Towards Unbiased of Neural Implicit Surfaces Reconstruction

Haolun Lan(36920231153204, AI), Yu Qin(36920231153228, AI), Weijia Zhou(36920231153269, AI)*

[1]Institute of Artificial Intelligence, Xiamen University, China

Abstract

Recent works on implicit neural representations have made significant strides. Learning implicit neural surfaces using volume rendering has gained popularity in multi-view reconstruction. However, accurately recovering fine details is still challenging, due to the underlying ambiguity of geometry and appearance representation. In this paper, we present TU-NeuS, a volume rendering-based neural implicit surface reconstruction method capable of recovering fine geometry details, which extends NeuS by two additional loss functions targeting enhanced reconstruction quality. First, we encourage the rendered surface points from alpha compositing to have zero signed distance values, alleviating the geometry bias arising from transforming SDF to density for volume rendering. Second, we directly locate the zero-level set of SDF networks and explicitly perform multi-view geometry optimization by leveraging the sparse geometry from structure from motion in multi-view stereo. Extensive quantitative and qualitative results demonstrate that our method reconstructs high-accuracy surfaces with details, and outperforms the state of the art.

Introduction

Recent advances in implicit representation and neural rendering NeRF(Mildenhall et al. 2021) has provided a new alternative for geometric modeling and novel view rendering. However, applying the vanilla NeRF with the soft density representation to accurately reconstruct the geometry with fine-grained surface details remain challenging. In contrast, the neural implicit surface NeuS(Wang et al. 2021) was proposed to apply the signed distance function (SDF) rather than the soft density to model the object surface within the NeRF framework explicitly. The object surface is represented as the zero-level set of the SDF modeled by the multilayer perceptron (MLP). NeuS and its variants have shown that SDF can flexibly represent the scene geometry with arbitrary topologies, and produce significantly better results in neural surface reconstruction than the vanilla NeRF approach.

In this paper, we propose a Details recovering Neural implicit Surface reconstruction method named TU-NeuS, with two constraints to guide the SDF field-based volume rendering and thus improve the reconstruction quality. our method is able to reconstruct more accurate geometry details than the state of the art(Mildenhall et al. 2021; Wang et al. 2021).

To get rid of geometric errors of the standard volume rendering approaches, NeuS applies a weight function that is occlusion-aware and unbiased in the first-order approximation of SDF. However, we argue that the weight function under a non-linearly distributed SDF field causes bias between the geometric surface point and rendered surface point from alpha compositing. To this end, we propose a novel scheme to mitigate this bias. Specifically, we generate additional distance maps during the volume rendering, back-project the distance into 3D points, and penalize their absolute SDF values predicted by the geometry MLP network. By doing this, we encourage the consistency between volume rendering and the underlying surface.

Meanwhile, we directly locate the zero-level set of SDF networks and explicitly perform multi-view geometry optimization by leveraging the sparse geometry from structure from motion (SFM). Directly locating the zero-level set of SDF networks guarantees that our geometry modeling is unbiased. This enables our method to focus on true surface optimization.

To summarize, the main contributions of our work are as follows:

- We provide a theoretical analysis of the geometry bias resulting from the unregularized SDF field in a volume rendering-based neural implicit surface network, and propose a novel constraint to regularize this bias.
- Based on our theoretical analysis, we propose to directly locate the zero-level set of SDF networks and leverage multi-view geometry constraints to explicitly supervise the training of SDF networks. In this way, the SDF networks are encouraged to focus on true surface optimization.
- We evaluate qualitatively and quantitatively the proposed method on DTU (Mescheder et al. 2019) datasets, and show that it outperforms the state of the art, with high-accuracy surface reconstruction.

