using the Dirichlet Energy. The Dirichlet Energy of a function Φ over a space Ω gives a notion for how smooth or variable the function is [11] (where lower implies smoother), defined by

$$E[\Phi] = \frac{1}{2} \int_{\Omega} \|\nabla\Phi(x)\|_2^2 \, dx. \tag{12}$$

To minimize this with respect to our constraints (A)–(D), it suffices to find the function satisfying our conditions whose magnitude of the divergence is as small as possible, i.e., minimizing our divergence term L_{div} in Equation 10 (see supplemental material for proof).

Why not explicitly minimize the Dirichlet energy as a loss function? In fact, doing so would be redundant, due to our Eikonal term (6): if the gradient is constrained to have unit norm on the sampled points, then the Dirichlet energy, as far as can be determined from at those sampled points, is already determined. However we argue that adding our divergence term does a much better job of reducing the variability of our learned function over the entire space, rather than just having the Eikonal term. The reason for this is that it constrains the local region more due to its second order nature. This can also be motivated by the divergence theorem [31] (see supplemental for a thorough discussion). We use the following toy problem to demonstrate this.

Consider the problem of learning the SDF to the line $y = 0$ where below the line is negative, i.e. $\Phi((x, y)) = y$. We train a 2-layer SIREN with point constraints (A) and Eikonal term (C2), and then add our divergence constraint for comparison. For point constraints we sample 10 points on the lines $y = -1$, $y = 0$ and $y = 1$ (for $x \in \{0, 1\}$), and for Eikonal and divergence constraints we sample on a $n \times n$ grid. We repeat the experiment for $n \in \{20, 200\}$ and evaluate our learned functions on a finer $m \times m$ grid ($m = 1000$). We perform 20 repetitions of these four experiments with different random initializations.

The results can be seen in Table 1, where we report the mean Dirichlet energy $\bar{E} = \overline{\|\nabla f\|_2^2}$, the mean gradient norm $\overline{\|\nabla f\|_2}$ and its standard deviation $\sigma(\|\nabla f\|_2)$. We

Loss	grid size	$\bar{E} = \overline{\|\nabla f\|_2^2}$	$\overline{\|\nabla f\|_2}$	$\sigma(\|\nabla f\|_2)$
A+C2	20x20	4.32 ± 3.07	1.63 ± 0.51	1.05 ± 0.56
A+C2 + Div	20x20	3.37 ± 3.89	1.46 ± 0.76	0.69 ± 0.43
A+C2	200x200	3.58 ± 3.55	1.51 ± 0.54	0.85 ± 0.52
A+C2 + Div	200x200	$\mathbf{1.37} \pm \mathbf{1.35}$	$\mathbf{1.04} \pm \mathbf{0.39}$	0.22 ± 0.33

Table 1. Results on the toy problem. Comparing mean Dirichlet energy and mean gradient norm. The divergence term significantly improves performance.

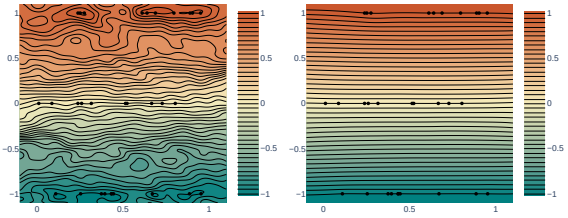

Figure 5. SDF contour lines for the toy problem on a 200x200 grid. With (right) and without (left) the divergence loss term.

also visualize the learnt SDFs for one of the repetitions in Figure 5. The results show that adding the divergence constraint both reduces the mean Dirichlet energy, indicating that the learned function is less variable, and makes the function more faithful to the Eikonal constraint. This can also be seen in the visualization, where the level set contours are less variable and have more consistent spacing. More visualizations are provided in the supplemental.

6. Experiments

We evaluate our method on the tasks of surface reconstruction and shape space learning. We test the former on the Surface Reconstruction Benchmark (SRB) [7], ShapeNet [15] and on a scene from Sitzmann et al. [37], and the latter on DFaust [10]. In both tasks, we extract the mesh representing the zero level set of the shape INR using the *marching cubes* algorithm [27]. We follow the same mesh generation procedure as in IGR [17] and use a grid whose shortest axis is 512 elements, tightly fitted onto the shape (adapting the grid range to the input scan bounding box). Unless otherwise specified, we use the same architecture (5 layers, 256 units) and hyperparameters as SIREN [37]. In the supplemental material, we provide full implementation details, extended results, ablations, additional experiments and additional visualizations.

