Towards Better Gradient Consistency for Neural Signed Distance Functions via Level Set Alignment

Baorui Ma†, Junsheng Zhou†, Yu-Shen Liu‡, Zhizhong Han

1School of Software, BNRist, Tsinghua University, Beijing, China
2Department of Computer Science, Wayne State University, Detroit, USA

Abstract

Neural signed distance functions (SDFs) have shown remarkable capability in representing geometry with details. However, without signed distance supervision, it is still a challenge to infer SDFs from point clouds or multi-view images using neural networks. In this paper, we claim that gradient consistency in the field, indicated by the parallelism of level sets, is the key factor affecting the inference accuracy. Hence, we propose a level set alignment loss to evaluate the parallelism of level sets, which can be minimized to achieve better gradient consistency. Our novelty lies in that we can align all level sets to the zero level set by constraining gradients at queries and their projections on the zero level set in an adaptive way. Our insight is to propagate the zero level set to everywhere in the field through consistent gradients to eliminate uncertainty in the field that is caused by the discreteness of 3D point clouds or the lack of observations from multi-view images. Our proposed loss is a general term which can be used upon different methods to infer SDFs from 3D point clouds and multi-view images. Our numerical and visual comparisons demonstrate that our loss can significantly improve the accuracy of SDFs inferred from point clouds or multi-view images under various benchmarks. Code and data are available at https://github.com/mabaorui/TowardsBetterGradient.

1. Introduction

Signed distance functions (SDFs) have shown remarkable abilities in representing high fidelity 3D geometry [6, 14, 16, 28, 32, 36, 38, 41, 42, 45, 51, 52, 54, 55, 62, 63, 67–69, 76]. Current methods mainly use neural networks to learn SDFs as a mapping from 3D coordinates to signed distances. Using gradient descent, we can train neural networks by adjusting parameters to minimize errors to either signed distance ground truth [9, 31, 45, 51, 52] or signed distances inferred from 3D point clouds [1, 2, 11, 22, 38, 60, 77] or multi-view images [19, 24, 66–69, 73, 74, 76]. However, factors like the discreteness in point clouds and the lack of observations in multi-view images result in 3D ambiguity, which makes inferring SDFs without ground truth signed distances remain a challenge.

Recent solutions [1, 23, 32, 60, 68] impose additional constraints on gradients with respect to input coordinates. The gradients determine the rate of change of signed distances in a field, which is vital for the accuracy of SDFs. Specifically, Eikonal term [1, 23, 32] is widely used to learn SDFs, which constrains the norm of gradients to be one at any location in the field. This regularization ensures the networks to predict valid signed distances. NeuralPull [38] constrains the directions of gradients to pull arbitrary queries onto the surface. One issue here is that these methods merely constrain gradients at single locations, without considering gradient consistency to their corresponding projections on different level sets. This results in inconsistent gradients in the field, indicated by level sets with poor parallelism, which drastically decreases the accuracy of inferred SDFs.

To resolve this issue, we introduce a level set alignment loss to pursue better gradient consistency for SDF inference without ground truth signed distances. Our loss is a general term which can be used to train different networks for learning SDFs from either 3D point clouds or multi-view images. Our key idea is to constrain gradients at corresponding locations on different level sets of the inferred SDF to be consistent. We achieve this by minimizing the cosine distance between the gradient of a query and the gradient of its projection on the zero level set. Minimize our loss is equivalent to aligning all level sets onto a reference, i.e. the zero level set, in a pairwise way. This enables us to propagate the zero level set to everywhere in the field, which eliminates uncertainty in the field that is caused by the discreteness of
3D point clouds or the lack of observations from multi-view images. Moreover, we introduce an adaptive weight to focus more on the gradient consistency nearer to the zero level set. We evaluate our loss upon the latest methods in surface reconstruction and multi-view 3D reconstruction under the widely used benchmarks. Our improvements over baselines justify not only our effectiveness but also the importance of gradient consistency to the inference of signed distance fields. Our contributions are listed below.

i) We introduce a level set alignment loss to achieve better gradient consistency for inferring SDFs without signed distance ground truth.

ii) We justify the importance of gradient consistency to the accuracy of SDFs inferred from 3D point clouds and multi-view images, and show that aligning level sets together is an effective way of learning more consistent gradients for eliminating 3D ambiguity.

iii) We show our superiority over the state-of-the-art methods in surface reconstruction from point clouds and multi-view 3D reconstruction under the widely used benchmarks.