*Corresponding author

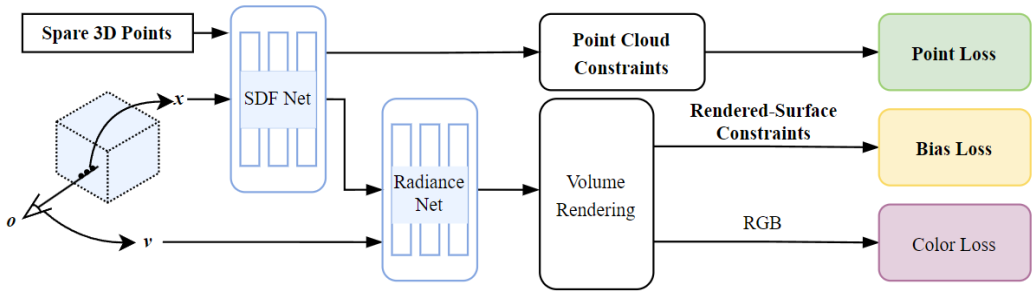

Figure 1: Overview of TU-NeuS. The modules in bold black font are our work. Previous neural implicit surfaces learning methods mainly depend on the color loss to implicitly supervise the SDF network. Our proposed TU-NeuS explicitly supervises the SDF network by introducing the Point loss from sparse 3D points and rendered-surface loss from rendered-surface sampling points.

Related work

Traditional multi-view 3D reconstruction.

Traditional multi-view 3D reconstruction is the classical pipeline of surface reconstruction from multi-view images. Given multi-view input images, traditional multi-view 3D reconstruction uses structure from motion (SFM)(Schonberger and Frahm 2016; Snavely, Seitz, and Szeliski 2006) to extract and match features of neighbor views, and estimate camera parameters and sparse 3D points. After that, multi-view stereo (MVS)(Furukawa and Ponce 2009; Xu and Tao 2019) is applied to estimate dense depth maps for each view, and then all the depth maps are fused into dense point clouds. With the development of deep learning, many attempts have been made but the problem still exists.

Implicit Surface Representation and Reconstruction.

The success of NeRF(Mildenhall et al. 2021) in representing a scene by 5D radiance field has recently drawn considerable attention from the community of both computer vision and computer graphics. Implicit neural representation leverages physics-based traditional volume rendering in a differentiable way, enabling photorealistic novel view synthesis without 3D supervision. While NeRF-like approaches(Mildenhall et al. 2021; Zhang et al. 2020; Barron et al. 2022) achieve impressive rendering quality, their underlying geometry is generally noisy and less favorable. To alleviate the above issue, the current implicit surface re- construction methods employ surface indicator functions, mapping continuous spatial coordinates to occupancy(Mescheder et al. 2019) and SDF (Wang et al. 2021; Mescheder et al. 2019), where Marching cubes(Lorensen and Cline 1998) is commonly applied to extract the implicit surface at any resolution. IDR(Yariv et al. 2020) renders the color of a ray only on the object surface point, and applies differentiable ray tracing to back-propagate the gradients to a local region near the intersection. VolSDF(Mescheder et al. 2019), NeuS(Wang et al. 2021) designs an occlusion-aware transformation function mapping signed distances to weights for volume rending, with a learnable parameter to control the slope of the logistic density function. However, this mapping function is only unbiased in a regularized SDF field which is linearly distributed, so we propose a novel constraint to compensate for the geometry bias. We build our framework on NeuS(Wang et al. 2021), but we believe our proposed method could be adapted to any volume rendering-based neural implicit surface reconstruction work.

Proposed Solution

Using multi-view images of an object, our objective is to reconstruct the surface through neural volume rendering. The object's spatial field is represented by a signed distance function (SDF), and the corresponding surface is obtained from the zero level set of the SDF. During volume rendering, our focus is on optimizing the signed distance function. In this section, we address the inherent bias in color rendering leading to inconsistencies between rendered colors and implicit geometry. To resolve this, we introduce explicit SDF optimization for achieving geometry consistency. Figure 1 provides an overview of our approach.

Bias in color rendering

During volume rendering, a discrepancy arises between the rendered colors and the object's geometry, resulting in inconsistency with the actual surface colors. For an opaque solid object $\Omega \in R^3$, the opacity can be represented by an indicator function $\mathcal{O}(\boldsymbol{p})$:

$$\mathcal{O}(\boldsymbol{p}) = \begin{cases} 1, \boldsymbol{p} \in \Omega \\ 0, \boldsymbol{p} \notin \Omega \end{cases} . \tag{1}$$

When observing or capturing colors through cameras, we perceive the light traveling along the light ray into our eyes or cameras. Utilizing the inherent optical properties of opaque solid objects, we approximate that the colors C in the image set $\{I_i\}$ correspond to the colors c of the object intersecting with the light ray $\boldsymbol{v}$ from the respective camera position $\boldsymbol{o}$:

$$C\left(\boldsymbol{o}, \boldsymbol{v}\right) = c\left(\boldsymbol{o} + t^* \boldsymbol{v}\right), \tag{2}$$