Evaluation metrics. To compare between two point sets we use the Chamfer (d_C) and Hausdorff (d_H) distances. We follow IGR [17] and sample 1M points on the reconstruction and compare to the GT and scan point clouds. For ShapeNet we follow NSP [45] and report the squared Chamfer and Intersection over Union (IoU). See the supplemental for definitions and further details.

6.1. Surface reconstruction

The task of surface reconstruction of unoriented point clouds is specified as follows. Given an input point cloud $\mathcal{X} \subset \mathbb{R}^3$ find the surface $\mathcal{S}$ from which $\mathcal{X}$ was sampled. The point set is usually acquired by a 3D scanner which introduces various types of data corruptions, e.g., noise, occlusions, nonuniform sampling. We evaluate the performance of our method on the Surface Reconstruction Benchmark dataset [7] and on ShapeNet [15].

We compare our results to recent prominent methods, most of which have shown to outperform classical methods, including SAL [2], IGR [17], SIREN [37], DGP [44] and NSP [45]. Note that apart from SAL, these methods report results using normal vector supervision, therefore for fair comparison we evaluate on SIREN and IGR without the normal vector loss term (note that the same cannot be done with DGP and NSP). Furthermore we compare to results reported from the concurrent work PHASE [26], who evaluate on SRB. While PHASE uses normal information, they also provide a version without normals that uses Fourier features, PHASE+FF, and a baseline for IGR without normal information, IGR+FF.

Surface Reconstruction Benchmark (SRB). We use the simulated scan and ground truth data provided in Williams et al. [44] (freely available for academic use). The dataset contains five shapes, each with it own challenging traits, e.g., complex topology, high level of detail, missing data, and different feature sizes. In Table 2 we report the average Chamfer and Hausdorff distance for the shapes, and since the shapes have varying difficulties, we also report the mean deviation from the best performing method for each shape. It shows that our method is consistently better than SoTA methods on both metrics when normal information is not available. Qualitative results for a subset of the dataset are shown in Figure 6. In the absence of normal information, SIREN and IGR struggle to converge to the correct zero level set and produce undesired artifacts (ghost geometries). DiGS, on the other hand, is able to remove such artifacts. When compared to the version with normal supervision added there is not much change other than DiGS being slightly smoother. In fact, DiGS manages to get similar results to the methods that use normal supervision. For extended results (results per shape, results with normal supervision and more metrics), visualization of all shapes and a video showing the shapes from multiple angles, see the supplemental material.

ShapeNet. The ShapeNet dataset contains 3D CAD models of a diverse range of objects. These shapes often have internal structure, inconsistent normals, and non-manifold meshes. We use the preprocessing and split of Williams et al. [45], who evaluate on 20 shapes in each of 13 categories, and preprocess to make the normals consistent and internal structure to be manifold meshes. The re-

Method	d_C	d_H	Δd_C	Δd_H
IGR wo n	1.38	16.33	1.20	12.84
SIREN wo n	0.42	7.67	0.23	4.18
SAL [2]	0.36	7.47	0.18	3.99
IGR+FF [26]	0.96	11.06	0.78	7.58
PHASE+FF [26]	0.22	4.96	0.04	1.48
Our DiGS	**0.19**	**3.52**	**0.00**	**0.04**

Table 2. Results on the Surface Reconstruction Benchmark [7]. We compare against other methods without normal information, for all methods see the supplemental. We report the mean Chamfer d_C and Hausdorff distance d_H to the GT scans and their mean deviation from the best performing method (Δd_C and Δd_H).