2. Related Work

Neural implicit representations have shown prominent performance in representing 3D geometry with details [14, 25, 32, 38, 43, 45, 46, 49, 51, 52, 68–70, 76]. With signed distances and occupancy labels as supervision, we can learn neural implicit representations as a regression [52] or classification [43] problem. In the following, we focus on reviewing methods inferring supervision from 3D point clouds [1, 23, 32] and multi-view images [46].

Supervision from 3D Point Clouds. Some methods learn SDFs or occupancy with 3D point clouds as conditions. They require signed distances and occupancy labels as supervision to learn global priors [8, 18, 21, 30, 35, 44, 51, 61, 64] or local priors [8, 9, 12, 31, 34, 65, 71], which can be generalized to unseen cases. With the ground truth field, these methods get benefits including perfect scalar fields with consistent gradients, but struggle to generalize the learned priors to unseen cases with large geometry variations.

Some other methods infer SDFs without supervision by training neural networks to overfit to single point cloud. These methods require additional constraints [1, 2, 5, 22, 75, 77], specially designed operations [11, 38, 54] or normal maps [7, 33, 72] to estimate signed distances or occupancy using point clouds as a reference. Using similar idea, we can infer unsigned distances from point clouds [13, 78]. Using inferred signed distances, some methods use the inferred SDFs as priors, and then guide the SDF inference from a novel point cloud [39, 40].

Supervision from Multi-View Images. With multi-view images as supervision, classic multi-view stereo (MVS) [58, 59] methods use multi-view consistency to estimate depth maps. With differentiable renderers [26, 32, 66], we can render images from the learned SDFs, and refine the learned SDFs by minimizing errors between the rendered images and ground truth images. Similarly, DVR [48] and IDR [74] infer the radiance on surfaces, where IDR models view direction as a condition to reconstruct high frequency details. However, these methods focus on surfaces, which makes them require masks of objects during optimization. Hence, we can not use them to reveal structures for scenes, where no masks are available.

NeRF [46] and the following work [47, 53, 56, 57] were proposed for novel view synthesis, and render images from radiance field use volume rendering without requiring masks. By simultaneously modelling geometry and color, we can infer signed distance or occupancy fields by minimizing rendering errors. With samples on rays shooting from pixels into the field, unsurf [49] and NeuS [68] use a revised rendering procedure to render occupancy and signed distance fields with radiance into pixel colors. Following methods improve accuracy of implicit functions using multi-view consistency [10, 20, 27, 68, 69, 76] or additional priors including depth [3, 76, 80], normals [24, 67, 76].

The SDFs inferred these methods are not accurate, due to the poor gradient consistency in signed distance fields, indicated by level sets with poor parallelism. This is the key factor that impacts on the accuracy of inferred SDFs through neural rendering in a multi-view context or reasoning on point clouds. We improve gradient consistency by aligning level sets on the zero level set via minimizing a level set alignment loss. Our loss is a general term that can be used upon different methods.

3. Method

Neural SDFs and Level Sets. We focus on inferring an SDF $f$ from a 3D point cloud or a set of multi-view images which does not provide ground truth signed distances. $f$ predicts a signed distance $s \in \mathbb{R}$ for an arbitrary query point $q \in \mathbb{R}^3$, as formulated by,

$$ s = f_\theta(q), \quad (1) $$

where we use a neural network parameterized by $\theta$ to learn the SDF $f$. Level sets of $f\_\theta$ are denoted as $\{S_l\}$, each of which is a set of points where $f_\theta$ takes on a given constant value $l$,

$$ S_l = \{q \mid f_\theta(q) = l\}, \quad (2) $$

where we regard zero level set $S_0$ as the surface of the scene. We extract the surface as a triangle mesh by running the marching cubes algorithm [37].
Inferring SDFs. Without signed distance ground truth, current methods infer SDFs by mining supervision from 3D point clouds with normals \([1,23,60]\), 3D point clouds without normals \([11,38,54]\), or multi-view images \([19,24,66–69,73,74,76]\). Although these methods use supervision in different modalities, all of them minimize a general form of loss function \(E\) to infer the SDF \(f_0\) below,

\[
\min_{\theta} E(T(f_0), G), \tag{3}
\]