Here, $t^* = \text{argmin}\{t|\boldsymbol{o}+t\boldsymbol{v}=\boldsymbol{p},\, \boldsymbol{p}\in\partial\Omega,\, t\in(0,\,\infty)\}$. The term $\partial\Omega$ denotes the geometric surfaces. This assumption is justified as we can neglect light transmission through the opaque object. The light intensity experiences a sharp decay to nearly zero upon passing through the surface of the opaque object. We express the object's surface mathematically through the signed distance function. The signed distance function $sdf(\boldsymbol{p})$ represents the signed distance between a spatial point $\boldsymbol{p}$ and the surface $\partial\Omega$. Thus, the surface $\partial\Omega$ can be mathematically represented as:

$$\partial\Omega = \{\boldsymbol{p}|sdf\,(\boldsymbol{p}) = 0\}\,. \tag{3}$$

With neural volume rendering, we estimate the signed distance function $\hat{sdf}$ and color field $\hat{c}$ by Multi-Layer Perceptron (MLP) networks F_Θ and G_Φ :

$$\hat{sdf}\,(\boldsymbol{p}) = F_\Theta\,(\boldsymbol{p})\,, \tag{4}$$

$$\hat{c}\,(\boldsymbol{o},\,\boldsymbol{v},t) = G_\Phi\,(\boldsymbol{o},\,\boldsymbol{v},t)\,. \tag{5}$$

Thus the estimated colors of the image with camera position $\boldsymbol{o}$ can be represented as:

$$\hat{C} = \int_0^{+\infty} w\,(t)\,\hat{c}\,(t)dt, \tag{6}$$

where t is the depth along the ray that comes from $\boldsymbol{o}$ with the direction $\boldsymbol{v}$ and $w(t)$ is a weight for the point at t. For simplicity, the notes $\boldsymbol{o}$ and $\boldsymbol{v}$ are omitted. To obtain discrete counterparts of w and $\hat{c}$, we also sample t_i discretely along the ray and use the Riemann sum:

$$\hat{C} = \sum_{i=1}^{n} w\,(t_i)\,\hat{c}\,(t_i). \tag{7}$$

Notably, the goal of novel view synthesis is to make an accurate prediction of the colors $\hat{C}$, and bend efforts to minimize the difference between the colors of ground truth images C and the prediction $\hat{C}$:

$$C = \hat{C} = \sum_{i=1}^{n} w\,(t_i)\,\hat{c}\,(t_i). \tag{8}$$

In surface reconstruction tasks, what we concentrate more is the surface of the object rather than the color. In this way, the above formula can be rewritten as:

$$\begin{aligned} C &= \sum_{i=1}^{j-1} w\,(t_i)\,\hat{c}\,(t_i) + w\,(t_j)\,\hat{c}\left(\hat{t^*}\right) \\ &\quad + w\,(t_j)\left(\hat{c}\,(t_j) - \hat{c}\left(\hat{t^*}\right)\right) + \sum_{i=j+1}^{n} w\,(t_i)\,\hat{c}\,(t_i) \\ &= w\,(t_j)\,\hat{c}\left(\hat{t^*}\right) + \varepsilon_{sample} + \sum_{\substack{i=1\\ i\neq j}}^{n} w\,(t_i)\,\hat{c}\,(t_i) \\ &= w\,(t_j)\,\hat{c}\left(\hat{t^*}\right) + \varepsilon_{sample} + \varepsilon_{weight}, \end{aligned} \tag{9}$$

where $\hat{sdf}(\hat{t^*}) = 0$, t_j denotes the nearest sample point from $\hat{t^*}$, ε_{sample} denotes the bias caused by sampling operation and ε_{weight} denotes the bias caused by weighted sum operation of volume rendering. With Formula (2), it can be rewritten as:

$$w\,(t_j)\,\hat{c}\left(\hat{t^*}\right) + \varepsilon_{sample} + \varepsilon_{weight} = c\,(t^*)\,, \tag{10}$$

$$\hat{c}\left(\hat{t^*}\right) = \frac{c\,(t^*) - \varepsilon_{sample} - \varepsilon_{weight}}{w\,(t_j)}. \tag{11}$$