	squared Chamfer ↓			IoU ↑		
Method	mean	median	std	mean	median	std
SPSR [23]	2.22e-4	1.70e-4	1.76e-4	0.6340	0.6728	0.1577
IGR [17]	5.12e-4	1.13e-4	2.15e-3	0.8102	0.8480	0.1519
SIREN [37]	1.03e-4	5.28e-5	1.93e-4	0.8268	0.9097	0.2329
FFN [39]	9.12e-5	8.65e-5	3.36e-5	0.8218	0.8396	0.0989
NSP [45]	**5.36e-5**	4.06e-5	3.64e-5	0.8973	0.9230	0.0871
DiGS + n	2.74e-4	**2.32e-5**	9.90e-4	**0.9200**	**0.9774**	0.1992
SIREN wo n	3.08e-4	2.58e-4	3.26e-4	0.3085	0.2952	0.2014
SAL [2]	1.14e-3	2.11e-4	3.63e-3	0.4030	0.3944	0.2722
Our DiGS	**1.32e-4**	**2.55e-5**	4.73e-4	**0.9390**	**0.9764**	0.1262

Table 3. Surface Reconstruction results on ShapeNet [15]. Methods above the line use ground truth normal information, and methods below do not. The mean, median and standard deviation of the squared Chamfer distance and IoU of all 260 shapes are reported.

Figure 6. Qualitative results of surface reconstruction on the anchor and gargoyle shapes from the Surface Reconstruction Benchmark [7] compared to state of the art approaches (IGR, SIREN) that use normal vectors as ground truth.

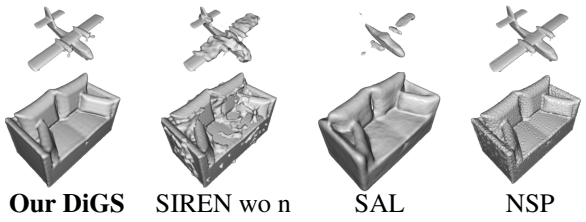

Figure 7. Results for surface reconstruction on ShapeNet [15].

sults are reported in Table 3 and we provide visualizations in Figure 7. When adding normal supervision to DiGS, our method has a much better median squared Chamfer distance and median IoU among the 260 shapes. We attribute this to our divergence term smoothing out the space, which does well on shapes without much internal structure. However

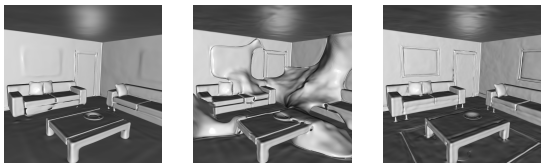

Figure 8. Qualitative results of scene reconstruction. SIREN (right) provides high level of detail, however, when normal vectors are not available (center) the proposed divergence constraint (left) demonstrates a significant improvement.

for some shapes with internal structure (*e.g.*, a loudspeaker with components inside or a sofa with structural beams) we get significant internal ghost geometry. As a result our mean squared Chamfer distance, while competitive with the other methods, is worse than methods such as SIREN. Note, however, that ths does not affect the IoU as much, our mean IoU is still better than other methods. When comparing without normals, DiGS has similar medians on both metrics to when normal supervision is added, however it has better means. We attribute this to having fewer internal ghost geometries when not attempting to fit normal vectors at internal points. Note that DiGS outperforms other methods that do not use normal supervision on both metrics. The improvement in IoU is significant, which we attribute to other methods failing to contain ghost geometry and being inconsistent with what is inside/outside the shape. On the other hand, DiGS appropriately deals with both of these challenges by its structured training procedure.

Scene reconstruction. Qualitative results for scene reconstruction are presented in Figure 8. Here we use eight layers with 512 units and train on the scene from Sitzmann et al. [37] which includes 10M oriented points. We train SIREN with and without the normal constraint, and the proposed DiGS method. In this experiment we did not use a geometric initialization because, unlike the shape reconstruction task, the target surface is vastly different than a sphere, and instead only increase the frequencies. When training SIREN without normal supervision we observe many ghost geometries (SIREN wo n). On the other hand, DiGS is able to reconstruct the scene without these, though it produces a very smooth result. This is desirable in most planar regions, e.g., ceiling, floor and table, however, this trait has the drawback of smoothing out fine details, e.g., sofa legs and picture frame.