where \(G\) is the supervision including 3D point clouds with or without normals or multi-view images, \(T\) is a transformation function that transforms signed distances into a representation in the same modality of \(G\), and \(E\) is the metric function that evaluates the error between the representation transformed from \(f_0\) and the ground truth supervision \(G\). More specifically, NeuralPull \([38]\) uses 3D point clouds without normals as \(G\), then the function \(T\) projects a query \(q\) on \(G\) using its signed distance \(f_0(q)\) and gradient \(\nabla f_0(q)\), and the loss \(E\) is mean squared error (MSE) between query projections and ground truth \(G\). Siren \([60]\) uses 3D point clouds with normals as \(G\), the function \(T\) is the Eikonal term regulating gradients and MSE over signed distances of surface points, the loss \(E\) is an energy based metric. NeuS \([68]\) uses a set of multi-view images as \(G\), then uses volume rendering as the function \(T\) to render \(f_0\) along with radiance into images, and compares the rendered images to \(G\) using a MSE \(E\).

Gradient Consistency. Our main contribution lies in pursuing better gradient consistency. We illustrate gradient consistency in the field using one query \(q\) in Fig. 1. If gradients are consistent, as shown in Fig. 1 (c), the gradient at query \(q\) and the gradient at its projection on each level set \(S_l\) should point to the same direction, which leads to level sets with great parallelism, while inconsistent gradients shown in Fig. 1 (a) result in level sets with poor parallelism. To evaluate gradient consistency at a query \(q\), we use cosine distance between gradients at query and its projection on a level set \(S_l\),

\[
c(q, S_l) = 1 - \frac{\nabla f_0(q) \cdot \nabla f_0(p')}{\|\nabla f_0(q)\|_2 \cdot \|\nabla f_0(p')\|_2}, \tag{4}
\]

where \(p'\) is the projection of \(q\) on the level set \(S_l\). \(c(q, S_l)\) is in a range of \([0, 2]\), where 0 indicates \(f_0(q)\) and \(f_0(p')\) are pointing to the same direction, which are the most consistent.

One issue here is that it costs extensive computation if we evaluate \(c(q, S_l)\) on each level set \(S_l\). Our solution here is to use zero level set \(S_0\) as a reference and project all queries onto the reference to evaluate the gradient consistency. In this pairwise way, we associate all level sets \(S_l\) to the zero level set \(S_0\), which can propagate consistency to all level sets through the projections on \(S_0\) since we randomly sample a large amount of queries in each iteration during optimization. Hence, we pursue a better gradient consistency by minimizing Eq. 4 over all sampled queries \(Q\),

\[
\min_{\theta} \sum_{q \in Q} c(q, S_0). \tag{5}
\]

Minimizing the loss in Eq. 5 is equivalent to align all level sets to the zero level set, which we named it as level set alignment loss, as illustrated in Fig. 1 (c).

Loss Function. We use our level set alignment loss upon methods for inferring neural SDF without signed distance ground truth. We formulate our loss function by combining Eq. 3 and Eq. 5 below,

\[
\min_{\theta} E_{q \in Q}(T(f_0(q)), G) + \alpha \sum_{q \in Q} \beta_q c(q, S_0). \tag{6}
\]

where \(\alpha\) is the balance weight for our level set alignment loss, and it scales the per point weight \(\beta_q\) which is an adaptive weight indicating the importance of each query \(q\), as defined below,

\[
\beta_q = \exp(-\delta \cdot |f_0(q)|), \tag{7}
\]

where we model \(\beta_q\) according to the predicted signed distance, which aims to encourage the optimization to focus more on the area near the surface. Another option to replace \(|f_0(q)|\) is to use the distance between \(q\) and its nearest point in point cloud representing a surface. However, finding the nearest point for each query \(q\) increases computational burden. Moreover, we can not use it in scenarios without point clouds such as multi-view images.