There the total bias between the colors of object surface and estimated surface is:

$$\begin{aligned} \Delta c &= \hat{c}\left(\hat{t^*}\right) - c\,(t^*) \\ &= \frac{(1-w\,(t_j))c\,(t^*) - \varepsilon_{sample} - \varepsilon_{weight}}{w\,(t_j)}. \end{aligned} \tag{12}$$

The relative bias is:

$$\delta c = \frac{\Delta c}{c\,(t^*)} = \frac{1}{w\,(t_j)} - 1 - \frac{\varepsilon_{sample} + \varepsilon_{weight}}{w\,(t_j)\,c\,(t^*)}. \tag{13}$$

When $w\,(t_j)$ approaches to 1, ε_{weight} approaches to 0 and δc approaches to $\varepsilon_{sample} c\,(t^*)$. In this case, the total bias is only caused by discrete sampling, which is small. So TUNeuS adopts a simple solution which is to directly use the geometry of the object for supervision.

Point cloud Constraint

The SDF network, which estimates the signed distance from any spatial point to the surface of the object, is the key network that we need to optimize. So we propose an explicit supervision method on the SDF network to ensure its accuracy directly with points in 3D space. For less extra cost, we use points generated by structure from motion (SFM) (Schonberger and Frahm 2016; Snavely, Seitz, and Szeliski 2006) to supervise the SDF network.

Since our focus is on opaque objects, certain parts of these objects may be invisible from the viewpoint of a specific camera position. Consequently, only a subset of sparse points is visible for each view. For an image I_i captured from the camera position $\boldsymbol{o}_i$, the visible points $\boldsymbol{P}_i$ align with the feature points $\boldsymbol{X}_i$ of I_i:

$$\boldsymbol{X}_i = \boldsymbol{K}_i\,[\boldsymbol{R}_i|\boldsymbol{t}_i]\,\boldsymbol{P}_i, \tag{14}$$

where $\boldsymbol{K}_i$ is the internal calibration matrix, $\boldsymbol{R}_i$ is the rotation matrix and $\boldsymbol{t}_i$ is the translation vector for image I_i. The coordinates of $\boldsymbol{X}_i$ and $\boldsymbol{P}_i$ are all homogeneous coordinates. The scale index before $\boldsymbol{X}_i$ is omitted for simplicity. According to feature points of each image, we get visible points for each view and use them to supervise the SDF network while rendering image from the corresponding view.

While rendering image I_i from view V_i, we use the SDF network to estimate SDF values for the visible points $\boldsymbol{P}_i$ of V_i. Based on the approximation that the SDF values of sparse points are zeroes, we propose the point cloud loss:

$$\begin{aligned} \mathcal{L}_{point} &= \sum_{\boldsymbol{p}_j\in\boldsymbol{P}_i} \frac{1}{N_i} |\hat{sdf}\,(\boldsymbol{p}_j) - sdf\,(\boldsymbol{p}_j)\,| \\ &= \sum_{\boldsymbol{p}_j\in\boldsymbol{P}_i} \frac{1}{N_i} |\hat{sdf}\,(\boldsymbol{p}_j)\,|, \end{aligned} \tag{15}$$

where N_i is the number of points in $\boldsymbol{P}_i$ and $|\cdot|$ denotes the L_1 distance. It is worth noting that the loss we use to supervise the SDF network varies according to the view being rendered. In this way, the introduced SDF loss is consistent with the process of color rendering.

With the explicit supervision on the SDF network, our network could converge faster owing to the use of geometry prior. Besides, because the complex geometric structures with strong textures are the concentrated distribution areas of the sparse points, our method could capture more meticulous geometries.