6.2. Shape Space Learning

Shape space experiments requires training a single model to learn to represent multiple shapes from a class of related shapes. We use the DFAUST dataset [10] which consists of $\sim 40k$ raw human scans of ten humans at different time points during multiple types of activities. The

	$d_{\vec{C}}$(reg, recon)		$d_{\vec{C}}$(recon, reg)		$d_{\vec{C}}$(scan, recon)		$d_{\vec{C}}$(recon, scan)	
	Mean	Median	Mean	Median	Mean	Median	Mean	Median
IGR [17]	1.053	0.509	4.916	0.540	1.054	0.509	4.916	0.540
DiGS + n	**0.568**	**0.458**	**1.834**	**0.461**	**0.568**	**0.458**	**1.834**	**0.461**
IGR wo n	3.745	2.689	**12.149**	**9.027**	3.745	2.687	**12.147**	**9.026**
Our DiGS	**0.856**	**0.707**	12.318	9.202	**0.856**	**0.707**	12.319	9.204

Table 4. Quantitative results on DFaust [10]. We compare the mean and median of the one-sided Chamfer distances (reported as $\times 10^2$) between the ground truth registration meshes (reg), reconstructions (recon) and raw input scans (scan).

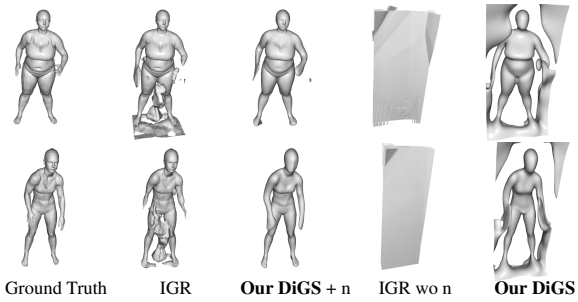

Figure 9. Qualitative results for the shape space experiment on the DFaust dataset [10]. Each row is a single pose of a different human from the test set.

scans are noisy and often incomplete (contains missing object surfaces). The dataset also provides ground truth registrations for each scan. Following IGR's [17] setup, we use their 75%/25% train-test split, and we use DiGS as an autodecoder [34]. Thus at training time, multiple scans of the humans are learnt with a separate latent vector for each pose, and at test time, unseen poses of those same humans are reconstructed by jointly estimating a suitable latent vector. Note this is a much harder problem than surface reconstruction, the model has to learn to be able to fit to multiple shapes given different (learnable) latent codes, stretching the test of the model's capacity.

Table 4 shows the quantitative results of one-sided Chamfer distances[1]. In particular, for registration & scan to reconstruction DiGS+n has a lower mean than IGR, showing it fits better to the ground truth and input surfaces and does not miss ground truth regions. For reconstruction to registration & scan the mean distances increase for both IGR and DiGS+n, indicating that both models create ghost geometry, however DiGS+n creates much less. Both of these are demonstrated in Figure 9: despite IGR sometimes having more detail (e.g., the face) it has multiple ghost geometries and significant missing parts (e.g., forearms). DiGS+n also achieves slightly smaller median distances.

For the methods without normals supervision, DiGS clearly outperforms IGR wo n, the latter is not able to converge. This can be seen in Figure 9, where the reconstructions do not even resemble the initial humans. DiGS on the other hand captures the human shapes, but is oversmoothed and has large ghost geometries framing the humans. Looking at Table 4 the means and medians for registration & scan to reconstruction are still quite small, showing that despite oversmoothing the detail DiGS learns to fit to the unseen test surfaces quite well. Due to the ghost geometry, for the reconstruction to registration & scan distances DiGS has very large values, similar to IGR wo n.

[1]For IGR we use the code and weights provided by [17]. Note that some methods report the squared Chamfer [2]. After correspondence with the authors of SAL and SALD [2, 3], we were unable to reproduce their results and therefore these baseline are omitted.

6.3. Limitations

DiGS is mainly limited in two aspects: (1) capturing very thin structures, e.g., the left sofa's legs in Figure 8, and (2) smoothing effects, e.g., the pictures on the walls in Figure 8. This is expected since these regions contain only very few points and without the normal vector information, uncovering the underlying surface is more challenging.