By optimizing the objective function in Eq. 6, we can achieve more consistent gradients in the field, as illustrated in Fig. 2. By optimizing with our level set alignment loss, we reformulate the loss function of NeuralPull \([38]\) into Eq. 6, which improves the gradient consistency in the field learned from 3D point clouds without normals. Fig. 2 (a) shows that we improve the parallelism of level sets, especially near the surface and inside of the dragon, where we visualize the signed distance field on a cross section of the reconstructed surface. This enables us to eliminate the swollen effect on the reconstructed surface of NeuralPull, which achieves a more compact surface with sharper edges. Similarly, we reformulate the loss function of NeuS \([68]\) into Eq. 6 by adding our level set alignment loss, which improves the gradient consistency in the field learned from multi-view images. Fig. 2 (b) shows that the better gradient consistency leads to level sets with better parallelism, which propagates the zero level sets to everywhere in the field. This is a key factor to eliminate the artifacts in the empty space.
Projections on the Zero Level Set. We project a query \( q \) onto the zero level set \( S_0 \), and use the projection \( p^0 \) to evaluate the gradient consistency defined in Eq. 4. As illustrated in Fig. 1(b), we follow the differentiable pulling operation in [38], and use the predicted signed distance \( f_\theta(q) \) and the gradient \( \nabla f_\theta(q) \) to project \( q \), as formulated by,

\[
p^0 = q - \frac{\nabla f_\theta(q)}{||\nabla f_\theta(q)||_2}.
\]

By replacing \( p^l \) in Eq. 4 into \( p^0 \), we obtain \( c(q, S_0) \) in Eq. 5 and Eq. 6 below,

\[
c(q, S_0) = 1 - \frac{\nabla f_\theta(q) \cdot \nabla f_\theta(p^l)}{||\nabla f_\theta(q)||_2 \cdot ||\nabla f_\theta(p^l)||_2}.
\]

4. Experiments

We conduct experiments to evaluate our method in learning neural signed distance functions for 3D reconstruction from 3D point clouds and multi-view images. We use our level set alignment loss upon different methods to improve the performance by encouraging more consistent gradients in the field. We extract the zero level set of the learned signed distance functions using the marching cubes algorithm [37] as a surface. Note that we do not evaluate our performance with methods learning from signed distance ground truth, since the supervision provides perfect gradient consistency in the field, which does not highlight our inference capability.

4.1. Surface Reconstruction from 3D Point Clouds

Datasets. We evaluate our performance under three datasets including the one released by SIREN [60], Stanford Scanning [15] and 3D Scene [79]. These datasets contain challenging cases including single objects and scenes with arbitrary topology and complex geometry. We use the point clouds in the dataset released by SIREN, which each scene contains millions points, and we sample 2 million points for each shape or scene in Stanford Scanning and 3D Scene.

Metrics. We evaluate the accuracy of the learned SDFs using the error between the reconstructed meshes and ground truth. We use L1 Chamfer distance (CD) and normal consistency (NC) to measure the error. We sample 100k points on the reconstructed mesh and ground truth to calculate CD in Stanford Scanning dataset, and sample 1 million points to calculate CD in SIREN dataset and 3D Scene. We also use the normals estimated on the reconstructed meshes for
We report our performance with the latest methods learning SDFs from 3D point clouds including I-GR [22], SIREN [60], NeuralPull [38]. These methods infer SDFs by training neural networks to overfit single 3D point cloud, with learning priors from large scale dataset. Specifically, IGR and SIREN adopt similar strategy to infer SDFs. They use Eikonal term to constrain the length of gradients to be one at everywhere, employ additional point normals to constrain gradients at points on surface, and set signed distances on surface to be zero. While NeuralPull constrains the gradients and signed distances together via pulling a query onto the surface without normals.

Details. To highlight our capability of inferring consistent gradients, we do not use the ground truth normal to produce our results with IGR and SIREN, since the ground truth normal is a direct supervision for gradients. We report our result upon the baselines using their official code. We use the loss function of the baseline to replace the first term in Eq. 6, which is combined with our level set alignment loss into a loss function we use to report our results. We set the weight $\alpha$ to make our loss contribute equally as the loss of the baseline.

Comparison. We report numerical evaluations in SIREN dataset in Tab. 1. In these point clouds with high frequency details, SIREN performs not well without using normals as supervision. Since ground truth normals determines the distance field near point clouds, which is the key to reconstruct accurate surface. But, without normals, the other loss terms in SIREN, such as the Eikonal term, can not infer accurate signed distances. While minimizing our loss can achieve more accurate signed distance field, which reveals surfaces with details even without using normal supervision. We also report the comparison with normal supervision in our supplementary materials.