Rendered-surface Constraints

To address rendered-surface bias, we propose a novel strategy to regulate the SDF field for volume rendering by mitigating the mentioned rendered-surface bias. By considering a 3D point along a ray, we can render the distance $t_{rendered}$ between the camera center and the average point for volume rendering through discretizing the volume integration:

$$t_{rendered} = \sum_{i}^{n} \frac{\omega_i t_i}{\sum_i^n \omega_i}, \tag{16}$$

where n is the number of sampling points along a ray, ω_i represents the discrete counterpart of the weight in $\omega(t) = exp\left(-\int_0^t \sigma(u)du\right)\sigma(t)$, and t_i is the distance from a sampling point to the camera center. Then the volume-rendered surface point $\mathbf{x}_{renderd}$ can be formed by back-projection:

$$\mathbf{x}_{rendered} = \mathbf{o} + t_{rendered}\mathbf{v}. \tag{17}$$

Finally, we build a rendered-surface bias loss:

$$\mathcal{L}_{bias} = \frac{1}{|S|} \sum_{\mathbf{x}_{rendered} \in S} |f(\mathbf{x}_{rendered})|, \tag{18}$$

where f is the geometry network outputting SDF values, S is the subset of $\mathbf{x}_{rendered}$ where ray-surface intersection has been found. By penalizing the absolute value of SDF of the rendered surface points, we encourage the geometry consistency between the implicit SDF field and the radiance field for volume rendering. Intuitively, this constraint regularizes the SDF distribution for unbiased volume rendering, and thus leads to more accurate surface reconstruction. It is also worth noting, that Eikonal loss (Gropp et al. 2020a) widely used in neural implicit surface reconstruction regularizes the gradient field of SDF by constraining the gradient norm. Both Eikonal loss and our rendered-surface bias loss support each other, enhancing the reconstruction quality.

Loss function

During rendering colors from a specific view, our total loss is:

$$\mathcal{L} = \mathcal{L}_{color} + \alpha\mathcal{L}_{reg} + \beta\mathcal{L}_{point} + \gamma\mathcal{L}_{bias}. \tag{19}$$

$\mathcal{L}_{color}$ is the difference between the ground truth colors and the rendered colors:

$$\mathcal{L}_{color} = \frac{1}{N}\sum_{i=1}^{N} |C_i - \hat{C}_i|. \tag{20}$$

And $\mathcal{L}_{reg}$ is an eikonal term (Gropp et al. 2020b) to regularize the gradients of SDF network:

$$\mathcal{L}_{reg} = \frac{1}{N}\sum_{i=1}^{N} (|\nabla \hat{sdf}(\boldsymbol{p}_i)| - 1)^2. \tag{21}$$

In our experiments, we choose α, β and γ as 0.3, 1.0 and 0.5 respectively.

Experiments

Experimental setting

Datasets. Following established practices (Yariv et al. 2020; Wang et al. 2021; Yariv et al. 2021), we perform surface reconstruction using 15 scans from the DTU dataset (Aanæs et al. 2016) to assess our method. The DTU dataset comprises objects of diverse categories, exhibiting variations in appearance and geometries. Each scan includes 49 or 64 images at a resolution of 1200 × 1600, accompanied by camera parameters. Additionally, we conduct tests on 7 challenging scenes from the low-res set of the BlendedMVS dataset (Yao et al. 2020). BlendedMVS scenes feature varying numbers of views and camera parameters, captured by images at a resolution of 768 × 576, with view counts ranging from 31 to 143. For evaluation on the DTU dataset, we assess our reconstructed surfaces using the Chamfer Distance provided by DTU evaluation metrics (Aanæs et al. 2016). Regarding the BlendedMVS dataset, we present visual representations of the reconstructed surfaces to illustrate their effects.

Scan	COLMAP	NeRF	NeuS	Ours
24	0.81	1.90	1.00	0.44
37	2.05	1.60	1.37	0.79
40	0.73	1.85	0.93	0.35
55	1.22	0.58	0.43	0.39
63	1.79	2.28	1.10	0.88
65	1.58	1.27	0.65	0.58
69	1.02	1.47	0.57	0.55
83	3.05	1.67	1.48	1.35
97	1.40	2.05	1.09	0.91
105	2.05	1.07	0.83	0.76
106	1.00	0.88	0.52	0.40
110	1.32	2.53	1.20	0.72
114	0.49	1.06	0.35	0.31
118	0.78	1.15	0.49	0.39
122	1.17	0.96	0.54	0.39
mean	1.36	1.49	0.84	0.61

Table 1: Results on DTU scenes. The surfaces produced by colmap are trimmed with trimming value 7.

Baselines. To better evaluate our method, we compare it with the-state-of-art learning-based methods and the traditional reconstruction method, colmap (Schönberger et al.