7. Conclusion

We introduce DiGS, a divergence guided shape implicit neural representation approach for raw unoriented point clouds without any pre-processing. Additionally, we derive a geometrically motivated initialization for sinusoidal representation networks while preserving high frequencies. Finally, we demonstrate that DiGS has the ability to reconstruct shapes of high fidelity with some limitations of smoothing and thin features reconstruction. We report state of the art results compared to other methods that do not use normal supervision, and show that our method is comparative to methods that do use such supervision.

All existing methods, including ours, struggle with missing data and thin features. Future work can explore extensions to use local self similarity to deal better with these regions. Point cloud density is also an important factor in the representation power and additional work can be done to mitigate density effects. Lastly, shape representation networks are highly dependant on the initialization and further work should be done to explore this direction.

Potential societal impact. The proposed DiGS approach enables accurate representation of 3D shapes from 3D point cloud data in a deep learning framework. Many down stream tasks may be enabled by DiGS, including avatar creation and computer aided design (CAD). These applications may be leveraged for negative and positive outcomes. For example, DiGS may be extended for generative tasks and enable novel approaches for shape generation. This has potential misuses including digital impersonation without consent and unauthorized reproduction of mechanical designs. This ties into DeepFakes which were discussed in depth in a recent review on neural rendering [41].

References

[1] Nina Amenta, Sunghee Choi, and Ravi Krishna Kolluri. The power crust, unions of balls, and the medial axis transform. *Computational Geometry*, 19:127–153, 2001. 2

[2] Matan Atzmon and Yaron Lipman. SAL: Sign Agnostic Learning of shapes from raw data. In *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2565–2574, 2020. 1, 2, 3, 4, 6, 7, 8

[3] Matan Atzmon and Yaron Lipman. SALD: Sign Agnostic Learning with Derivatives. In *Proc. of the International Conference on Learning Representations (ICLR)*, 2021. 1, 2, 3, 4, 8

[4] Dejan Azinovic, Ricardo Martin-Brualla, Dan B Goldman, Matthias Nießner, and Justus Thies. Neural RGB-D surface reconstruction. *arXiv preprint arXiv:2104.04532*, 2021. 2

[5] Yizhak Ben-Shabat and Stephen Gould. DeepFit: 3D surface fitting via neural network weighted least squares. In *Proc. of the European Conference on Computer Vision (ECCV)*, pages 20–34. Springer, 2020. 1

[6] Yizhak Ben-Shabat, Michael Lindenbaum, and Anath Fischer. Nesti-net: Normal estimation for unstructured 3D point clouds using convolutional neural networks. In *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 10112–10120, 2019. 1

[7] Matthew Berger, Joshua A Levine, Luis Gustavo Nonato, Gabriel Taubin, and Claudio T Silva. A benchmark for surface reconstruction. *ACM Trans. on Graphics (ToG)*, 32:1–17, 2013. 2, 6, 7

[8] Matthew Berger, Andrea Tagliasacchi, Lee M Seversky, Pierre Alliez, Gael Guennebaud, Joshua A Levine, Andrei Sharf, and Claudio T Silva. A survey of surface reconstruction from point clouds. In *Computer Graphics Forum*, volume 36, pages 301–329. Wiley Online Library, 2017. 2, 3

[9] Fausto Bernardini, Joshua Mittleman, Holly Rushmeier, Claudio Silva, and Gabriel Taubin. The ball-pivoting algorithm for surface reconstruction. *IEEE Trans. on Visualization and Computer Graphics*, 5:349–359, 1999. 2

[10] Federica Bogo, Javier Romero, Gerard Pons-Moll, and Michael J Black. Dynamic FAUST: Registering human bodies in motion. In *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 6233–6242, 2017. 2, 6, 7, 8

[11] Michael M Bronstein, Joan Bruna, Yann LeCun, Arthur Szlam, and Pierre Vandergheynst. Geometric deep learning: Going beyond Euclidean data. *IEEE Signal Processing Magazine*, 34:18–42, 2017. 5

[12] Jonathan C Carr, Richard K Beatson, Jon B Cherrie, Tim J Mitchell, W Richard Fright, Bruce C McCallum, and Tim R Evans. Reconstruction and representation of 3D objects with radial basis functions. In *Proc. of the Conference on Computer Graphics and Interactive Techniques*, pages 67–76, 2001. 2

[13] Frédéric Cazals and Joachim Giesen. Delaunay triangulation based surface reconstruction. In *Effective Computational Geometry for Curves and Surfaces*, pages 231–276. Springer, 2006. 2

[14] Frédéric Cazals and Marc Pouget. Estimating differential quantities using polynomial fitting of osculating jets. *Computer Aided Geometric Design*, 22:121–146, 2005.