We report our evaluation in Stanford scanning dataset in Tab. 2. Without using normals as supervision, SIREN and IGR reconstruct surfaces with artifacts. With our loss, we improve the gradient consistency in the field, especially near the surfaces. As shown in Fig. 5, we can eliminate the artifacts in the empty area, and obtain more completed and smoother surface. Since NeuralPull can not infer the zero level set very accurately, its reconstructed surfaces look a little bit “fat”. The swollen effect is mainly caused by inference uncertainty near surfaces. Training with our loss can make NeuralPull infer more accurate distance fields with much less uncertainty. This leads to more compact surfaces with more details, as shown in Fig. 5.

We visualize the level sets learned with our loss in Fig. 3 and Fig. 4. The comparisons of level sets shown in Fig. 3 indicate that better gradient consistency can achieve more completed level sets near the surface. More visualization of level sets can be found in Fig. 4.

We further evaluate our method in 3D scene dataset in Tab. 3. Our level set alignment loss significantly improves the performance of baselines. Visual comparisons in Fig. 6 illustrate that we improve the field by removing artifacts near the surface, reconstructing thinner and more compact surfaces, and sharpening surface edges. Moreover, we also report visual comparisons with methods using learned priors in our supplementary materials.

### 4.2. 3D Reconstruction from Multi-view Images

**Dataset.** We further evaluate our loss in reconstructing 3D shapes from multi-view images in the DTU dataset [29]. Following previous methods [19,49,68,69,73,74,76], we
Table 3. Numerical comparison with baselines in 3D scene dataset.

<table>
<thead>
<tr>
<th></th>
<th>Burghers</th>
<th>Lounge</th>
<th>Copyroom</th>
<th>Stonewall</th>
<th>Totempole</th>
</tr>
</thead>
<tbody>
<tr>
<td>MPU [50]</td>
<td>0.456</td>
<td>0.720</td>
<td>0.206</td>
<td>0.832</td>
<td>0.428</td>
</tr>
<tr>
<td>ConvOcc [61]</td>
<td>0.077</td>
<td>0.865</td>
<td>0.042</td>
<td>0.857</td>
<td>0.066</td>
</tr>
<tr>
<td>LIG [31]</td>
<td>0.018</td>
<td>0.904</td>
<td>0.017</td>
<td>0.910</td>
<td>0.018</td>
</tr>
<tr>
<td>NP [38]</td>
<td>0.010</td>
<td>0.883</td>
<td>0.059</td>
<td>0.857</td>
<td>0.011</td>
</tr>
<tr>
<td>Ours (NP)</td>
<td>0.008</td>
<td>0.947</td>
<td>0.020</td>
<td>0.936</td>
<td>0.009</td>
</tr>
<tr>
<td>SIREN [60]</td>
<td>0.025</td>
<td>0.944</td>
<td>0.064</td>
<td>0.933</td>
<td>0.032</td>
</tr>
<tr>
<td>Ours (SIREN)</td>
<td>0.016</td>
<td>0.948</td>
<td>0.021</td>
<td>0.929</td>
<td>0.026</td>
</tr>
</tbody>
</table>

Figure 6. Visual comparison with baselines in 3D scene dataset.

report our results on the widely used 15 scenes, each of which shows single object with background in 49 to 64 images with different shape appearances. For larger scale scenes, we report our results under ScanNet [17]. For fair comparison, we follow MonoSDF [76] to conduct evaluations using the same scenes.

**Metrics.** For evaluations under DTU dataset, we use L1 Chamfer distance to evaluate the error of points randomly sampled on the reconstructed surfaces compared to the ground truth. Following previous methods [19, 49, 68, 73, 74, 76], we clean the reconstructed meshes using the respective masks. We use the official evaluation code released by the DTU dataset to measure our accuracy. For evaluations under ScanNet [17], we use the same metrics as MonoSDF, which includes Chamfer distance, F-Scores with a threshold of 5cm, and normal consistency to measure the error between the reconstructed surface and the ground truth surface in ScanNet.

**Baselines.** We add our loss on the latest methods for learning SDFs from multi-view images. We use NeuS [68] and MonoSDF [76] as baselines. NeuS does not use priors, and infer an SDF using multi-view consistency through volume rendering. MonoSDF adopts the same strategy and learns SDFs with depth and normal priors on images.

**Details.** We use the official code released by NeuS and MonoSDF to produce our results with our level set alignment loss into a loss function we use to report our results. We set the weight \( \alpha \) to make our loss contribute equally as the loss of the baseline.