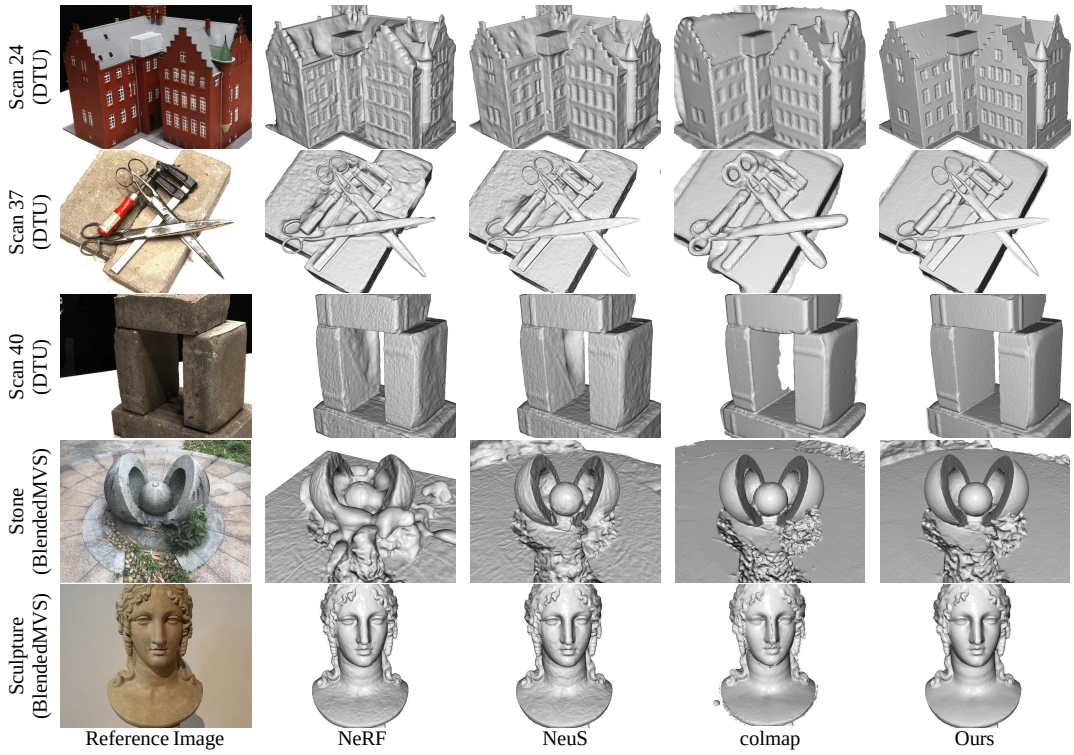

Figure 2: Surfaces reconstructed on DTU and BlendedMVS. We use NueS trained with mask supervision and colmap with trimming value 7.

2016). For learning-based methods, we compare with NeRF, NeuS and Colmap. For colmap, we use the reconstructed surface with trim parameter 7 (the best performance).

Comparisons

We compare the reconstruction quality with the Chamfer distances of our method and baselines on DTU dataset. Table 1 shows the quantitative results. Notably, our method outperforms baselines by a large margin. Specifically, it outperforms state-of-the-art neural implicit surfaces learning methods by over 25% and outperforms the traditional method colmap by 22%. As shown qualitatively in Fig. 2, our method achieves high-quality surface reconstruction in both complex thin structures and large smooth regions. For example, our method can recover abrupt depth changes in Scan 37 and reconstruct planar structures in Scan 24 and 40. To test the capability of handling various scenes, we test on 7 challenging scenes of the BlendedMVS dataset. Qualitative results in Fig. 2 show that our method yields more smooth and consistent surface quality than other methods.

Ablation Study

To evaluate the effect of our proposed contributions, we conduct an ablation study on DTU dataset. NeuS is adopted as our baseline. Different modules are progressively added to the baseline to investigate their efficacy. Results are reported in Table 2. We see that, with rendered-surface constraint , Model-A has begun to 0.76. With the proposed the point coloud constraint, Model-B can lead to much more performance improvement.