[15] Angel X Chang, Thomas Funkhouser, Leonidas Guibas, Pat Hanrahan, Qixing Huang, Zimo Li, Silvio Savarese, Manolis Savva, Shuran Song, Hao Su, et al. ShapeNet: An information-rich 3D model repository. *arXiv preprint arXiv:1512.03012*, 2015. 2, 6, 7

[16] Zhiqin Chen and Hao Zhang. Learning implicit fields for generative shape modeling. In *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5939–5948, 2019. 1

[17] Amos Gropp, Lior Yariv, Niv Haim, Matan Atzmon, and Yaron Lipman. Implicit Geometric Regularization for learning shapes. In *Proc. of the International Conference on Machine Learning (ICML)*, volume 119, pages 3789–3799. PMLR, 2020. 1, 2, 3, 4, 5, 6, 7, 8

[18] Thibault Groueix, Matthew Fisher, Vladimir G Kim, Bryan C Russell, and Mathieu Aubry. A papier-mâché approach to learning 3D surface generation. In *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 216–224, 2018. 2

[19] Paul Guerrero, Yanir Kleiman, Maks Ovsjanikov, and Niloy J Mitra. PCPNet: Learning local shape properties from raw point clouds. In *Computer Graphics Forum*, volume 37, pages 75–85. Wiley Online Library, 2018. 1

[20] Chiyu Jiang, Avneesh Sud, Ameesh Makadia, Jingwei Huang, Matthias Nießner, Thomas Funkhouser, et al. Local implicit grid representations for 3D scenes. In *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 6001–6010, 2020. 2

[21] Yue Jiang, Dantong Ji, Zhizhong Han, and Matthias Zwicker. SDFDiff: Differentiable rendering of signed distance fields for 3D shape optimization. In *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1251–1261, 2020. 2

[22] Michael Kazhdan, Matthew Bolitho, and Hugues Hoppe. Poisson surface reconstruction. In *Proc. of the Eurographics Symposium on Geometry Processing (SGP)*, volume 7, 2006. 2

[23] Michael Kazhdan and Hugues Hoppe. Screened poisson surface reconstruction. *ACM Trans. on Graphics (ToG)*, 32:1–13, 2013. 2, 7

[24] Ravikrishna Kolluri, Jonathan Richard Shewchuk, and James F O'Brien. Spectral surface reconstruction from noisy point clouds. In *Proc. of the Eurographics Symposium on Geometry Processing (SGP)*, pages 11–21, 2004. 2

[25] Jan Eric Lenssen, Christian Osendorfer, and Jonathan Masci. Deep iterative surface normal estimation. In *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 11247–11256, 2020. 1

[26] Yaron Lipman. Phase transitions, distance functions, and implicit neural representations. *arXiv preprint arXiv:2106.07689*, 2021. 2, 6, 7

[27] William E Lorensen and Harvey E Cline. Marching cubes: A high resolution 3D surface construction algorithm. In *Proc. of the Intl. Conference on Computer Graphics and In-*

teractive Techniques (SIGGRAPH), volume 21, pages 163–169. ACM New York, NY, USA, 1987. 6

[28] Jerrold E Marsden and Anthony Tromba. *Vector calculus*. Macmillan, 2003. 5

[29] Lars Mescheder, Michael Oechsle, Michael Niemeyer, Sebastian Nowozin, and Andreas Geiger. Occupancy networks: Learning 3D reconstruction in function space. In *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4460–4470, 2019. 1, 2

[30] Mateusz Michalkiewicz, Jhony K Pontes, Dominic Jack, Mahsa Baktashmotlagh, and Anders Eriksson. Implicit surface representations as layers in neural networks. In *Proc. of the International Conference on Computer Vision (ICCV)*, pages 4743–4752, 2019. 2