**Comparison.** We report numerical evaluations in DTU Tab. 4. We achieve better performance in 10 out of 15 scenes, and get comparable results in the other 5 scenes. In terms of the Chamfer distance, our improvements over NeuS are subtle. The reason is that the advantages of better gradient consistency lie in the ability of improving the smoothness of surfaces and removing artifacts in empty space. However, the smoothness does not significantly improve the numerical results, and artifacts in empty space has been cleaned using respective masks following the evaluation protocol. Hence, we highlight our improvements in visual comparison in Fig. 7, where we show the reconstructed surfaces before the cleaning. More analysis and comparisons with NeuS can be found in supplementary materials.

As we can see, unisurf and NeuS learn neural implicit fields with lots of uncertainty, which is caused by the lack of multi-view consistency constraints or the ambiguity with the textureless background. This uncertainty results in artifacts especially in empty space. By minimizing our level set alignment loss, we can propagate the zero level set to all other level sets everywhere in the field through consistent gradients, which eliminates the uncertainty that can not get inferred from multi-view images. Hence, our results produce much less artifacts even in the area that few images can cover. We also show our rendered images as reference.
<table>
<thead>
<tr>
<th>Method</th>
<th>24</th>
<th>37</th>
<th>40</th>
<th>55</th>
<th>63</th>
<th>65</th>
<th>69</th>
<th>83</th>
<th>97</th>
<th>105</th>
<th>106</th>
<th>110</th>
<th>114</th>
<th>118</th>
<th>122</th>
<th>Mean</th>
</tr>
</thead>
<tbody>
<tr>
<td>COLMAP [59]</td>
<td>0.81</td>
<td>2.05</td>
<td>0.73</td>
<td>1.22</td>
<td>1.79</td>
<td>1.58</td>
<td>1.02</td>
<td>3.05</td>
<td>1.40</td>
<td>2.05</td>
<td>1.00</td>
<td>1.32</td>
<td>0.49</td>
<td>0.78</td>
<td>1.17</td>
<td>1.36</td>
</tr>
<tr>
<td>NeRF [46]</td>
<td>1.90</td>
<td>1.60</td>
<td>1.85</td>
<td>0.58</td>
<td>2.28</td>
<td>1.27</td>
<td>1.47</td>
<td>1.67</td>
<td>2.05</td>
<td>1.07</td>
<td>0.88</td>
<td>2.53</td>
<td>1.06</td>
<td>1.15</td>
<td>0.96</td>
<td>1.49</td>
</tr>
<tr>
<td>UNISURF [49]</td>
<td>1.32</td>
<td>1.36</td>
<td>1.72</td>
<td>0.44</td>
<td>1.35</td>
<td>0.79</td>
<td>0.80</td>
<td>1.49</td>
<td>1.37</td>
<td>0.89</td>
<td>0.59</td>
<td>1.47</td>
<td>0.46</td>
<td>0.59</td>
<td>0.62</td>
<td>1.02</td>
</tr>
<tr>
<td>VolSDF [73]</td>
<td>1.14</td>
<td>1.26</td>
<td>0.81</td>
<td>0.49</td>
<td>1.25</td>
<td>0.70</td>
<td>0.72</td>
<td>1.29</td>
<td>1.18</td>
<td>0.70</td>
<td>0.66</td>
<td>1.08</td>
<td>0.42</td>
<td>0.61</td>
<td>0.55</td>
<td>0.86</td>
</tr>
<tr>
<td>NeuS [68]</td>
<td>0.88</td>
<td>0.90</td>
<td>0.80</td>
<td>0.41</td>
<td>1.13</td>
<td>0.63</td>
<td>0.58</td>
<td>1.37</td>
<td>1.157</td>
<td>0.83</td>
<td>0.51</td>
<td>1.26</td>
<td>0.33</td>
<td>0.48</td>
<td>0.52</td>
<td>0.78</td>
</tr>
<tr>
<td>Ours(NeuS)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Ours(MonoSDF)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Table 4. Numerical comparison with baselines in DTU dataset. The bars above numbers indicate the best.

<table>
<thead>
<tr>
<th>Weight $\alpha$</th>
<th>0</th>
<th>0.001</th>
<th>0.01</th>
<th>0.1</th>
<th>1.0</th>
</tr>
</thead>
<tbody>
<tr>
<td>CD $\times$ 100</td>
<td>0.586</td>
<td>0.550</td>
<td><strong>0.394</strong></td>
<td>0.433</td>
<td>0.601</td>
</tr>
<tr>
<td>NC</td>
<td>0.955</td>
<td>0.955</td>
<td><strong>0.958</strong></td>
<td>0.957</td>
<td>0.940</td>
</tr>
</tbody>
</table>

Table 6. Effect of weight $\alpha$.