	$\mathcal{L}_{color}$	$\mathcal{L}_{bias}$	$\mathcal{L}_{point}$	Mean Chamfer
Baseline	✓			0.84
Model-A	✓	✓		0.76
Model-B	✓		✓	0.63
TU-NeuS	✓	✓	✓	**0.61**

Table 2: Ablation study on DTU scenes.

Conclusion

We introduce TU-NeuS, a volume rendering-based neural implicit surface reconstruction method recovering fine-level geometric details. We analyze the cause for geometry bias between the SDF field and the volume rendered color, and propose two novel loss functions to constrain the bias. Extensive experiments on different datasets show that TU-NeuS is able to reconstruct high-quality surfaces with fine details and outperforms the state of the art both qualitatively and quantitatively.

References

Aanæs, H.; Jensen, R. R.; Vogiatzis, G.; Tola, E.; and Dahl, A. B. 2016. Large-scale data for multiple-view stereopsis. *International Journal of Computer Vision*, 120(2): 153–168.

Barron, J. T.; Mildenhall, B.; Verbin, D.; Srinivasan, P. P.; and Hedman, P. 2022. Mip-nerf 360: Unbounded anti-aliased neural radiance fields. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 5470–5479.

Furukawa, Y.; and Ponce, J. 2009. Accurate, dense, and robust multiview stereopsis. *IEEE transactions on pattern analysis and machine intelligence*, 32(8): 1362–1376.

Gropp, A.; Yariv, L.; Haim, N.; Atzmon, M.; and Lipman, Y. 2020a. Implicit geometric regularization for learning shapes. *arXiv preprint arXiv:2002.10099*.

Gropp, A.; Yariv, L.; Haim, N.; Atzmon, M.; and Lipman, Y. 2020b. Implicit geometric regularization for learning shapes. *arXiv preprint arXiv:2002.10099*.

Lorensen, W. E.; and Cline, H. E. 1998. Marching cubes: A high resolution 3D surface construction algorithm. In *Seminal graphics: pioneering efforts that shaped the field*, 347–353.

Mescheder, L.; Oechsle, M.; Niemeyer, M.; Nowozin, S.; and Geiger, A. 2019. Occupancy networks: Learning 3d reconstruction in function space. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 4460–4470.

Mildenhall, B.; Srinivasan, P. P.; Tancik, M.; Barron, J. T.; Ramamoorthi, R.; and Ng, R. 2021. Nerf: Representing scenes as neural radiance fields for view synthesis. *Communications of the ACM*, 65(1): 99–106.

Schonberger, J. L.; and Frahm, J.-M. 2016. Structure-from-motion revisited. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 4104–4113.

Schönberger, J. L.; Zheng, E.; Frahm, J.-M.; and Pollefeys, M. 2016. Pixelwise View Selection for Unstructured Multi-View Stereo. In *Proceedings of the European Conference on Computer Vision*, 501–518.

Snavely, N.; Seitz, S. M.; and Szeliski, R. 2006. Photo tourism: exploring photo collections in 3D. In *ACM siggraph 2006 papers*, 835–846.

Wang, P.; Liu, L.; Liu, Y.; Theobalt, C.; Komura, T.; and Wang, W. 2021. Neus: Learning neural implicit surfaces by volume rendering for multi-view reconstruction. *arXiv preprint arXiv:2106.10689*.

Xu, Q.; and Tao, W. 2019. Multi-scale geometric consistency guided multi-view stereo. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 5483–5492.

Yao, Y.; Luo, Z.; Li, S.; Zhang, J.; Ren, Y.; Zhou, L.; Fang, T.; and Quan, L. 2020. Blendedmvs: A large-scale dataset for generalized multi-view stereo networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 1790–1799.

Yariv, L.; Gu, J.; Kasten, Y.; and Lipman, Y. 2021. Volume rendering of neural implicit surfaces. *Advances in Neural Information Processing Systems*, 34: 4805–4815.

Yariv, L.; Kasten, Y.; Moran, D.; Galun, M.; Atzmon, M.; Ronen, B.; and Lipman, Y. 2020. Multiview neural surface reconstruction by disentangling geometry and appearance. In *Advances in Neural Information Processing Systems*, volume 33, 2492–2502.

Zhang, K.; Riegler, G.; Snavely, N.; and Koltun, V. 2020. Nerf++: Analyzing and improving neural radiance fields. *arXiv preprint arXiv:2010.07492*.