[31] Philip M Morse and Herman Feshbach. Methods of theoretical physics. *American Journal of Physics*, 22:410–413, 1954. 5

[32] Yukie Nagai, Yutaka Ohtake, and Hiromasa Suzuki. Smoothing of partition of unity implicit surfaces for noise robust surface reconstruction. In *Computer Graphics Forum*, volume 28, pages 1339–1348. Wiley Online Library, 2009. 2

[33] Yutaka Ohtake, Alexander Belyaev, and Marc Alexa. Sparse low-degree implicit surfaces with applications to high quality rendering, feature extraction, and smoothing. In *Proc. of the Eurographics Symposium on Geometry Processing (SGP)*, pages 149–158. Citeseer, 2005. 2

[34] Jeong Joon Park, Peter Florence, Julian Straub, Richard Newcombe, and Steven Lovegrove. DeepSDF: Learning continuous signed distance functions for shape representation. In *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 165–174, 2019. 1, 2, 4, 8

[35] Songyou Peng, Chiyu Jiang, Yiyi Liao, Michael Niemeyer, Marc Pollefeys, and Andreas Geiger. Shape As Points: A differentiable poisson solver. *arXiv preprint arXiv:2106.03452*, 2021. 2

[36] Songyou Peng, Michael Niemeyer, Lars Mescheder, Marc Pollefeys, and Andreas Geiger. Convolutional occupancy networks. In *Proc. of the European Conference on Computer Vision (ECCV)*, 2020. 1, 2

[37] Vincent Sitzmann, Julien Martel, Alexander Bergman, David Lindell, and Gordon Wetzstein. Implicit neural representations with periodic activation functions. *Advances in Neural Information Processing Systems (NeurIPS)*, 33, 2020. 1, 2, 3, 4, 5, 6, 7

[38] Vincent Sitzmann, Michael Zollhöfer, and Gordon Wetzstein. Scene Representation Networks: Continuous 3d-structure-aware neural scene representations. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2019. 2

[39] Matthew Tancik, Pratul P. Srinivasan, Ben Mildenhall, Sara Fridovich-Keil, Nithin Raghavan, Utkarsh Singhal, Ravi Ramamoorthi, Jonathan T. Barron, and Ren Ng. Fourier features let networks learn high frequency functions in low dimensional domains. *Advances in Neural Information Processing Systems (NeurIPS)*, 2020. 2, 7

[40] Maxim Tatarchenko, Alexey Dosovitskiy, and Thomas Brox. Octree generating networks: Efficient convolutional architectures for high-resolution 3D outputs. In *Proc. of the International Conference on Computer Vision (ICCV)*, pages 2088–2096, 2017. 2

[41] Ayush Tewari, Ohad Fried, Justus Thies, Vincent Sitzmann, Stephen Lombardi, Kalyan Sunkavalli, Ricardo Martin-Brualla, Tomas Simon, Jason Saragih, Matthias Nießner, et al. State of the art on neural rendering. In *Computer Graphics Forum*, volume 39, pages 701–727. Wiley Online Library, 2020. 8

[42] Dahlia Urbach, Yizhak Ben-Shabat, and Michael Lindenbaum. DPDist: Comparing point clouds using deep point cloud distance. In *Proc. of the European Conference on Computer Vision (ECCV)*, pages 545–560. Springer, 2020. 1

[43] Christopher Williams and Kenneth Duru. Provably stable full-spectrum dispersion relation preserving schemes. *arXiv preprint arXiv:2110.04957*, 2021. 3

[44] Francis Williams, Teseo Schneider, Claudio Silva, Denis Zorin, Joan Bruna, and Daniele Panozzo. Deep Geometric Prior for surface reconstruction. In *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 10130–10139, 2019. 2, 6

[45] Francis Williams, Matthew Trager, Joan Bruna, and Denis Zorin. Neural Splines: Fitting 3D surfaces with infinitely-wide neural networks. In *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 9949–9958, 2021. 2, 6, 7

[46] Wang Yifan, Lukas Rahmann, and Olga Sorkine-Hornung. Geometry-consistent neural shape representation with implicit displacement fields. *arXiv preprint arXiv:2106.05187*, 2021. 2