We further evaluate our method in ScanNet. We report average performance over each scene in Tab. 5. The numerical comparison show that we achieve the best performance among the state-of-the-art methods. Visual comparisons in Fig. 8 show that better gradient consistency reveals more geometry details.

4.3. Ablation Studies

We conduct ablation studies to justify the effectiveness of modules in our method. We use NeuralPull as a baseline and train it using our loss as one term in the loss function. We report our ablation studies under 3D scene dataset.

Weights. We explore the effect of our loss by adjusting the weight $\alpha$ in Eq. 6. We report our results with different candidates $\{0, 0.001, 0.01, 0.1, 1.0\}$. The comparison in Tab. 6 shows that our level set alignment loss can improve the accuracy of inferred SDFs, it may affect the optimization to converge if we weight it too much.

Adaptive per Point Weights. We show the effect of the adaptive weight $\beta_q$ for each query in Tab. 7. We report the result without the weight $\beta_q$, and the result with $\beta_q$ that is obtained using the nearest distance to the point cloud rather than the predicted signed distance. The result of “0” indicates that weighting queries nearer to the surface more is important for the level set alignment, since all level sets are aligned to the zero level set. The result of “Euclidean” indicates that using inferred SDF achieves comparable results (the results of “10”) with using its nearest distance to the point cloud, but finding the nearest point for each query may increase the computational burden in large scale point clouds. We also compare the decay parameters $\delta$ to obtain $\beta_q$ in Eq. 7, and $\delta = 10$ performs the best.

Consistency with Surface Points. We further justify how we compute gradient consistency. We report results with maximizing consistency between gradients at queries and gradients at their nearest points on the surface, rather than their projections on the zero level set. Compared to the projections on the zero level set which is optimized in different iterations, the nearest point on surface is fixed. In Tab. 8, the results of “Fixed” degenerate from the results of “Cosine”. The reason is that, during the early stage of optimization, the surface may not be the zero level set of the learned SDF, which brings lots of ambiguity and conflict if we use the nearest point as a reference. Hence, using projections on the zero level set in current iteration produces better results.

Cosine Distance. We show the advantages of cosine distance in Eq. 4. We replace cosine distance using a mean squared error with normalized gradients or with gradients without normalization. The results of “MSE-Norm” and “MSE” in Tab. 8 show that cosine distance performs better than MSE in SDF inference.

Constraint on Second Order Derivatives. We compare our loss with the constraints on second order derivatives in [4] which aims to smooth the change of gradients. Although our loss also involves second order derivatives during gradient descent, we do not explicitly add constraints on the second order derivatives, which may result in unstable optimization. The comparison with the results of “DiGS” and the results of “DiGS+Cosine” indicate that our loss can reveal more accurate SDFs than the constraint on second order derivatives. The visual comparison with [4] in Fig. 9 shows that the constraint on second order derivatives can not achieve more compact and sharper surfaces as ours.
5. Conclusion

We improve the learning of SDFs without signed distance supervision by pursuing better gradient consistency. Our analysis shows that consistent gradients in the field are the key factor affecting the accuracy of inferred SDFs. To evaluate the gradient consistency, we introduce a level set alignment loss. By minimizing our loss, we successfully align all level sets onto the zero level set, which propagates the zero level set to eliminate 3D ambiguity through better gradient consistency. Our loss can be applied upon different methods as a general term in loss function to improve the gradient consistency in the SDFs inferred from 3D point clouds or multi-view images. The visual and numerical comparisons with the state-of-the-art methods justify our effectiveness and show our superiority over the latest methods in SDF inference.
References


[10] Chao Chen, Zhizhong Han, Yu-Shen Liu, and Matthias Zwicker. Unsupervised learning of fine structured 3D point clouds by 2D projection matching. In IEEE International Conference on Computer Vision, 2021. 2


[25] Zhizhong Han, Chao Chen, Yu-Shen Liu, and Matthias Zwicker. ShapeCaptioner: Generative caption network for 3D shapes by learning a mapping from parts detected in multiple views to sentences. ArXiv, abs/1908.00120, 2019. 2


